

Specyfikacja OPL

API Trouble Ticket Management

Na podstawie TMF621

Release 1.0.8

Wrzesień 2020

Latest Update:	Orange Approved

NOTICE

TABLE OF CONTENTS

NOTICE.....	3
TABLE OF CONTENTS	4
LIST OF TABLES	6
INTRODUCTION	7
RESOURCE MODEL.....	9
Managed Entity and Task Resource Models.....	9
TroubleTicket resource	9
AssistTicket resource.....	17
FaultTicket resource.....	17
ComplaintTicket resource	18
MassFaultTicket resource	18
CancelAssistTicketTask, CancelFaultTicketTask resources.....	21
ErrorRepresentation	24
Notification Resource Models	26
Trouble Ticket Change Notification.....	26
Trouble Ticket Status Change Notification.....	27
Trouble Ticket Information Required Notification.....	27
Trouble Ticket Resolved Notification	28
Cancel Assist Ticket Task State Change Notification	28
Cancel Fault Ticket Task State Change Notification	28
API OPERATIONS.....	30
Operations on Trouble Ticket	30
Create Trouble Ticket.....	30
Patch Trouble Ticket.....	33
Operations on CancelAssistTicketTask, CancelFaultTicketTask.....	36
Create CancelAssitTicketTask.....	36
Create CancelFaultTicketTask	38
API NOTIFICATIONS	42
Procesowanie zgłoszeń.....	43
Uszkodzenia	44
Awaria masowa	55
Reklamacja.....	62
Asysta techniczna	69
Słowniki.....	78

RR-FIA – kody odrzucenia dla uszkodzeń, asysty i awarii masowej od OA.....	78
EC-FIA – kody wyjścia dla uszkodzeń, reklamacji i asysty	79
FAULT-CANCEL-REASON – kody anulowania uszkodzeń/asyst kierowane do operatora	80
CANCEL-REASON-CODE – kody anulowania uszkodzeń/asyst podawane przez operatora	81
COMPLAINT-SUBJECT – przedmiot reklamacji	81
SYMPTOMS – symptom uszkodzenia.....	81

LIST OF TABLES

N/A

INTRODUCTION

Poniższy dokument opisuje specyfikację interfejsu API REST zasobu zgłoszenia problemu – Trouble Ticket. Obejmuje on definicję modelu, cykl jego życia oraz wszystkie dostępne dla niego operacje. Na zasobie zgłoszenia można wykonać typowe dla REST API operacje: tworzenie zgłoszenia, częściowa aktualizacja zgłoszenia.

Zgłoszenie posiada następujące cechy:

- reprezentuje problem zgłaszany przez operatora, który należy rozwiązać. Zasób zgłoszenia używany jest do zgłaszania uszkodzeń, reklamacji, asyst, awarii.
- główne atrybuty zgłoszenia to jego opis, dotkliwość, rodzaj, powiązane daty (utworzenie, oczekiwane rozwiązanie, rozwiązanie), stan i powiązane informacje (powód zmiany i data zmiany), podmioty powiązane (inicjator, właściciel itp.), powiązane podmioty (produkt, zamówienie produktu) i notatki.

REST API wykonuje na zasobie zgłoszenia następujące działania:

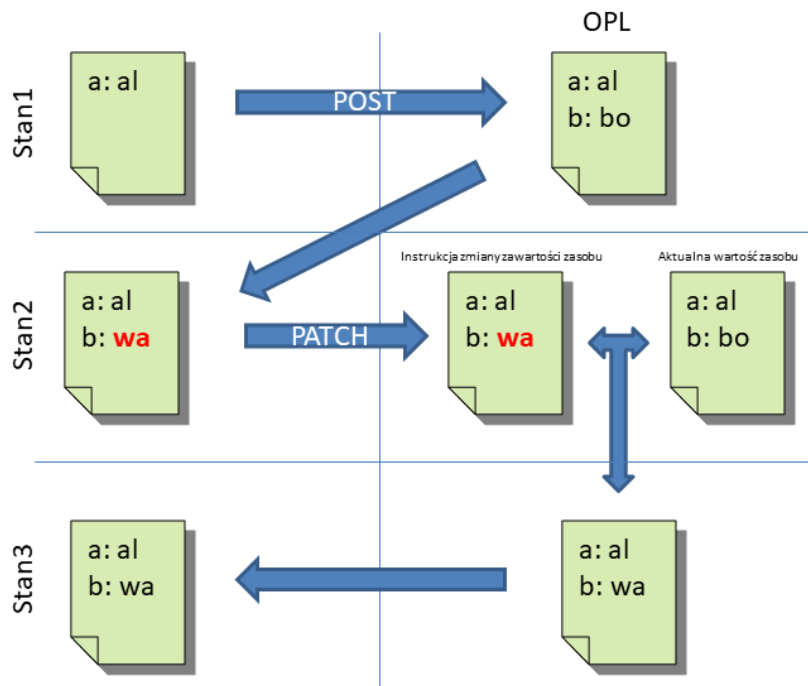
- Częściowe zaktualizowanie zgłoszenia.
- Utworzenie zgłoszenia.
- Wykonanie zadania anulowania zgłoszenia.
- Wykonanie powiadomień dystrybuowanych do odbiorców powiadomień:
 - zmiana stanu
 - zmiana atrybutów innych niż stan
 - rozwiązano
 - utworzono
 - konieczne uzupełnienie zasobu o atrybuty

W czasie implementacji należy pamiętać o pryncypiach API REST tj:

- Zasoby nie są komunikatami przesyłanymi i interpretowanymi przez obie strony a realnymi bytami na poziomie modelu danych OPL które posiadają rzeczywiste identyfikatory.
- Wykonywane operacje są atomowe i przyporządkowane http verbs. Metody http POST, GET, PATCH posiadają odpowiednie znaczenie (Create, Read, Update) i tylko w takim kontekście mogą być używane.
- Przesyłane są obrazy, stany zasobu między stronami (GET, POST) lub instrukcje jego modyfikacji (PATCH).
- Obie strony komunikacji zobowiązane są do przechowywania zgodnego stanu zasobu tj. zgodnych wartości atrybutów i ich list.
- Obiekty dla operacji PATCH podlegają walidacji dopiero po wykonaniu MERGE co skutkuje tym że atrybuty nie zmieniane a niedopuszczane do modyfikacji gdy pojawiają się w reprezentacji zmiany nie spowodują błędu.
- Wszystkie daty występujące w komunikacji są datami UTC/GMT.

- PATCH jest metodą idempotentną, wielokrotne wywołanie metody PATCH z tym samym stanem obrazu nie powoduje zmiany stanu tego zasobu . Metoda zawsze zwróci aktualny stan zasobu.
- Usługa REST API zawsze zwraca Etag który jest wyznaczony jako funkcja skrótu aktualnego stanu zasobu. W żądaniu PATCH operator musi zamieścić w nagłówku **If-Match** zawartość pozyskanego wcześniej Etag. Gdy wartość Etag nie zgadza się z zawartością aktualną zasobu operacja aktualizacji nie zostanie wykonana a klient otrzyma informację 412 Precondition Failed wraz z reprezentacją stanu aktualnego zasobu.

Zalecana obsługa przekazywania stanu zasobu pomiędzy OA a OPL:



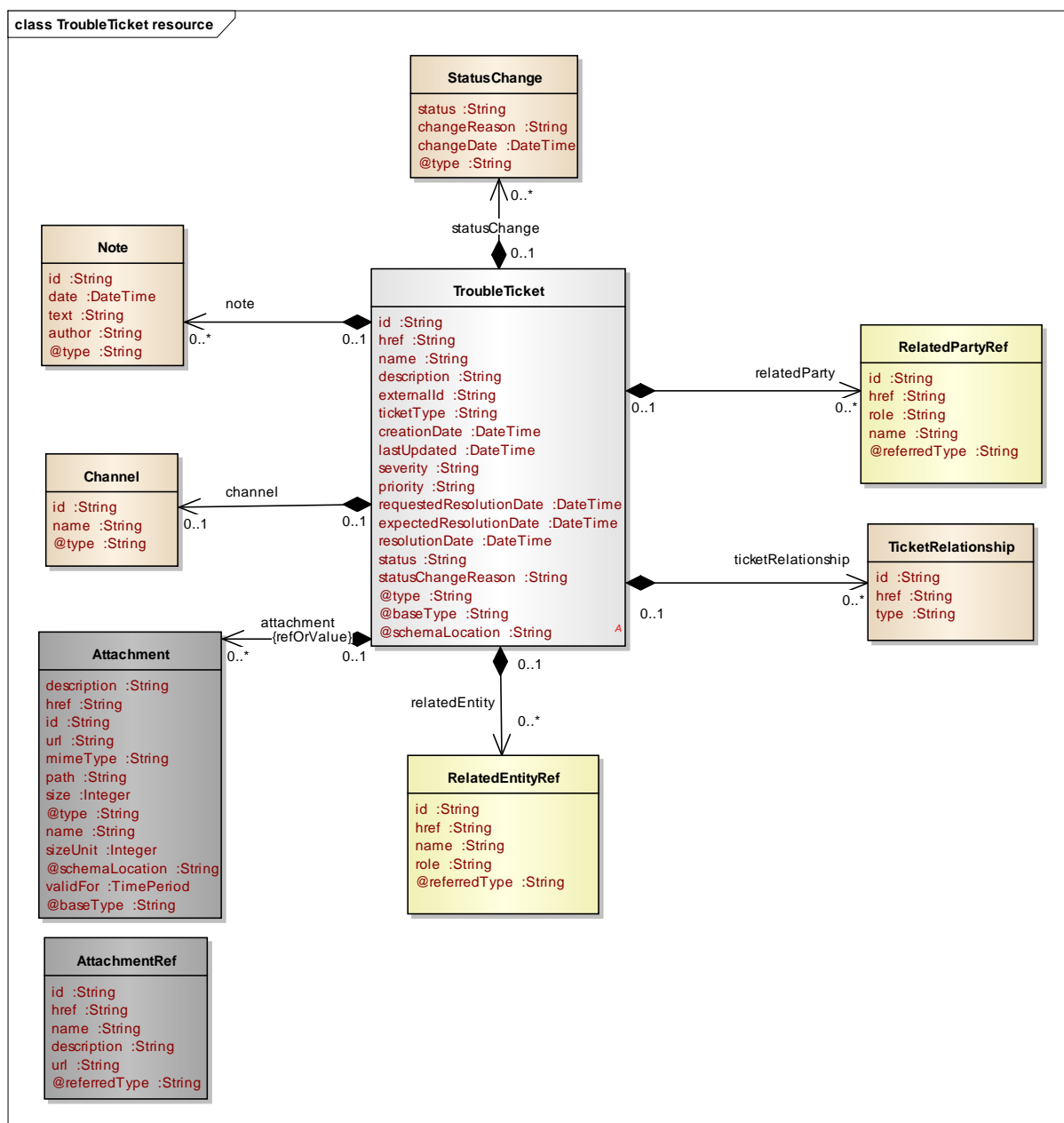
RESOURCE MODEL

Managed Entity and Task Resource Models

TroubleTicket resource

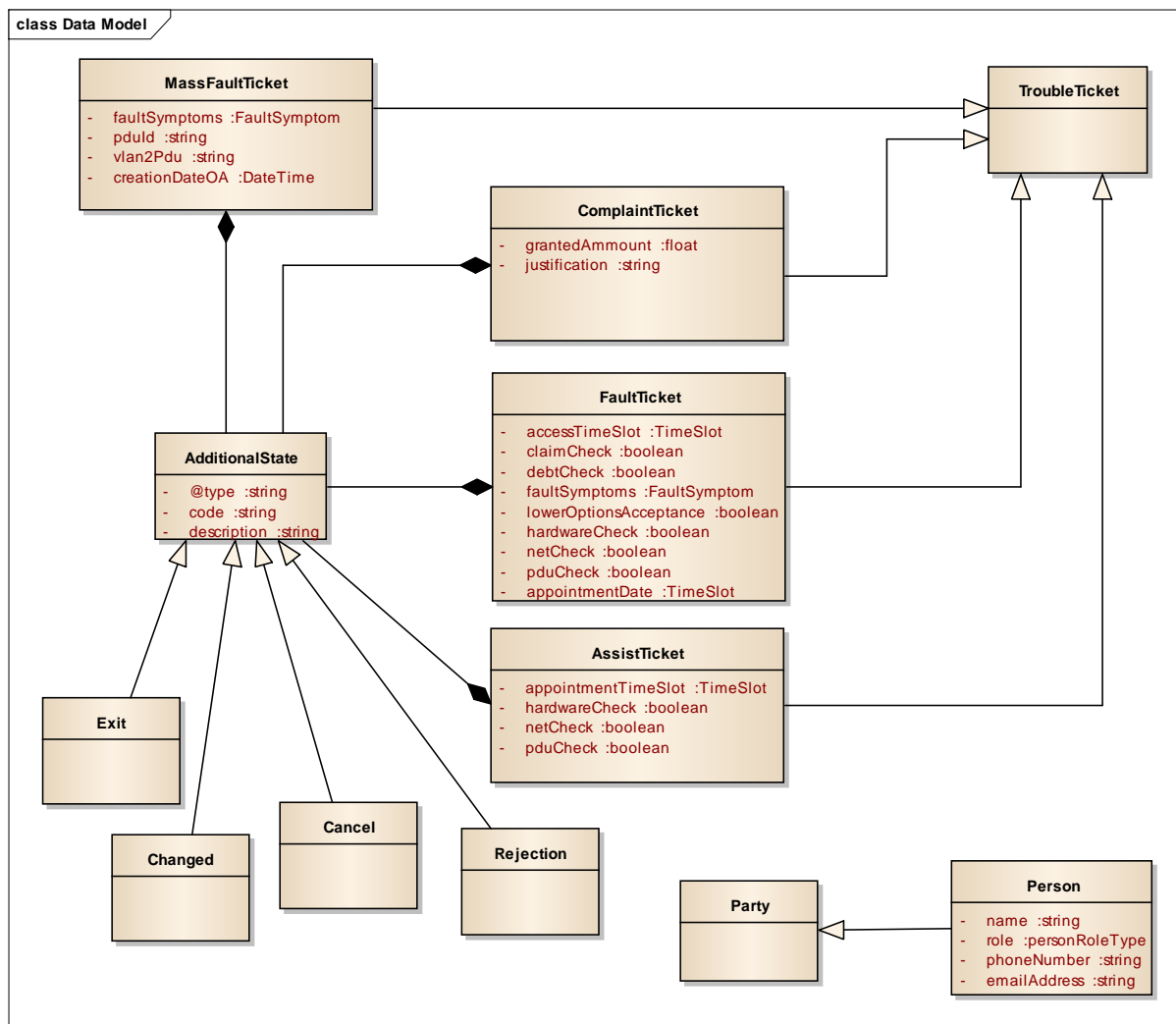
TroubleTicket jest zasobem reprezentującym zgłoszenie. W trakcie swojego cyklu życiowego jest on tworzony i zarządzany przez system klasy IMS (np. Issue Tracking System) w OPL. Model encji zasobu przedstawiony jest na poniższym diagramie.

Resource model



Kolorem szarym zaznaczono zasoby które nie są obsługiwane w obecnej wersji API.

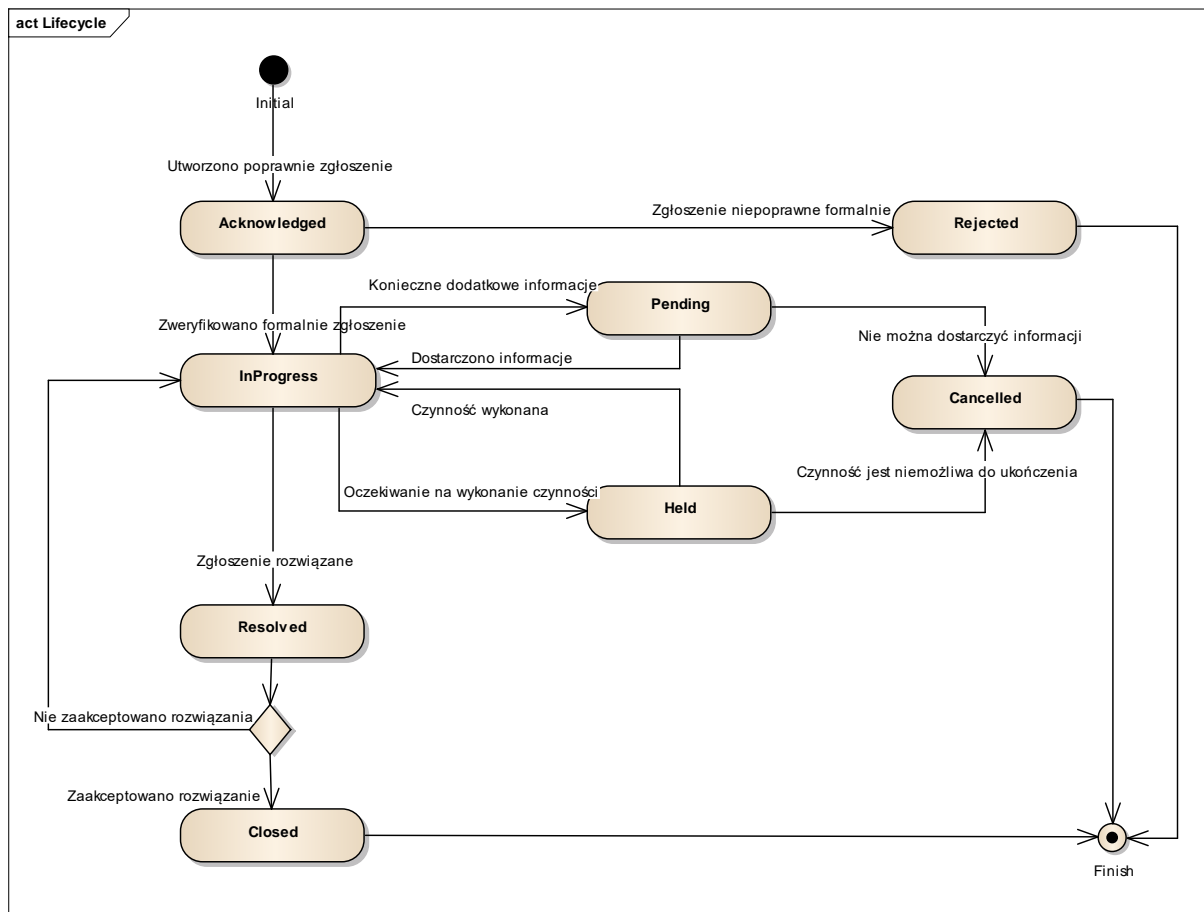
Standardowy model zasobu zgłoszenia zostanie rozszerzony o nowe klasy dziedziczące atrybuty z **TroubleTicket**.



Life cycle

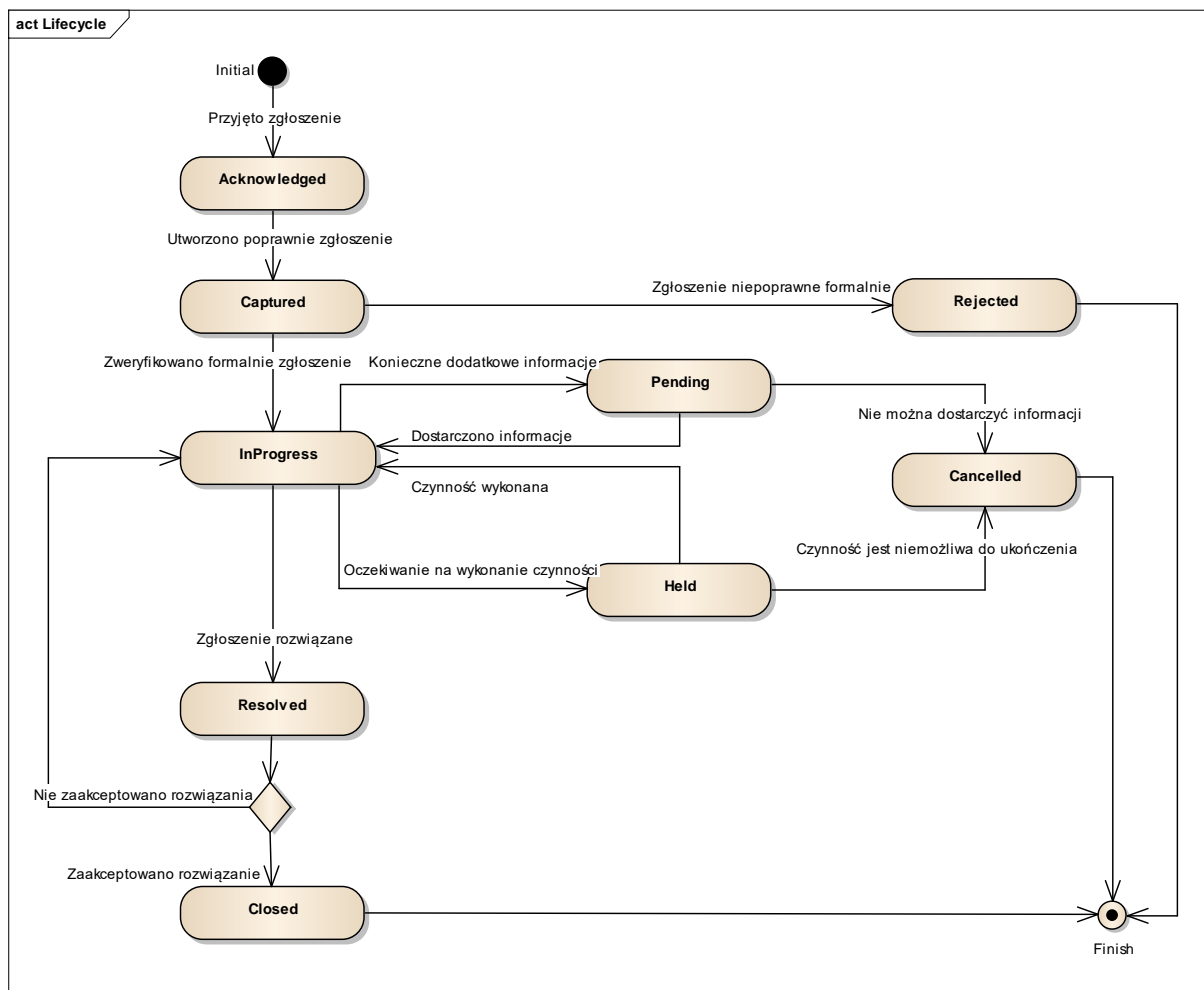
Ze względu na różnice związane z procesowaniem zgłoszeń różnych typów zgłoszenia dzielą się na dwie grupy posiadające różne cykle życiowe:

1) ComplaintTicket, MassFaultTicket, AssistTicket



- **acknowledged** – przyjęcie zgłoszenia poprawnie zweryfikowanego informatycznie.
- **rejected** – odrzucenie zgłoszenia ze względów formalnych.
- **inprogress** – rozpoczęcie obsługi zgłoszenia po zakończonej pozytywnie weryfikacji formalnej.
- **pending** – oczekiwanie na dostarczenie dodatkowych informacji.
- **held** – oczekiwanie na zakończenie aktywności poza systemem zgłoszeniowym.
- **cancelled** – anulowanie zgłoszenia.
- **resolved** – rozwiązanie zgłoszenia.
- **closed** – zamknięcie zgłoszenia.

2) FautTicket



- **acknowledged** – przyjęcie zgłoszenia poprawnie zweryfikowanego informatycznie.
- **captured** – rozpoczęcie weryfikacji formalnej.
- **rejected** – odrzucenie zgłoszenia ze względów formalnych.
- **inprogress** – rozpoczęcie obsługi zgłoszenia po zakończonej pozytywnie weryfikacji formalnej.
- **pending** – oczekiwanie na dostarczenie dodatkowych informacji.
- **held** – oczekiwanie na zakończenie aktywności poza systemem zgłoszeniowym.
- **cancelled** – anulowanie zgłoszenia.
- **resolved** – rozwiązanie zgłoszenia.
- **closed** – zamknięcie zgłoszenia.

TroubleTicket resource

id <i>required</i>	Unikalny identyfikator zasobu nadawany przez OPL.
name <i>optional</i>	Nazwa zgłoszenia, krótki opis zgłoszenia.
externalId <i>optional</i>	Identyfikator zgłoszenia w systemie zewnętrznym jeśli istnieje.
reason <i>optional</i>	Powód utworzenia zgłoszenia.
ticketType <i>required</i>	Typ zgłoszenia: technicalAssist, fault, complaint, massFault.
creationDate <i>required</i>	Data utworzenia zgłoszenia (UTC).
lastUpdate <i>required</i>	Data aktualizacji zgłoszenia (UTC).
description <i>required</i>	Opis zgłoszenia.
severity <i>required</i>	Wpływ na proces biznesowy. Dopuszczalne wartości: critical , major , minor .
priority <i>required</i>	Priorytet zgłoszenia: high , low .
requestedResolutionDate <i>required</i>	Wymagana data rozwiązania zgłoszenia. Data wypełniana na podstawie KPI przez OPL.
expectedResolutionDate <i>required</i>	Spodziewana data rozwiązania zgłoszenia pochodząca z OPL.
resolutionDate <i>optional</i>	Faktyczna data rozwiązania zgłoszenia.
status <i>required</i>	Aktualny stan zgłoszenia. Maszyna stanów zgłoszeń przedstawiona jest na diagramie.
@baseType <i>optional</i>	Pole techniczne: typ bazowy zgłoszenia. Pole opcjonalne.
@type <i>required</i>	Pole techniczne: informacja o typie zasobu wyprowadzonym z typu bazowego np. FaultTicket , MassFaultTicket .

relatedEntity <i>required</i>	Lista powiązanych ze zgłoszeniem zasobów. W przypadku zgłoszeń jest to instancja produktu dla którego zgłoszenie zostało utworzone.
statusChange <i>optional</i>	Historia zmian statusu zasobu w postaci listy stanów. Pole uzupełniane przez system OPL.
statusChangeReason <i>required</i>	Powód ostatniej zmiany statusu zasobu. Pole wypełniane przez OPL.
note <i>optional</i>	Opcjonalna lista notatek. Notatki posiadają typ który umożliwia automatyczne przetwarzanie notatek.
relatedParty <i>required</i>	Lista referencji do obiektów o typie Organization lub wartości obiektów Person. Obiekty reprezentują strony biorące udział w obsłudze zgłoszenia – zgłaszający, OPL, abonent/klient.
ticketRelationship <i>optional</i>	Lista powiązanych zgłoszeń jeśli jest konieczna.
channel <i>required</i>	Kanał komunikacji z operatorem. Wypełniane przez OPL.

Channel sub-resource

Referencja kanału kontaktu z operatorem.

id <i>required</i>	Unikalny dla OPL identyfikator kanału kontaktu utworzony i wypełniany przez OPL.
name <i>optional</i>	Nazwa kanału.
@type <i>required</i>	Pole techniczne określające obiekt reprezentujący kanał sprzedażowy.
@baseType <i>optional</i>	Pole techniczne określające obiekt bazowy obiektu reprezentującego kanał sprzedażowy.

Note sub-resource

Zasób przechowujący dodatkowe informacje.

date <i>required</i>	Wymagane. Data utworzenia notatki.
author <i>required</i>	Autor notatki.
text <i>required</i>	Treść notatki.

@type Pole techniczne określające typ zasobu reprezentującego notatki.
required

@baseType Pole techniczne określające typ bazowy zasobu reprezentującego notatki.
optional

StatusChange sub-resource

Przechowuje informacje o zmianie stanu zasobu. Pole wypełniane i zarządzane przez system ITS.

status Stan zasobu, tutaj zgłoszenia.
required

changeDate Data zmiany stanu zasobu.
required

changeReason Treść powodu zmiany stanu.
optional

@baseType Pole techniczne określające typ bazowy zasobu zmiany stanu zgłoszenia.
optional

@type Pole techniczne określające typ zasobu reprezentujący zmianę stanu.
required

TicketRelationship sub-resource

Obiekt reprezentuje relację między zgłoszeniem a innymi zgłoszeniami.

id Unikalny identyfikator zgłoszenia powiązanego.
required

type Typ relacji np. dependant.
optional

@type Pole techniczne określające typ zasobu reprezentującego relację między zgłoszeniami.
required

@baseType Pole techniczne określające typ bazowy zasobu.
optional

RelatedEntityRef relationship

Obiekt reprezentujący relację między zgłoszeniem a obiektami związanymi z danym typem zgłoszenia.

id Unikalny identyfikator obiektu do którego odnosi się referencja.
required

name <i>optional</i>	Nazwa zasobu będącego w relacji np. zamówienie.
role <i>required</i>	Rola w jakiej danych zasób w relacji występuje z enumeracji <code>relatedEntityRoleType</code> .
@referredType <i>required</i>	Pole techniczne określające typ zasobu do którego odnosi się referencja. Obecnie akceptowany jest typ <code>Product</code> lub jego typy pochodne.

Enumerations

RelatedEntityRoleType

Słownik typów relacji encji ze zgłoszeniem.

- **damagedService** – uszkodzona usługa,
- **reportedService** – usługa raportowana będąca przedmiotem reklamacji,
- **service** – usługa, przedmiot asysty.

RelatedPartyRef relationship

Obiekt reprezentujący relacje między stronami występującymi w procesie obsługi zdarzenia tj. operatora, OPL itp.

id <i>required/absent</i>	Unikalny identyfikator zasobu reprezentującego stronę. Pole jest wymagane w czasie przekazywania stanu zasobu w postaci referencji, niewymagane i nieobecne gdy przekazywana jest reprezentacja w postaci wartości.
role <i>required</i>	Rola w jakiej strona bierze udział w procesie zgłoszenia: owner (operator), technicalPartnerOA (partner techniczny operatora) itp.
name <i>optional</i>	Nazwa strony biorącej udział w procesowaniu zgłoszenia.
@referredType <i>required</i>	Identyfikator typu zasobu reprezentującego stronę, pole techniczne zawsze wymagane.

Person sub-resource

Zasób przechowujący dane klienta końcowego.

@type <i>required</i>	Pole techniczne określające typ reprezentujący zasób.
@baseType <i>optional</i>	Pole techniczne określające typ bazowy zasobu.

name <i>required</i>	Imię i nazwisko osoby.
role <i>required</i>	Rola w jakiej osoba występuje w danym kontekście: customer, technicalPartnerOA.
number <i>optional</i>	Numer telefonu kontaktowego.
emailAddress <i>optional</i>	Adres kontaktowy email

AssistTicket resource

AssistTicket jest obiektem dziedziczący pola klasy **TroubleTicket**.

Klasa **AssistTicket** została rozszerzona o następujące atrybuty:

appointmentTimeSlot <i>required</i>	Data i przedział czasu dotyczący umówienia służb technicznych (TimeSlot)
pduCheck <i>optional</i>	Sprawdzono usługi na PDU, boolean
netCheck <i>optional</i>	Sprawdzono infrastrukturę w lokalu abonenta, boolean
hardwareCheck <i>optional</i>	Sprawdzono sprzęt u abonenta, boolean

FaultTicket resource

FaultTicket jest obiektem dziedziczącym atrybuty i pola klasy **TroubleTicket** oraz zawierający dodatkowe pola konieczne do procesowania zgłoszenia błędu.

Klasa **TroubleTicket** została rozszerzona o następujące atrybuty:

claimCheck <i>optional</i>	Sprawdzono dostępność usługi do usługi na zasobach operatora, boolean
debtCheck <i>optional</i>	Sprawdzono blokadę windykacyjną, boolean

faultSymptom <i>required</i>	Lista symptomów błędu FaultSymptom.
accessTimeSlot <i>required</i>	Daty i godziny w jakich abonent jest dostępny w miejscu instalacji usługi. (TimeSlot).
appointmentDate <i>optional</i>	Daty i godziny w jakich zostało uzgodnione umówienie (TimeSlot).
lowerOptionAcceptance <i>optional</i>	Zgoda na obniżenie opcji prędkości: true , false .
pduCheck <i>optional</i>	Sprawdzono usługi na PDU, boolean
netCheck <i>optional</i>	Sprawdzono infrastrukturę w lokalu abonenta, boolean
hardwareCheck <i>optional</i>	Sprawdzono sprzęt u abonenta, boolean

ComplaintTicket resource

ComplaintTicket jest obiektem dziedziczącym atrybuty i pola klasy **TroubleTicket**. Obiekt dodaje następujące nowe atrybuty.

grantedAmount <i>optional</i>	Kwota uznanych roszczeń w PLN.
justification <i>required</i>	Krótki opis uzasadnienia reklamacji.
pduCheck <i>optional</i>	Sprawdzono usługi na PDU, boolean
netCheck <i>optional</i>	Sprawdzono infrastrukturę w lokalu abonenta, boolean
hardwareCheck <i>optional</i>	Sprawdzono sprzęt u abonenta, boolean

MassFaultTicket resource

MassFaultTicket jest obiektem dziedziczącym atrybuty i pola klasy **TroubleTicket** oraz zawierający dodatkowe pola konieczne do procesowania zgłoszenia awarii masowych.

Klasa **TroubleTicket** została rozszerzona o następujące atrybuty:

faultSymptom <i>required</i>	Lista symptomów błędu FaultSymptom, maksymalnie trzy symptomy.
creationDateOA <i>required</i>	Data rejestracji zgłoszenia w środowisku operatora alternatywnego.
pduId <i>optional</i>	Identyfikator pdu.
vlan2pdu <i>optional</i>	Identyfikator vlan w punkcie styku z PDU.

FaultSymptom sub-resource

symptom <i>required</i>	Wymagane. Identyfikator symptomu błędu wybrany ze słownika SYMPTOMS.
description <i>optional</i>	Dodatkowy opis symptomu wprowadzony przez operatora.
@type <i>required</i>	Pole techniczne określające typ obiektu reprezentującego Fault Symptom.
@baseType <i>optional</i>	Pole techniczne określające typ bazowy obiektu reprezentujący zasób Fault Symptom.

AdditionalState sub-resource

@type <i>required</i>	Pole techniczne opisujące typ obiektu: Rejection – kody odrzucenia, Exit -, kody Wyjścia, Changed kody związane ze zmianą danych w zasobie, Cancel - anulowania.
@baseType <i>optional</i>	Pole techniczne opisujące typ bazowy obiektu reprezentującego zasób.
code <i>required</i>	Kod opisujący przyczyny odrzuceń, kody wyjścia oraz inne kody stanu zasobu które nie są ujęte w stanach opisanych maszyną stanu TMF a wymagane w MWDP. Kody są pobierane z odpowiednich słowników w zależności od typu AdditionalState: Rejection, Changed: RR-FIA Exit: EC-FIA Cancel: CANCEL-REASON-CODE
description <i>required</i>	Dodatkowy opis stanu zgodny z odpowiednim słownikiem.

TimeSlot sub-resource

@type <i>required</i>	Pole techniczne typ reprezentujący zasób slotu czasowego.
@baseType <i>optional</i>	Pole techniczne opisujące typ bazowy obiektu reprezentującego zasób slotu.
startDateTime <i>required</i>	Data początkowa przedziału czasu.
endDateTime <i>required</i>	Data końcowa przedziału czasu.

Przykład reprezentacji zasobu w JSON

Poniżej znajduje się przykład zapisanego w postaci JSON obiektu zgłoszenia (asysty).

```
{
  "id": "3180",
  "name": "Zgłoszenie asysty TM",
  "externalId": "21308090",
  "ticketType": "technicalAssist",
  "creationDate": "2018-05-01T00:00",
  "lastUpdate": "2018-05-01T00:00",
  "description": "Wymagana asysta techniczna w miejscu",
  "severity": "minor",
  "priority": "low",
  "requestedResolutionDate": "2018-05-10T00:00",
  "expectedResolutionDate": "2018-05-10T00:00",
  "status": "inprogress",
  "statusChangeReason": "Zgłoszenie procesowane",
  "@type": "AssistTicket",
  "@baseType": "TroubleTicket",
  "appointmentTimeSlot": {
    "@type": "TimeSlot",
    "startDateTime": "2018-05-01T08:00",
    "endDateTime": "2018-05-01T10:00"
  },
  "relatedEntity": [
    {
      "id": "1234567890",
      "role": "service",
      "@referredType": "Product"
    }
  ],
  "statusChange": [
    {
      "@type": "StatusChange",
      "status": "acknowledged",
      "changeDate": "2018-05-01T00:00:00"
    }
  ],
  "note": [
    {
      "@type": "Note",
      "date": "2018-05-01T00:00:00",
      "author": "Janusz Nowak",
      "text": "Proszę o szybka interwencję w tej sprawie"
    }
  ]
}
```

```

    }
  ],
  "relatedParty": [
    {
      "id": "4",
      "name": "T-Mobile",
      "role": "owner",
      "@referredType": "Organization"
    },
    {
      "@type": "Person",
      "name": "Jan Kowalski",
      "role": "customer",
      "number": "+48506078978"
    }
  ],
  "channel": {
    "id": "WEB",
    "name": "Kanał webowy",
    "type": "Channel"
  }
}

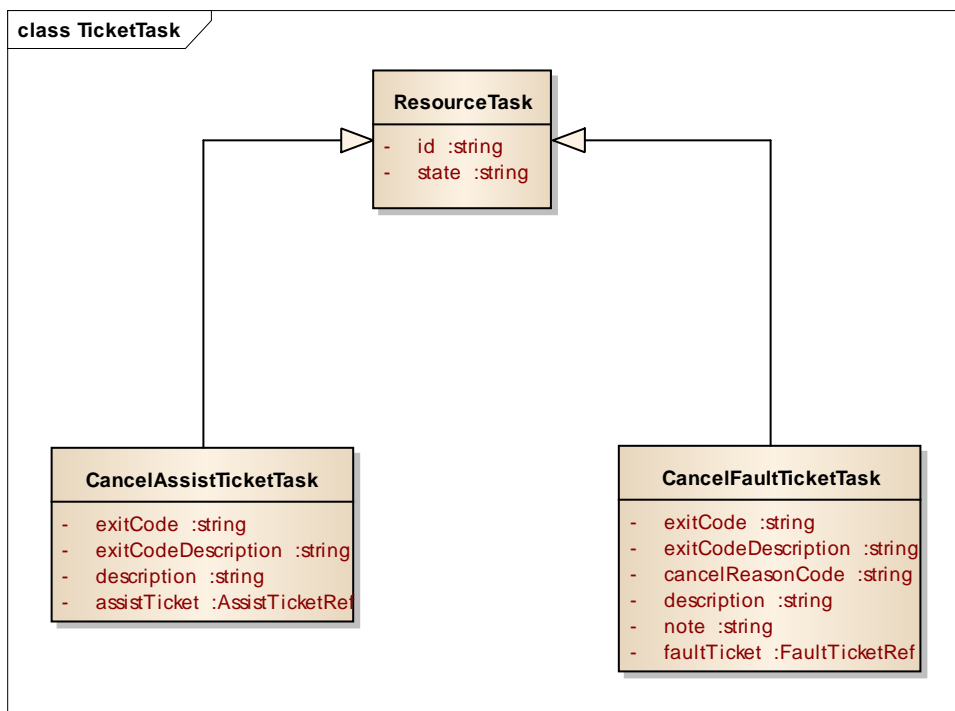
```

CancelAssistTicketTask, CancelFaultTicketTask resources

Zasób zadania anulowania zgłoszenia asysty jest używany w jednym celu – anulowania zgłoszenia. Jego utworzenie spowoduje anulowanie asynchroniczne procesowanej, istniejącej asysty do której zadanie się odwołuje.

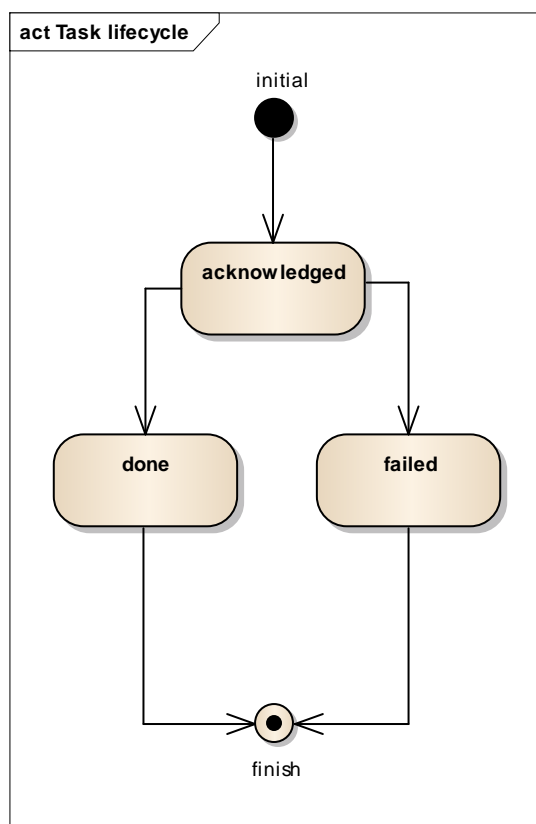
Zasób zadania anulowania uszkodzenia podobnie jak dotyczący asysty, służy jedynie do anulowania zgłoszenia uszkodzenia. Jego utworzenie spowoduje anulowanie asynchroniczne procesowanego, uszkodzenia do którego zadanie się odwołuje.

Resource model



Life cycle

Zasób zadania anulowania zgłoszenia asysty oraz anulowania uszkodzenia posiadają prosty cykl życia przedstawiony na poniższym diagramie:



Opis pól zasobu zadania anulowania asysty technicznej

id <i>required</i>	Identyfikator zadania anulowania zgłoszenia asysty tworzony przez OPL w czasie tworzenia zasobu.
state <i>required</i>	Aktualny stan zadania anulowania asysty. Stan zarządzany przez OPL.
exitCode <i>optional</i>	Kod zakończenia zadania zgodny ze słownikiem FAULT-CANCEL-REASON. Pole wypełniane przez OPL.
exitCodeDescription <i>optional</i>	Opis kodu zakończenia zadania zgodny ze słownikiem FAULT-CANCEL-REASON. Pole wypełniane przez OPL.
description <i>required</i>	Opis swobodny powodu anulowania asysty.
assistTicket <i>required</i>	Referencja do obiektu będącego przedmiotem zadania – zgłoszenia asysty.
@type <i>required</i>	Pole techniczne określające typ reprezentujący zasób zadania.
@baseType <i>optional</i>	Pole techniczne określające typ bazowy zasobu zadania.

Przykład utworzenia zadania anulowania asysty

```
POST
https://host:port/tmf/troubleTicketManagement/v1/cancelAssistTicketTask
Content-Type: application/json; charset=UTF-8

{
  "@type": "CancelAssistTicketTask",
  "description": "Opis powodu anulowania", #dodatkowy opis
  "assistTicket": {
    "id": "123456",
    "@referredType": "AssistTicket"
  }
}
```

Przykład reprezentacji zadania zakończonego ze stanem failed (reprezentacja przekazywana notyfikacją).

```
{
  "id": "1234567890",
  "@type": "CancelAssistTicketTask",
  "state": "failed",
  "exitCode": "2",
  "exitCodeDescription": "Anulowanie zgłoszenia nie jest możliwe na tym etapie",
  "description": "Opis powodu anulowania", #dodatkowy opis
  "assistTicket": {
    "id": "123456",
```

```

    "@referredType": "AssistTicket"
  }
}

```

Opis pól zasobu zadania anulowania zgłoszenia uszkodzenia

id <i>required</i>	Identyfikator zadania anulowania zgłoszenia uszkodzenia tworzony przez OPL w czasie tworzenia zasobu.
state <i>required</i>	Aktualny stan zadania anulowania uszkodzenia. Stan zasobu jest zarządzany przez OPL.
exitCode <i>optional</i>	Kod zakończenia zadania zgodny ze słownikiem ze słownikiem FAULT-CANCEL-REASON. Pole wypełniane przez OPL.
exitCodeDescription <i>optional</i>	Opis kodu zakończenia zadania zgodny ze słownikiem ze słownikiem FAULT-CANCEL-REASON. Pole wypełniane przez OPL.
cancelReasonCode <i>required</i>	Kod powodu anulowania zgłoszenia ze słownika CANCEL-REASON-CODE.
description <i>required</i>	Swobodny opis powodu anulowania uszkodzenia.
note <i>optional</i>	Dodatkowy opis uszkodzenia
faultTicket <i>required</i>	Referencja do obiektu będącego przedmiotem zadania – zgłoszenia uszkodzenia.
@type <i>required</i>	Pole techniczne określające typ reprezentujący zasób zadania.
@baseType <i>optional</i>	Pole techniczne określające typ bazowy zasobu zadania.

POST https://host:port/tmf/troubleTicketManagement/v1/cancelFaultTicketTask
Content-Type: application/json; charset=UTF-8

```

{
  "@type": "CancelFaultTicketTask",
  "cancelReasonCode": "15002", #słownik CANCEL-REASON-CODE
  "description": "Opis powodu anulowania",
  "note": "Dodatkowy opis",
  "faultTicket": {
    "id": "123456",
    "@referredType": "FaultTicket"
  }
}

```

ErrorRepresentation

Struktura reprezentująca zaistniały w czasie przetwarzania błąd.

Name	Description	Schema
code <i>required</i>	Kod błędu zgodny z kodami błędów w tabelce dla poszczególnych kodów HTTP.	Integer (int32)
description <i>optional</i>	Opis błędu w formie czytelnej dla użytkownika komputera.	String
details <i>optional</i>	Opis szczegółowy w formie czytelnej dla użytkownika komputera.	<ZErrorDetail> array
message <i>optional</i>	Szczegółowy opis błędu który można zaprezentować użytkownikowi na ekranie komputera.	String
reason <i>required</i>	Krótki opis błędu zgodny z opisem błędów wymienionych w tablicy. Tekst ten można zaprezentować użytkownikowi na ekranie komputera.	String
status <i>optional</i>	Stan systemu po wystąpieniu błędu. Stan zasobu po wystąpieniu błędu.	String

ZErrorDetail

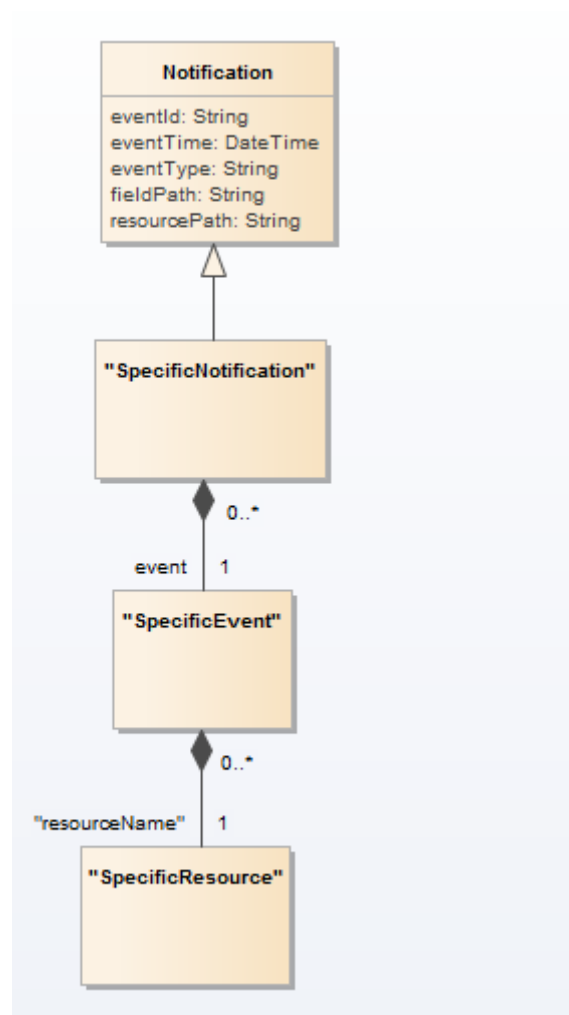
Name	Schema
code <i>optional</i>	integer (int32)
description <i>optional</i>	string
message <i>required</i>	string

Notification Resource Models

OPL będzie wysyłało różne notyfikacje związane ze zmianą stanu zasobów API:

- **TroubleTicketChangeNotification** – informacja o zmianie danych zasobu zgłoszenia,
- **TroubleTicketStatusChangeNotification** – informacja o zmianie stanu zasobu zgłoszenia,
- **TroubleTicketInformationRequiredNotification** – informacja o konieczności uzupełnienia danych zasobu zgłoszenia,
- **TroubleTicketResolvedNotification** – informacja o poprawnym zakończeniu realizacji zgłoszenia
- **CancelAssistTicketTaskStateChangeNotification** – informacja o zmianie stanu zadania anulowania asysty technicznej,
- **CancelFaultTicketTaskStateChangeNotification** – informacja o zmianie stanu zadania anulowania zgłoszenia uszkodzenia.

Poniżej znajduje się model danych zasobu notyfikacji.



Trouble Ticket Change Notification

Notyfikacja wysyłana w chwili zmiany danych zgłoszenia.

```
{
  "eventId": "00001",
  "eventTime": "2015-11-16T16:42:25-04:00",
  "eventType": "TroubleTicketChangeNotification",
  "event": {
    "troubleTicket": {
      "@type": "FaultTicket",
      "id": "3180",
      -- Tutaj znajdują się te atrybuty zgłoszenia które uległy
zmianie
    }
  }
}
```

Trouble Ticket Status Change Notification

Notyfikacja wysyłana w sytuacji zmiany stanu zgłoszenia.

```
{
  "eventId": "00001",
  "eventTime": "2015-11-16T16:42:25-04:00",
  "eventType": "TroubleTicketStatusChangeNotification",
  "event": {
    "troubleTicket": {
      "@type": "FaultTicket",
      -- Tutaj znajduje się kompletna reprezentacja zasobu
zgłoszenia
    }
  }
}
```

Trouble Ticket Information Required Notification

Notyfikacja informującą o konieczności uzupełnienia zasobu, kierowana do podmiotu który utworzył zgłoszenie.

- **"resourcePath"** - określa który zasób o identyfikatorze URL powinien być uzupełniony.
- **"fieldPath"** – określa które pole zasobu, atrybut wymaga poprawy, uzupełnienia lub odpowiedzi:
 - o **"accept"**= wskazuje pole jakie należy zaakceptować,
 - o **"quest"**= wskazuje pole w jakim znajduje się zapytanie na które należy udzielić odpowiedzi,
 - o **"escalation"**= wskazuje pole w jakim znajduje się zapytanie na które należy udzielić odpowiedzi w ramach eskalacji.

```
{
  "eventId": "00001",
  "eventTime": "2015-11-16T16:42:25-04:00",
  "eventType": "TroubleTicketInformationRequiredNotification",
  "resourcePath": "TroubleTicketManagement/v1/TroubleTicket/3248",
  "fieldPath": "quest=note[5]" //answer, missing
  "event": {
    "troubleTicket": {
```

```

        "@type": "FaultTicket",
        "status": "pending",
        -- Tutaj znajduje się pozostała reprezentacja zasobu zgłoszenia
    }
}

```

Trouble Ticket Resolved Notification

Notyfikacja wysyłana w sytuacji poprawnym zakończeniu realizacji zgłoszenia.

```

{
  "eventId": "00001",
  "eventTime": "2015-11-16T16:42:25-04:00",
  "eventType": "TroubleTicketResolvedNotification",
  "event": {
    "troubleTicket": {
      "@type": "FaultTicket",
      -- Tutaj znajduje się kompletna reprezentacja zasobu zgłoszenia
    }
  }
}

```

Cancel Assist Ticket Task State Change Notification

Notyfikacja wysyłana w po zmianie statusu zadania anulowania asysty.

```

{
  "eventId": "00001",
  "eventTime": "2015-11-16T16:42:25-04:00",
  "eventType": "CancelAssistTicketTaskStateChangeNotification",
  "event": {
    "cancelAssistTicketTask": {
      "@type": "CancelAssistTicketTask",
      "exitCode": "1", #kod ze słownika
      "exitCodeDescription": "opis ze słownika FAULT-CANCEL-
REASON",
      "state": "done",
      -- Tutaj znajduje się pozostała reprezentacja zasobu zadania
    }
  }
}

```

Cancel Fault Ticket Task State Change Notification

Notyfikacja wysyłana w po zmianie statusu zadania anulowania uszkodzenia.

```

{
  "eventId": "00001",
  "eventTime": "2015-11-16T16:42:25-04:00",
  "eventType": "CancelFaultTicketTaskStateChangeNotification",
  "event": {

```

```
        "cancelFaultTicketTask ": {
            "@type": "CancelFaultTicketTask",
            "exitCode": "1", #kod ze słownika
            "exitCodeDescription": "opis ze słownika FAULT-CANCEL-
REASON",
            "state": "done",
            -- Tutaj znajduje się pozostała reprezentacja zasobu zadania
        }
    }
}
```

API OPERATIONS

Zgodnie ze specyfikacją TMF API REST usługa OPL spełnia następujący kontrakt:

Operacja na encjach	Operacja API	Opis
Tworzenie zgłoszeń	POST Resource	POST tworzy kompletny obiekt zgłoszenia
Aktualizacja części zgłoszenia	PATCH Resource	PATCH aktualizuje częściowo obiekt zgłoszenia wykonując JSON merge. Merge jest wykonywany przed walidacją atrybutów i weryfikacją spójności

Operations on Trouble Ticket

Create Trouble Ticket

POST /troubleTicket

Opis

Ta operacja tworzy zgłoszenie. Oprócz reprezentacji zasobu metoda zwraca również Etag jednoznacznie identyfikujący stan zasobu (hash).

Wymagane i niewymagane atrybuty zgłoszenia

Wymagane atrybuty	Reguła
description	Krótki opis zgłoszenia.
ticketType	complaint, fault, massFault, technicalAssist
@type	Pole techniczne określające typ zgłoszenia.

Pozostałe pola wymagane i niewymagane zostały opisane w rozdziale procesowanie zgłoszeń dla każdego typu zgłoszenia oddzielnie. Poniżej zamieszczone są tylko przykłady.

Przykład:

W tym przykładzie jest wysyłane są tylko wymagane atrybuty zgłoszenia.

Request

```
POST https://host:port/tmf/troubleTicketManagement/v1/troubleTicket
Content-Type: application/json; charset=UTF-8

{
  "description": "Reklamacja wykonanych prac",
  "ticketType": "complaint",
  "@type": "ComplaintTicket"
}
```

Response

System przekazuje informację o częściowym utworzeniu zasobu HTTP 202 (partial create) wraz następującymi polami.

id	Unikalny identyfikator zasobu utworzonego przez OPL.
status	Stan po utworzeniu zasobu - acknowledged

Reszta atrybutów zasobu zostanie przekazana w notyfikacji informującej o zmianie stanu zasobu.

```
202
Etag: e6c19694e0a94702c50ea93d0455ec83

{
  "id": "3180",
  "status": "acknowledged",
  "@type": "ComplaintTicket"
}
```

Podsumowanie wyników działania operacji:

HTTP Code	Description	Schema
202	Accepted Zaakceptowano żądanie utworzenia zasobu	TroubleTicket
400	Bad Request Błędna postać żądania - 22: Nieprawidłowa postać komunikatu HTTP - 23: Brak wymaganego pola zasobu - 24: Nieprawidłowa wartość pola zasobu	ErrorRepresentation

HTTP Code	Description	Schema
401	- 25: Brak nagłówka HTTP	ErrorRepresentation
	- 26: Nieprawidłowa wartość w nagłówku HTTP	
	Unauthorized	
	Brak autoryzacji	
	- 40: Brak informacji autentykacyjnej	
	- 41: Nieprawidłowa autentykacja	
403	- 42: Uprawnienia wygasły	ErrorRepresentation
	Forbidden	
	Dostęp zabroniony	
404	- 50: Dostęp zabroniony	ErrorRepresentation
	Not Found	
	Nie znaleziono zasobu	
405	- 60: Nie znaleziono zasobu	ErrorRepresentation
	Method not allowed	
	Metoda niedozwolona	
406	- 61: Metoda niedozwolona	ErrorRepresentation
	Not acceptable	
	Nie można zrealizować żądania	
422	- 62: Nie można zrealizować żądania	ErrorRepresentation
	Unprocessable entity	

HTTP Code	Description	Schema
	Błąd funkcjonalny - 1 Błąd funkcjonalny	
500	Internal Server Error Błąd wewnętrzny - 1: Błąd wewnętrzny	ErrorRepresentation

Patch Trouble Ticket

PATCH /troubleTicket/{id}

Opis

Operacja PATCH pozwala na częściową aktualizację zgłoszenia. Aktualizacja zasobu powinna być realizowana z wykorzystaniem reguł i zasad **json/merge** (<https://tools.ietf.org/html/rfc7386>) oraz **json/patch** <http://tools.ietf.org/html/rfc5789>. Metoda wymaga wskazania reguły aktualizacji w nagłówku **If-Match** z wartością wersji zwróconym w czasie tworzenia zasobu. Metoda zwraca reprezentację zasobu po aktualizacji wraz z nagłówkiem **Etag** określającym aktualną wersję zasobu (wyznaczany funkcja skrótu).

Patchable and Non Patchable Attributes

W rozdziale procesowanie zgłoszeń w poszczególnych punktach obsługi danego zgłoszenia opisano wymagane i opcjonalne dla operacji PATCH dane.

Poniżej dla przypomnienia lista atrybutów które nigdy nie mogą zostać zmienione

Id

creationDate

lastUpdate

statusChange

@baseType

@type

Przykład użycia**Request**

```
PATCH /tmf/troubleTicketManagement/v1/troubleTicket/3180
Content-Type: application/merge-patch+json; charset=UTF-8
If-Match: e6c19694e0a94702c50ea93d0455ec83

{
  "description": "Nowy opis, zmieniony",
  "@type": "FaultTicket"
}
```

Response

```
200
Etag: e6c19694e0a94702c50ea93d0455ec83

{
  "id": "3180",
  -- Tutaj znajdują się pozostałe atrybuty zasobu zgłoszenia--
}
```

W wyniku prawidłowej realizacji operacji PATCH (HTTP200) zawsze zwracana jest pełna zawartość zasobu.

Podsumowanie wyników działania operacji:

HTTP Code	Description	Schema
200	Ok Sukces	TroubleTicket
400	Bad Request Błędna postać żądania - 22: Nieprawidłowa postać komunikatu HTTP - 23: Brak wymaganego pola zasobu - 24: Nieprawidłowa wartość pola zasobu - 25: Brak nagłówka HTTP - 26: Nieprawidłowa wartość w nagłówku HTTP	ErrorRepresentation

HTTP Code	Description	Schema
401	Unauthorized Brak autoryzacji - 40: Brak informacji autentykacyjnej - 41: Nieprawidłowa autentykacja - 42: Uprawnienia wygasły	ErrorRepresentation
403	Forbidden Dostęp zabroniony - 50: Dostęp zabroniony	ErrorRepresentation
404	Not Found Nie znaleziono zasobu - 60: Nie znaleziono zasobu	ErrorRepresentation
405	Method not allowed Metoda niedozwolona - 61: Metoda niedozwolona	ErrorRepresentation
406	Not acceptable Nie można zrealizować żądania - 62: Nie można zrealizować żądania	ErrorRepresentation
412	Precondition failed Żądanie nie spełnia warunków jego wykonania Nie spełniony warunek wykonania operacji PATCH. Nagłówek If-Match żądania zawiera nieprawidłową wartość Etag niezgodną z wartością na serwerze OPL.	TroubleTicket

HTTP Code	Description	Schema
422	Unprocessable entity Błąd funkcjonalny - 1: Błąd funkcjonalny	ErrorRepresentation
500	Internal Server Error Błąd wewnętrzny - 1: Błąd wewnętrzny	ErrorRepresentation

Operations on CancelAssistTicketTask, CancelFaultTicketTask

Create CancelAssistTicketTask

POST /cancelAssistTicketTask

Opis

Ta operacja tworzy zadanie anulowania asysty.

Wymagane i niewymagane atrybuty zadania

Wymagane atrybuty	Reguła
description	Krótki opis powodu anulowania asysty.
@type	Pole techniczne określające typ zadania.
assistTicket	Referencja do obiektu zgłoszenia asysty.

Przykład:

```
POST
https://host:port/tmf/troubleTicketManagement/v1/cancelAssistTicketTask
Content-Type: application/json; charset=UTF-8

{
  "@type": "CancelAssistTicketTask",
  "description": "Opis powodu anulowania", #dodatkowy opis
  "assistTicket": {
    "id": "123456",
    "@referredType": "AssistTicket"
  }
}
```

```
}
}
```

Podsumowanie wyników działania operacji:

HTTP Code	Description	Schema
202	Accepted Zaakceptowano żądanie utworzenia zasobu	CancelAssistTicketTask
400	Bad Request Błędna postać żądania - 22: Nieprawidłowa postać komunikatu HTTP - 23: Brak wymaganego pola zasobu - 24: Nieprawidłowa wartość pola zasobu - 25: Brak nagłówka HTTP - 26: Nieprawidłowa wartość w nagłówku HTTP	ErrorRepresentation
401	Unauthorized Brak autoryzacji - 40: Brak informacji autentykacyjnej - 41: Nieprawidłowa autentykacja - 42: Uprawnienia wygasły	ErrorRepresentation
403	Forbidden Dostęp zabroniony - 50: Dostęp zabroniony	ErrorRepresentation
404	Not Found	ErrorRepresentation

HTTP Code	Description	Schema
405	Nie znaleziono zasobu	ErrorRepresentation
	- 60: Nie znaleziono zasobu	
405	Method not allowed	ErrorRepresentation
	Metoda niedozwolona	
405	- 61: Metoda niedozwolona	
406	Not acceptable	ErrorRepresentation
	Nie można zrealizować żądania	
406	- 62: Nie można zrealizować żądania	
422	Unprocessable entity	ErrorRepresentation
	Błąd funkcjonalny	
422	- 1: Błąd funkcjonalny	
500	Internal Server Error	ErrorRepresentation
	Błąd wewnętrzny	
500	- 1: Błąd wewnętrzny	

Create CancelFaultTicketTask

POST /cancelFaultTicketTask

Opis

Ta operacja tworzy zadanie anulowania zgłoszenia uszkodzenia.

Wymagane i niewymagane atrybuty zadania

Wymagane atrybuty	Reguła
description	Krótki opis zgłoszenia.

cancelReasonCode	Kod powodu anulowania zgłoszenia ze słownika CANCEL-REASON-CODE.
@type	Pole techniczne określające typ zadania anulowania zgłoszenia.
faultTicket	Referencja do zasobu uszkodzenia.

Przykład:

```
POST https://host:port/tmf/troubleTicketManagement/v1/cancelFaultTicketTask
Content-Type: application/json; charset=UTF-8
```

```
{
  "@type": "CancelFaultTicketTask",
  "cancelReasonCode": "15001", #słownik CANCEL-REASON-CODE
  "description": "Opis powodu anulowania", # swobodny opis
  "note": "Dodatkowy opis",
  "faultTicket": {
    "id": "123456",
    "@referredType": "FaultTicket"
  }
}
```

Podsumowanie wyników działania operacji:

HTTP Code	Description	Schema
202	Accepted Zaakceptowano żądanie utworzenia zasobu	CancelFaultTicketTask
400	Bad Request Błędna postać żądania - 22: Nieprawidłowa postać komunikatu HTTP - 23: Brak wymaganego pola zasobu - 24: Nieprawidłowa wartość pola zasobu - 25: Brak nagłówka HTTP - 26: Nieprawidłowa wartość w nagłówku HTTP	ErrorRepresentation

HTTP Code	Description	Schema
401	<p>Unauthorized</p> <p>Brak autoryzacji</p> <p>- 40: Brak informacji autentykacyjnej</p> <p>- 41: Nieprawidłowa autentykacja</p> <p>- 42: Uprawnienia wygasły</p>	ErrorRepresentation
403	<p>Forbidden</p> <p>Dostęp zabroniony</p> <p>- 50: Dostęp zabroniony</p>	ErrorRepresentation
404	<p>Not Found</p> <p>Nie znaleziono zasobu</p> <p>- 60: Nie znaleziono zasobu</p>	ErrorRepresentation
405	<p>Method not allowed</p> <p>Metoda niedozwolona</p> <p>- 61: Metoda niedozwolona</p>	ErrorRepresentation
406	<p>Not acceptable</p> <p>Nie można zrealizować żądania</p> <p>- 62: Nie można zrealizować żądania</p>	ErrorRepresentation
422	<p>Unprocessable entity</p> <p>Błąd funkcjonalny</p> <p>- 1: Błąd funkcjonalny</p>	ErrorRepresentation

HTTP Code	Description	Schema
500	<p>Internal Server Error</p> <p>Błąd wewnętrzny</p> <p>- 1: Błąd wewnętrzny</p>	ErrorRepresentation

API NOTIFICATIONS

OPL nie dostarcza operatorom huba realizującego notyfikacje co oznacza że nie jest możliwe wykonanie subskrypcji tylko na pewien zakres powiadomień wysyłanych przez OPL. Jakie powiadomienia są przesyłane przez OPL operatorowi i na jakich etapach procesu opisane jest w rozdziale – procesowanie zgłoszeń.

PROCESOWANIE ZGŁOSZEŃ

Asysta, awaria, uszkodzenie oraz reklamacja są obsługiwane przy użyciu tego samego zasobu - zgłoszenia. Pole o nazwie **ticketType** informuje z jakim typem zgłoszenia mamy do czynienia. Cykl życia każdego typu zgłoszenia jest taki sam jedyną różnicą jest inny zestaw notyfikacji wysyłanych w trakcie przetwarzania zgłoszenia i jest on zgodny z procesem opisanym w dokumencie MWDP.

Uszkodzenia

Poniższa tabelka obrazuje powiązania między stanami zasobu zgłoszenia a procesem obsługi zgłoszenia uszkodzenia. Wytłuszczenie wskazuje na stany końcowe (terminalne).

Etap	Stan procesu zgłoszenia	Stan zasobu zgłoszenia	Informacja zwracana OA
1	Przyjęcie komunikatu pozytywnie zweryfikowanego o informatycznie	acknowledged	HTTP202 (partial create)
1a	Negatywna weryfikacja informatyczna.	Zasób zgłoszenia nie jest tworzony nie posiada stanu	HTTP4XX ErrorRepresentation
1b	Rozpoczęto weryfikację zgłoszenia	captured	Notyfikacja TroubleTicketStatusChangeNotification
2	Pozytywny wynik weryfikacji formalnej.	inprogress	Notyfikacja TroubleTicketStatusChangeNotification
2a	Negatywna weryfikacja formalna.	rejected	Notyfikacja TroubleTicketStatusChangeNotification
3	Zmiana terminu umówienia przez OPL.	inprogress	Notyfikacja TroubleTicketChangeNotification
4	Anulowanie zgłoszenia ze strony OA.	cancelled	OA tworzy zadanie anulowania uszkodzenia - CancelFaultTicketTask . Po zakończeniu zadania przekazywana jest notyfikacja CancelFaultTicketStateChangeNotification informująca o zakończeniu zadania oraz notyfikacja TroubleTicketStatusChangeNotification o zmianie stanu zasobu uszkodzenia. W przypadku błędów związanych z utworzeniem zadania przekazywany jest ErrorRepresentation
5	Realizacja skuteczna.	resolved	Notyfikacja TroubleTicketResolvedNotification
6	Potwierdzenie skutecznej realizacji przez OA.	closed	HTTP (200) successfull patch lub HTTP (400) z ErrorRepresentation
6a	Upłynął czas na odpowiedź OA	closed	Notyfikacja TroubleTicketStatusChangeNotification

	(brak odpowiedzi).		
6b	Odrzucenie rozwiązania przez OA, dalsze procesowanie od punktu 3.	inprogress	HTTP (200) successfull patch lub HTTP (400) z ErrorRepresentation
6c	Zakończenie realizacji zgłoszenia	closed	Notyfikacja TroubleTicketStatusChangeNotification
7	Podpięcie zgłoszenia pod Awarię Masową/Pracę Planową.	held	Notyfikacja TroubleTicketStatusChangeNotification
7a	Odlączenie zgłoszenia od Awarii Masowej/Pracy Planowej.	inprogress	Notyfikacja TroubleTicketStatusChangeNotification

1. Utworzenie przez operatora zgłoszenia o uszkodzeniu.

Poniższa tabelka zawiera listę atrybutów zgłoszenia uszkodzenia tworzonych w trakcie wykonania operacji na zasobie. Część atrybutów określa operator, pozostałe OPL. Należy pamiętać o polach wymaganych wynikających z modelu danych oraz wynikających z zastosowania metody.

Atrybuty niewymagane których operator nie wypełnia i nie są zamieszczone w poniższej tabelce nie powinny być do OPL przesyłane. Pola nieużywane i niewymagane nie są opisane w tabelce.

Atrybuty	Wymagane	Wartość domyślna	Reguła
externalId	Nie		Identyfikator własny operatora, pole nie jest przetwarzane przez OPL
ticketType	Tak		fault
description	Tak		Opis zgłoszenia.
name	Nie		Nazwa zgłoszenia
creationDate			Wypełnianie przez OPL
lastUpdate			Wypełnianie przez OPL
priority			Wypełniane przez OPL

Atrybuty	Wymagane	Wartość domyślna	Reguła
severity	Tak		Wpływ na proces biznesowy: minor, major, critical .
requestedResolutionDate			Wypełniane przez OPL
expectedResolutionDate			Wypełniane przez OPL
resolutionDate			Wypełnianie przez OPL
status			Wypełnianie przez OPL
@type	Tak	FaultTicket	Typ obiektu, w przypadku uszkodzenia FaultTicket
@baseType	Nie	TroubleTicket	TroubleTicket
relatedEntity	Tak		Pole zawiera listę referencji instancji produktów których dotyczy zgłoszenie (błędnie działających). Lista musi zawierać referencje tylko instancji produktów ACCESS i BISTREAML2 w roli " damagedService ". Wyspecyfikowanie instancji produktów nie należących do danego operatora spowoduje błąd walidacji informatycznej związany z nieprawidłową zawartością pola.
statusChange			Wypełnianie przez OPL, lista zmian statusów zasobu
note	nie		
relatedParty	Tak		Pole zawiera listę zawierającą referencję do obiektu Organization reprezentującego operatora alternatywnego ("role": " owner ") oraz obiekt Person przechowujący informacje o abonencie. Obiekt reprezentujący abonenta musi posiadać wypełnione pole number . Wyspecyfikowanie obiektów w innej roli lub brak wyżej wymienionych spowoduje błąd w trakcie walidacji informatycznej związany z nieprawidłową zawartością pola.

Atrybuty	Wymagane	Wartość domyślna	Reguła					
channel			Wypełniane przez OPL					
pduCheck	Nie	false	sprawdzono usługi na PDU: true, false					
claimCheck	Nie	false	sprawdzono dostępność usługi do usługi na zasobach operatora: true, false					
netCheck	Nie	false	Sprawdzono infrastrukturę w lokalu abonenta: true, false					
hardwareCheck	Nie	false	Sprawdzono sprzęt u abonenta: true, false					
debtCheck	Nie	false	Sprawdzono blokadę windykacyjną: true, false					
accessTimeSlot	Tak		<p>Okres dostępności abonenta okres dostępności powinien być podany w trzygodzinnych slotach według schematu:</p> <table><tr><td>Pn. – Pt. od 9:00 do 12:00</td></tr><tr><td>Pn. – Pt. od 12:00 do 15:00</td></tr><tr><td>Pn. – Pt. od 15:00 do 18:00</td></tr><tr><td>Sobota od 9:00 do 12:00</td></tr><tr><td>Sobota od 12:00 do 15:00</td></tr></table> <p>Slot jest określany datami od do tj:</p> <pre>"accessTimeSlot": { "@type": "TimeSlot", "startDateTime": "2018-09-10T09:00:00Z", "endDateTime": "2018-09-10T12:00:00Z" }</pre>	Pn. – Pt. od 9:00 do 12:00	Pn. – Pt. od 12:00 do 15:00	Pn. – Pt. od 15:00 do 18:00	Sobota od 9:00 do 12:00	Sobota od 12:00 do 15:00
Pn. – Pt. od 9:00 do 12:00								
Pn. – Pt. od 12:00 do 15:00								
Pn. – Pt. od 15:00 do 18:00								
Sobota od 9:00 do 12:00								
Sobota od 12:00 do 15:00								
faultSymptom	Tak		List obiektów FaultSymptom, maksymalnie można podać do 3 symptomów.					
appointmentDate	Nie		Wypełniane przez OPL.					

Atrybuty	Wymagane	Wartość domyślna	Reguła
			<p>Daty i godziny w jakich zostało uzgodnione umówienie (TimeSlot):</p> <pre>"appointmentDate": { "@type": "TimeSlot", "startDateTime": "2018-09-10T09:00:00Z", "endDateTime": "2018-09-10T12:00:00Z" }</pre>

Błędne wartości pól powodują zgłoszenie błędu w czasie weryfikacji informatycznej. OPL zwróci zawsze poprawną reprezentację zgłoszenia zapewniając spójny obraz zasobu w OPL i u operatora.

Przykład utworzenia zgłoszenia uszkodzenia (JSON):

POST https://host:port/tmf/troubleTicketManagement/v1/troubleTicket
Content-Type: application/json; charset=UTF-8

```
{
  "name": "Zgłoszenie uszkodzenia TM",
  "externalId": "21308090",
  "description": "Usługa nie działa",
  "ticketType": "fault",
  "severity": "critical",
  "@type": "FaultTicket",
  "faultSymptom": [
    {
      "@type": "FaultSymptom",
      "symptom": "11001",
      "description": "opis symptomu nr 1"
    },
    {
      "@type": "FaultSymptom",
      "symptom": "11002",
      "description": "opis symptomu nr 2"
    },
    {
      "@type": "FaultSymptom",
      "symptom": "11003",
      "description": "opis symptomu nr 3"
    }
  ],
  "accessTimeSlot": {
    "@type": "TimeSlot",
    "startDateTime": "2018-06-02T09:00:00.000Z",
    "endDateTime": "2018-06-02T12:00:00.000Z"
  },
  "pduCheck": true,
  "claimCheck": true,
}
```



```

    "netCheck": false,
    "hardwareCheck": true,
    "debtCheck": false,
    "relatedEntity": [
      {
        "id": "1234567890",
        "role": "damagedService",
        "@referredType": "Product"
      },
      {
        "id": "1234567890",
        "role": "damagedService",
        "@referredType": "Product"
      }
    ],
    "note": [{
      "@type": "Note",
      "date": "2018-05-01T00:00:00",
      "author": "Janusz Nowak",
      "text": "Proszę o szybka interwencję w tej sprawie"
    }],
    "relatedParty": [
      {
        "id": "4",
        "name": "TM-Polska",
        "role": "owner",
        "@referredType": "Organization"
      },
      {
        "@type": "Person",
        "name": "Jan Kowalski",
        "role": "customer",
        "number": "+436641234567"
      }
    ]
  }
}

```

Co zwraca system OPL opisano w rozdziale opisującym operacje na zasobie.

1b. Rozpoczęto weryfikację uszkodzenia.

Przykład:

```

POST https://host:port/notifications
Content-Type: application/json; charset=UTF-8

{
  "eventId": "00001",
  "eventTime": "2015-11-16T16:42:25-04:00",
  "eventType": "TroubleTicketStatusChangeNotification",
  "event": {
    "troubleTicket": {
      "statusChangeReason": "Rozpoczęto weryfikację formalną
zgłoszenia",
      "status": "captured",
      "@type": "FaultTicket"
      --Tutaj znajduje się pozostała reprezentacja zasobu zgłoszenia--
    }
  }
}

```

```
}

```

2. Informacja o przyjęciu zgłoszenia, zakończeniu weryfikacji formalnej ze skutkiem pozytywnym, wysyłana przez OPL.

Przykład:

```
POST https://host:port/notifications
Content-Type: application/json; charset=UTF-8

{
  "eventId": "00001",
  "eventTime": "2015-11-16T16:42:25-04:00",
  "eventType": "TroubleTicketStatusChangeNotification",
  "event": {
    "troubleTicket": {
      "statusChangeReason": "Zgłoszenie procesowane",
      "status": "inprogress",
      "@type": "FaultTicket"
      --Tutaj znajduje się pozostała reprezentacja zasobu zgłoszenia-
    }
  }
}
```

- 2a. Informacja o odrzuceniu zgłoszenia, zakończeniu weryfikacji formalnej ze skutkiem negatywnym, wysyłana przez OPL.

Przykład:

```
POST https://host:port/notifications
Content-Type: application/json; charset=UTF-8

{
  "eventId": "00001",
  "eventTime": "2015-11-16T16:42:25-04:00",
  "eventType": "TroubleTicketStatusChangeNotification",
  "event": {
    "troubleTicket" : {
      "@type": "FaultTicket",
      "additionalState": {
        "@type": "Rejection",
        "description": "opis powodu odrzucenia ze słownika RR-FIA",
        "code": "1" #ze słownika RR-FIA
      },
      "status": "rejected",
      "statusChangeReason": "Zgłoszenie odrzucone",
      -- Tutaj znajdują się pozostałe atrybuty zasobu zgłoszenia-
    }
  }
}
```

3. Informacja o zmianie terminu umówienia przez OPL.

Przykład:

```

POST https://host:port/notifications
Content-Type: application/json; charset=UTF-8

{
  "eventId": "00001",
  "eventTime": "2015-11-16T16:42:25-04:00",
  "eventType": "TroubleTicketChangeNotification",
  "event": {
    "troubleTicket": {
      "@type": "FaultTicket",
      "id": "3180",
      "appointmentDate": {
        "@type": "TimeSlot",
        "startDateTime": "2018-06-02T09:00:00.000Z",
        "endDateTime": "2018-06-02T10:00:00.000Z"
      }
    }
  }
}

```

4. Anulowanie zgłoszenia ze strony OA.

Anulowanie zgłoszenia uszkodzenia wymaga utworzenia zasobu zadania **CancelFaultTicketTask** i wypełnienia następujących pól.

cancelReasonCode	Kod powodu anulowania zgłoszenia ze słownika CANCEL-REASON-CODE.
description	Opis powodu anulowania zgłoszenia.
faultTicket	Obiekt referencji wskazujący uszkodzenie które jest przedmiotem zadania.
@type	Pole techniczne określające typ reprezentujący zasób zadania CancelFaultTicketTask.

Opcjonalnie

note	Dodatkowy opis
------	----------------

Przykład:

```

POST https://host:port/tmf/troubleTicketManagement/v1/cancelFaultTicketTask
Content-Type: application/json; charset=UTF-8

{
  "@type": "CancelFaultTicketTask",
  "description": "Opis powodu anulowania",
  "cancelReasonCode": "15001",
  "note": "Dodatkowy opis",
  "faultTicket": {
    "id": "123456",
    "@referredType": "FaultTicket"
  }
}

```

OPL przekazuje informacje o zakończeniu zadania anulowania zgłoszenia uszkodzenia.

```
POST https://host:port/notifications
Content-Type: application/json; charset=UTF-8

{
  "eventId": "00001",
  "eventTime": "2015-11-16T16:42:25-04:00",
  "eventType": "CancelFaultTicketTaskStateChangeNotification",
  "event": {
    "cancelFaultTicketTask": {
      "@type": "CancelFaultTicketTask",
      "exitCode": "1", #kod ze słownika
      "exitCodeDescription": "opis ze słownika FAULT-CANCEL-REASON",
      "state": "done",
      -- Tutaj znajduje się pozostała reprezentacja zasobu zadania
    }
  }
}
```

OPL przekazuje informacje o zmianie stanu zasobu uszkodzenia.

```
POST https://host:port/notifications
Content-Type: application/json; charset=UTF-8

{
  "eventId": "00001",
  "eventTime": "2015-11-16T16:42:25-04:00",
  "eventType": "TroubleTicketStatusChangeNotification",
  "event": {
    "faultTicket": {
      "@type": "FaultTicket",
      "additionalState": {
        "code": "15001",
        "@type": "Cancel",
        "description": "swobodny opis powodu odrzucenia skopiowany
z zadania",
      },
      "status": "cancelled",
      "statusChangeReason": "Zgłoszenie anulowane",
      -- Tutaj znajduje się pozostała reprezentacja zasobu zadania
    }
  }
}
```

5. Przesłanie informacji o skutecznej realizacji do OA.

Przykład:

```
POST https://host:port/notifications
Content-Type: application/json; charset=UTF-8

{
  "eventId": "00001",
  "eventTime": "2015-11-16T16:42:25-04:00",
  "eventType": "TroubleTicketResolvedNotification",
  "event": {
```

```

"troubleTicket": {
  "@type": "FaultTicket",
  "resolutionDate": "2019-01-30T08:00:00",
  "additionalState": {
    "@type": "Exit",
    "description": "opis kodu wyjścia ze słownika EC-FIA",
    "code": "12009" # ze słownika EC-FIA
  },
  "status": "resolved",
  "statusChangeReason": "Zgłoszenie rozwiązane",
  "note": [{
    "date": "2019-01-30T08:00:00",
    "author": "OPL",
    "text": "Odebrano urządzenia: ONTHG8010H (STRCT_f0c166),
MODTMPLSTANDARD (STRCT_f0c165). Dostarczono urządzenia: ONTHG8010H
(STRCT_f0c166), MODTMPLSTANDARD (GEN_AT_1a6599)",
    "@type": "Note"}],
  -- Tutaj znajduje się pozostała reprezentacja zasobu zgłoszenia
}
}
}

```

W sytuacji wymiany urządzenia zasób uszkodzenia zostanie uzupełniony o sformalizowaną notatkę o następującej konstrukcji:

- Fragment „Odebrano urządzenia:” wstawiany będzie tylko jeśli występują rekordy z odebranymi od abonenta urządzeniami.
- Fragment „Dostarczono urządzenia:” wstawiany będzie tylko jeśli występują rekordy z dostarczonymi do abonenta urządzeniami.

Informacja o urządzeniach będzie prezentowana w postaci listy "<<model_code>> (<<serial_number>>)". Opis pary atrybutów urządzeń "<<model_code>> (<<serial_number>>)" będzie rozdzielona przecinkami.

6. Przesłanie informacji o potwierdzeniu skutecznej realizacji uszkodzenia przez OA.

Następujące pola zasobu operator musi wypełnić w ramach realizacji operacji PATCH:

id	Unikalny identyfikator zasobu zgłoszenia uszkodzenia.
status	Stan zasobu zgłoszenia. Dopuszczalny tylko closed .
@type	Pole technicznie określające jakie typu jest zasób.

Pola opcjonalne:

Note	Dodatkowe notatki.
------	--------------------

Przykład:

```
PATCH https://host:port/tmf/troubleTicketManagement/v1/troubleTicket/3180
Content-Type: application/merge-patch+json; charset=UTF-8
If-Match: e6c19694e0a94702c50ea93d0455ec83

{
  "@type": "FaultTicket",
  "id": "3180",
  "status": "closed"
}
```

6a Przekazanie informacji o zamknięciu zgłoszenia uszkodzenia.

Przykład:

```
POST https://host:port/notifications
Content-Type: application/json; charset=UTF-8

{
  "eventId": "00001",
  "eventTime": "2015-11-16T16:42:25-04:00",
  "eventType": "TroubleTicketStatusChangeNotification",
  "event": {
    "troubleTicket": {
      "@type": "FaultTicket",
      "status": "closed",
      "statusChangeReason": "Zgłoszenie zakończone",
      --Tutaj znajduje się pozostała reprezentacja zasobu zgłoszenia--
    }
  }
}
```

6b. Przesłanie informacji o odrzuceniu przez OA rozwiązania OPL

Następujące pola zasobu operator musi wypełnić w ramach realizacji operacji:

id	Unikalny identyfikator zasobu zgłoszenia uszkodzenia.
Status	Stan zasobu zgłoszenia. Dopuszczalny tylko inprogress .

Pola opcjonalne:

lowerOptionAcceptance	Zgoda na obniżenie prędkości.
Note	Lista notatek rozbudowana z dodaną kolejną pozycją. Lista musi zawierać wszystkie dotychczasowe notatki plus nową.

Przykład:

```
PATCH /tmf/troubleTicketManagement/v1/troubleTicket/3180
Content-Type: application/merge-patch+json; charset=UTF-8
If-Match: e6c19694e0a94702c50ea93d0455ec83
```

```
{
  „@type”: „FaultTicket”,
  „id”: „3180”,
  „status”: „inprogress”
}
```

6c. Przesłanie informacji o zakończeniu realizacji zgłoszenia w OPL powoduje zakończenie procesowania uszkodzenia.

Przykład:

```
POST https://host:port/notifications
Content-Type: application/json; charset=UTF-8

{
  „eventId”: „00001”,
  „eventTime”: „2015-11-16T16:42:25-04:00”,
  „eventType”: „TroubleTicketStatusChangeNotification”,
  „event”: {
    „troubleTicket”: {
      „@type”: „FaultTicket”,
      „status”: „closed”,
      „statusChangeReason”: „Zgłoszenie zakończone”,
      --Tutaj znajduje się pozostała reprezentacja zasobu zgłoszenia-
    }
  }
}
```

7. Wysłanie informacji o podpięcie zgłoszenia pod Awarię Masową/Pracę Planową. Na liście notatek pojawi się nowa notatka informująca o numerach powiązanych zgłoszeń.

Przykład:

```
POST https://host:port/notifications
Content-Type: application/json; charset=UTF-8

{
  „eventId”: „00001”,
  „eventTime”: „2015-11-16T16:42:25-04:00”,
  „eventType”: „TroubleTicketStatusChangeNotification”,
  „event”: {
    „troubleTicket”: {
      „@type”: „FaultTicket”,
      „status”: „held”,
      „statusChangeReason”: „Zgłoszenie oczekuje na zakończenie
innych czynności”,
      „note”: [
        ...
        {
          „date”: „2019-01-30T10:00:00”,
          „author”: „OPL”,
          „text”: „Awaria masowa 1234567890”,
          „@type”: „Note”
        }
      ],
      --Tutaj znajduje się pozostała reprezentacja zasobu zgłoszenia-
    }
  }
}
```

```
}
}
```

7a. Wysłanie informacji o odłączeniu zgłoszenia pod Awarię Masową/Pracę Planową. Na liście notatek pojawi się nowa notatka informująca o numerach dotychczas powiązanych zgłoszeń.

Przykład:

```
POST https://host:port/notifications
Content-Type: application/json; charset=UTF-8

{
  „eventId”: „00001”,
  „eventTime”: „2015-11-16T16:42:25-04:00”,
  „eventType”: „TroubleTicketStatusChangeNotification”,
  „event”: {
    „troubleTicket”: {
      „@type”: „FaultTicket”,
      „status”: „inprogress”,
      „statusChangeReason”: „Zgłoszenie procesowane”,
      --Tutaj znajduje się pozostała reprezentacja zasobu zgłoszenia--
    }
  }
}
```

Awaria masowa

Poniższa tabelka obrazuje powiązania między stanami zasobu zgłoszenia a procesem obsługi zgłoszenia awarii masowej. Wytluszczenie wskazuje na stany końcowe (terminalne).

Etap	Stan procesu zgłoszenia	Stan zasobu zgłoszenia	Informacja zwracana OA
1	Przyjęcie komunikatu pozytywnie zweryfikowanego informatycznie	acknowledged	HTTP202 (partial create)
1a	Negatywna weryfikacja informatyczna.	Zasób zgłoszenia nie jest tworzony nie posiada stanu	HTTP4XX ErrorRepresentation
2	Pozytywny wynik weryfikacji formalnej.	Inprogress	Notyfikacja TroubleTicketStatusChangeNotification
2a	Negatywna weryfikacja formalna.	Rejected	Notyfikacja TroubleTicketStatusChangeNotification
3	Uzupełnienie danych o awarii przez OPL	inprogress	Notyfikacja TroubleTicketChangeNotification
4	Realizacja skuteczna.	Resolved	Notyfikacja TroubleTicketResolvedNotification

5	Upłynął czas na odpowiedź OA	closed	Notyfikacja TroubleTicketStatusChangeNotification
6	Potwierdzenie skutecznej realizacji przez OA	closed	http200 Successful PATCH lub http4XX z opisem błędu

Przykład:

1. Utworzenie zgłoszenia masowego.

Poniższa tabelka zawiera listę atrybutów zgłoszenia modyfikowanych w trakcie wykonania operacji na zasobie. Część atrybutów określa operator, pozostałe OPL. Należy pamiętać o polach wymaganych wynikających z modelu danych oraz wynikających z zastosowania metody.

Atrybuty niewymagane których operator nie wypełnia i nie są zamieszczone w tabelce nie powinny być przesyłane. Pola nieużywane i niewymagane nie są opisane w tabelce.

Atrybuty	Wymagane	Wartość domyślna	Reguła
externalId	Nie		Identyfikator własny operatora, pole nie jest przetwarzane przez OPL
ticketType	Tak		massFault
description	Tak		Opis swobodny zgłoszenia
name	Nie		Nazwa zgłoszenia
creationDate			Wypełniane przez OPL
creationDateOA	Tak		Data rejestracji zgłoszenia w środowisku operatora
lastUpdate			Wypełnianie przez OPL
severity			Wypełniane przez OPL
priority			Wypełniane przez OPL
requestedResolutionDate			Wypełniane przez OPL
expectedResolutionDate			Wypełniane przez OPL
resolutionDate			Wypełnianie przez OPL
status			Wypełnianie przez OPL

Atrybuty	Wymagane	Wartość domyślna	Reguła
@type	Tak	MassFaultTicket	Typ obiektu: MassFaultTicket
@baseType	Nie	TroubleTicket	TroubleTicket
relatedEntity	Nie		<p>Pole relatedEntity może zawierać identyfikator instancji produktu w roli damagedService którego dotyczy zgłoszenie. Nie jest on wymagany. W ramach awarii OA może podać jedną usługę objętą awarią masową (rozumianą jako para produktów ACCESS i BITSTREAML2).</p> <p>Lista musi zawierać referencje tylko do instancji produktów ACCESS i BISTREAML2 w roli damagedService. Wyprecyzowanie instancji produktów nie należących do danego operatora spowoduje błąd walidacji informatycznej związany z nieprawidłową zawartością pola.</p>
statusChange			Wypełnianie przez OPL, lista zmian statusów zasobu
note	Nie		
relatedParty	Tak		Lista referencja do obiektów party. Lista ta może zawierać jedynie jeden obiekt operatora alternatywnego Organization w roli owner . Niewłaściwa zawartość listy spowoduje błąd na etapie weryfikacji informatycznej zawartością pola.

Atrybuty	Wymagane	Wartość domyślna	Reguła
ticketRelationship			Wypełniane przez OPL, gdy awaria jest powiązana z innym zgłoszeniem.
Channel			Wypełniane przez OPL.
pduId	Tak		Identyfikator PDU.
faultSymptom	Tak		Lista symptomów awarii, można podać maksymalnie 3 symptomów.

OPL waliduje zawartość atrybutów i w sytuacji błędu zwraca informację w postaci błędu weryfikacji informatycznej.

```
POST https://host:port/tmf/troubleTicketManagement/v1/troubleTicket
Content-Type: application/json; charset=UTF-8
```

```
{
  „name”: „Awaria masowa”,
  „externalId”: „21308090”,
  „ticketType”: „massFault”,
  „description”: „Usługi z listy nie działają”,
  „creationDateOA”: „2015-11-15T16:00:25-01:00”,
  „pduId”: „id pdu”,
  „vlan2pdu”: „id vlan”,
  „@type”: „MassFaultTicket”,
  „faultSymptom”: [
    {
      „@type”: „FaultSymptom”,
      „symptom”: „20001”,
      „description”: „opis symptom nr 1”
    },
    {
      „@type”: „FaultSymptom”,
      „symptom”: „20002”,
      „description”: „opis symptom nr 2”
    },
    {
      „@type”: „FaultSymptom”,
      „symptom”: „20003”,
      „description”: „opis symptom nr 3”
    }
  ],
  „relatedEntity”: [
    {
      „id”: „1234567890”,
      „role”: „damagedService”,
      „@referredType”: „Product”
    },
    {
      „id”: „1234567891”,
```

```

        „role”: „damagedService”,
        „@referredType”: „Product”
    },
    },
    „note”: [{
        „@type”: „Note”,
        „date”: „2018-05-01T00:00”,
        „author”: „Janusz Nowak”,
        „text”: „Proszę o szybka interwencję w tej sprawie”
    }],
    „relatedParty”: [{
        „id”: „4”,
        „name”: „TM-Polska”,
        „role”: „owner”,
        „@referredType”: „Organization”
    }
    ]
}

```

2. Informacja o pozytywnej weryfikacji formalnej.

Przykład:

```

POST https://host:port/notifications
Content-Type: application/json; charset=UTF-8

{
    „eventId”: „00001”,
    „eventTime”: „2015-11-16T16:42:25-04:00”,
    „eventType”: „TroubleTicketStatusChangeNotification”,
    „event”: {
        „troubleTicket”: {
            „@type”: „MassFaultTicket”,
            „status”: „inprogress”,
            „statusChangeReason”: „Zgłoszenie procesowane”,
            --Tutaj znajduje się pozostała reprezentacja zasobu zgłoszenia
        }
    }
}

```

2a. Informacja o negatywnej weryfikacji formalnej.

Przykład:

```

POST https://host:port/notifications
Content-Type: application/json; charset=UTF-8

{
    „eventId”: „00001”,
    „eventTime”: „2015-11-16T16:42:25-04:00”,
    „eventType”: „TroubleTicketStatusChangeNotification”,
    „event”: {
        „troubleTicket”: {
            „@type”: „MassFaultTicket”,
            „additionalState”: {
                „type”: „Rejection”,

```

```

        „description”: „opis kodu wyjścia”,
        „code”: „1001”, # ze słownika RR-FIA
    },
    „status”: „rejected”,
    „statusChangeReason”: „Zgłoszenie odrzucone”,
    --Tutaj znajduje się pozostała reprezentacja zasobu zgłoszenia
}
}
}

```

3. Informacja o zmianie danych awarii przesyłana przez OPL.

Przykład:

```

POST https://host:port/notifications
Content-Type: application/json; charset=UTF-8

{
  "eventId": "00001",
  "eventTime": "2015-11-16T16:42:25-04:00",
  "eventType": "TroubleTicketChangeNotification",
  "event": {
    "troubleTicket": {
      "@type": "MassFaultTicket",
      "id": "3180",
      "ticketRelationship": {
        "id": "3456",
        "type": "dependant",
        "@type": "TicketRelationship"
      }
    }
  }
}

```

4. Informacja o skutecznej realizacji.

Przykład:

```

POST https://host:port/notifications
Content-Type: application/json; charset=UTF-8

{
  „eventId”:„00001”,
  „eventTime”:„2015-11-16T16:42:25-04:00”,
  „eventType”:„TroubleTicketResolvedNotification”,
  „event”: {
    „troubleTicket”: {
      „@type”: „MassFaultTicket”,
      „status”: „resolved”,
      „resolutionDate”: „2019-01-30T08:00:00”,
      „statusChangeReason”: „Zgłoszenie rozwiązane”,
      --Tutaj znajduje się pozostała reprezentacja zasobu zgłoszenia
    }
  }
}

```

5. Upłynął czas na odpowiedź OA.

Przykład:

```
POST https://host:port/notifications
Content-Type: application/json; charset=UTF-8

{
  „eventId”: „00001”,
  „eventTime”: „2015-11-16T16:42:25-04:00”,
  „eventType”: „TroubleTicketStatusChangeNotification”,
  „event”: {
    „troubleTicket”: {
      „@type”: „MassFaultTicket”,
      „status”: „closed”,
      „statusChangeReason”: „Zgłoszenie zakończone”,
      --Tutaj znajduje się pozostała reprezentacja zasobu zgłoszenia--
    }
  }
}
```

6. Potwierdzenie skutecznej realizacji zgłoszenia przez OA.

Następujące pola są wymagane, operator wypełnia je przed wykonaniem operacji PATCH na zasobie.

id	Unikalny identyfikator zgłoszenia.
status	Status zgłoszenia – dopuszczalny jedynie closed .
@type	Pole techniczne określające jakiego typu jest zasób.

Pozostałe pola nie będą dostępne do modyfikacji.

Przykład:

```
PATCH /tmf/troubleTicketManagement/v1/troubleTicket/3180
Content-Type: application/merge-patch+json; charset=UTF-8
If-Match: e6c19694e0a94702c50ea93d0455ec83

{
  "@type": "MassFaultTicket",
  "id": "3180",
  "status": "closed"
}
```

Reklamacja

Poniższa tabelka obrazuje powiązania między stanami zasobu zgłoszenia a procesem obsługi zgłoszenia reklamacji. Wytluszczenie wskazuje na stany końcowe (terminalne).

Etap	Stan procesu zgłoszenia	Stan zasobu zgłoszenia	Informacja zwracana OA
1	Przyjęcie komunikatu pozytywnie	acknowledged	HTTP202 (partial create)

	zweryfikowanego informatycznie		
1a	Negatywna weryfikacja informatyczna.	Zasób zgłoszenia nie jest tworzony nie posiada stanu	HTTP4XX ErrorRepresentation
2	Pozytywny wynik weryfikacji formalnej.	inprogress	Notyfikacja TroubleTicketStatusChangeNotification
3	Wymagane dodatkowe informacje	pending	Notyfikacja TroubleTicketInformationRequiredNotification
3a	Odpowiedź OA	inprogress	HTTP200 (successful patch) lub HTTP4XX (unsuccessful path)
3a1	Wymagane dodatkowe informacje - eskalacja	pending	Notyfikacja TroubleTicketInformationRequiredNotification
3a2	Minął termin dla prawidłowej odpowiedzi od OA	resolved	Notyfikacja TroubleTicketResolvedNotification
4	Przesunięcie terminu przez OPL	inprogress	Notyfikacja TroubleTicketChangeNotification
5	Realizacja skuteczna.	resolved	Notyfikacja TroubleTicketResolvedNotification
6	Zamknięcie zgłoszenia	closed	Notyfikacja TroubleTicketStatusChangeNotification

1. Utworzenie zgłoszenia reklamacji przez OA

Poniższa tabelka zawiera listę atrybutów zgłoszenia modyfikowanych w trakcie wykonania operacji na zasobie. Część atrybutów określa operator, pozostałe OPL. Należy pamiętać o polach wymaganych wynikających z modelu danych oraz wynikających z zastosowania metody.

Atrybuty niewymagane których operator nie wypełnia i nie są zamieszczone w poniższej tabelce nie powinny być przesyłane. Pola nieużywane i niewymagane nie są opisane w tabelce.

Atrybuty	Wymagane	Wartość domyślna	Reguła
externalId	Nie		Identyfikator własny operatora, pole nie jest przetwarzane przez OPL
ticketType	Tak		complaint
description	Tak		Opis zgłoszenia.

Atrybuty	Wymagane	Wartość domyślna	Reguła
name	Nie		Nazwa zgłoszenia
creationDate			Wypełnianie przez OPL
lastUpdate			Wypełnianie przez OPL
reason	Tak		Przedmiot reklamacji ze słownika COMPLAINT-SUBJECT .
priority			Wypełniane przez OPL
severity			Wypełniane przez OPL
requestedResolutionDate			Wypełniane przez OPL
expectedResolutionDate			Wypełniane przez OPL
resolutionDate			Wypełnianie przez OPL
status			Wypełnianie przez OPL
@type	Tak	ComplaintTicket	Typ obiektu: ComplaintTicket
@baseType	Nie	TroubleTicket	TroubleTicket
relatedEntity	Tak		Lista referencji do obiektów reprezentujących instancje produktów w roli reportedService które są przedmiotem reklamacji. Dopuszczalne jest wyspecyfikowanie dwóch produktów: ACCESS oraz BITSTREAML2 . Nieistniejące produkty, nieprzypisane do właściwego operatora spowodują błąd związany z weryfikacją informatyczną (niewłaściwa wartość w polu).
statusChange			Wypełnianie przez OPL, lista zmian statusów zasobu
note	Nie		

Atrybuty	Wymagane	Wartość domyślna	Reguła
relatedParty	Tak		Lista zawierająca referencję do obiektu Organization reprezentującego operatora alternatywnego ("role": " owner ") i obiektu Person zawierającego dane abonenta. Tylko lista z tymi dwoma obiektami jest dopuszczalna. Nieprawidłowo wypełniona lista spowoduje błąd weryfikacji informatycznej (niewłaściwa wartość w polu).
channel			Wypełniane przez OPL.
justification	Tak		Pole wymagane, krótki opis uzasadnienia reklamacji.

Przykład:

POST https://host:port/tmf/troubleTicketManagement/v1/troubleTicket
 Content-Type: application/json; charset=UTF-8

```
{
  "name": "Zgłoszenie reklamacji TM",
  "externalId": "21308090",
  "ticketType": "complaint",
  "description": "DSL nie osiąga prędkości z umowy",
  "@type": "ComplaintTicket",
  "reason": "7",
  "relatedEntity": [
    {
      {
        "id": "1234567890",
        "role": "reportedService",
        "@referredType": "Product"
      },
      {
        "id": "1234567890",
        "role": "reportedService",
        "@referredType": "Product"
      }
    ]
  ],
  "justification": "Krótki opis uzasadnienia reklamacji",
  "relatedParty": [
    {
      "id": "4",
      "name": "T-Mobile",
      "role": "owner",

```

```

        "@referredType": "Organization"
    },
    {
        "name": "Jan Kowalski",
        "role": "customer",
        "number": "+436641234567",
        "@type": "Person"
    }
]
}

```

2. Przesłanie informacji o pozytywnej weryfikacji formalnej reklamacji.

```

POST https://host:port/notifications
Content-Type: application/json; charset=UTF-8

{
    "eventId": "00001",
    "eventTime": "2015-11-16T16:42:25-04:00",
    "eventType": "TroubleTicketStatusChangeNotification",
    "event": {
        "troubleTicket": {
            "@type": "ComplaintTicket",
            "status": "inprogress",
            "statusChangeReason": "Zgłoszenie procesowane",
            -- Tutaj znajduje się pozostała reprezentacja zasobu zgłoszenia
        }
    }
}

```

3. Przesłanie informacji o konieczności uzupełnienia reklamacji.

Informacja opisowa określająca co jest przedmiotem koniecznego uzupełnienia reklamacji. Informacja znajduje się w notatce wskazanej w akcji „quest” w polu **fieldPath**.

```

POST https://host:port/notifications
Content-Type: application/json; charset=UTF-8

{
    "eventId": "00001",
    "eventTime": "2015-11-16T16:42:25-04:00",
    "eventType": "TroubleTicketInformationRequiredNotification",
    "resourcePath": "TroubleTicketManagement/v2/TroubleTicket/3248",
    "fieldPath": "quest=note[5] //answer, missing",
    "event": {
        "troubleTicket": {
            "@type": "ComplaintTicket",
            "status": "pending",
            "statusChangeReason": "Zgłoszenie oczekuje na informacje",
            -- Tutaj znajduje się pozostała reprezentacja zasobu zgłoszenia
        }
    }
}

```

3a. OA uzupełnia reklamację.

Następujące pola są wymagane, operator wypełnia je przed wykonaniem operacji PATCH na zasobie.

id	Unikalny identyfikator zgłoszenia
note	Lista notatek z nową notatką zawierającą odpowiedź operatora
@type	Pole techniczne określające jakiego typu jest zasób.

```
PATCH /tmf/troubleTicketManagement/v1/troubleTicket/3248
Content-Type: application/merge-patch+json; charset=UTF-8
If-Match: e6c19694e0a94702c50ea93d0455ec83
```

```
{
  "@type": "ComplaintTicket",
  "id": "3248",
  "note": [{
    "@type": "Note",
    "author": "OA",
    "date": "2018-05-01T00:00",
    "text": "Przykładowe informacje uzupełniające"
  }],
}
```

3a1 Powiadomienie OA o upływie czasu przeznaczonego na odpowiedź - eskalacja.

```
POST https://host:port/notifications
Content-Type: application/json; charset=UTF-8
```

```
{
  "eventId": "00001",
  "eventTime": "2015-11-16T16:42:25-04:00",
  "eventType": "TroubleTicketInformationRequiredNotification",
  "resourcePath": "TroubleTicketManagement/v1/TroubleTicket/3248",
  "fieldPath": "escalation=note[5]", // eskalacja
  "event": {
    "troubleTicket": {
      "@type": "ComplaintTicket",
      "status": "pending",
      -- Tutaj znajduje się pozostała reprezentacja zasobu zgłoszenia
    }
  }
}
```

3a2. Powiadomienie OA o upływie czasu przeznaczonego na odpowiedź anulowanie zgłoszenia.

```
POST https://host:port/notifications
Content-Type: application/json; charset=UTF-8
```

```
{
  "eventId": "00001",
  "eventTime": "2015-11-16T16:42:25-04:00",
  "eventType": "TroubleTicketResolvedNotification",
  "event": {
    "troubleTicket": {
      "@type": "ComplaintTicket",

```

```

        "additionalState": {
            "@type": "Exit",
            "description": "Rozpatrzone negatywnie",
            "code": "14002" #Kod i opis ze słownika EC-FIA
        },
        "status": "resolved",
        --Tutaj znajduje się pozostała reprezentacja zasobu zgłoszenia
    }
}

```

4. Przesunięcie terminu realizacji reklamacji przez OPL.

```

POST https://host:port/notifications
Content-Type: application/json; charset=UTF-8

{
    "eventId": "00001",
    "eventTime": "2015-11-16T16:42:25-04:00",
    "eventType": "TroubleTicketChangeNotification",
    "event": {
        "troubleTicket": {
            "@type": "ComplaintTicket",
            "id": "3180",
            "expectedResolutionDate": "2015-11-16T00:00:00",
            "additionalState": {
                "@type": "Changed",
                "description": "Uzasadnienie, przyczyna przesunięcia
terminu rozpatrzenia reklamacji",
                "code": "14003" #Kod i opis ze słownika EC-FIA
            }
        }
    }
}

```

5. Powiadomienie o realizacji skutecznej zgłoszenia reklamacji

```

POST https://host:port/notifications
Content-Type: application/json; charset=UTF-8

{
    "eventId": "00001",
    "eventTime": "2015-11-16T16:42:25-04:00",
    "eventType": "TroubleTicketResolvedNotification",
    "event": {
        "troubleTicket": {
            "@type": "ComplaintTicket",
            "additionalState": {
                "@type": "Exit",
                "description": "Opis uzasadnienia ze słownika",
                "code": "14001", #Kod i opis ze słownika EC-FIA
            },
            "grantedAmount": "150.10" #jeśli uznano roszczenia,
            "resolutionDate": "2019-01-30T08:00:00",
            "status": "resolved",
            "statusChangeReason": "Zgłoszenie rozwiązane",
            --Tutaj znajduje się pozostała reprezentacja zasobu zgłoszenia
        }
    }
}

```

```
}
}
```

6. Powiadomienie o zamknięciu zgłoszenia reklamacji

```
POST https://host:port/notifications
Content-Type: application/json; charset=UTF-8

{
  "eventId": "00001",
  "eventTime": "2015-11-16T16:42:25-04:00",
  "eventType": "TroubleTicketStatusChangeNotification",
  "event": {
    "troubleTicket": {
      "@type": "ComplaintTicket",
      "status": "closed",
      "statusChangeReason": "Zgłoszenie zakończone",
      --Tutaj znajduje się pozostała reprezentacja zasobu zgłoszenia
    }
  }
}
```

Asysta techniczna

Poniższa tabelka obrazuje powiązania między stanami zasobu zgłoszenia a procesem obsługi zgłoszenia asysty technicznej. Wyłuszczenie wskazuje na stany końcowe (terminalne).

Etap	Stan procesu zgłoszenia	Stan zasobu zgłoszenia	Informacja zwracana OA
1	Przyjęcie komunikatu pozytywnie zweryfikowanego informatycznie.	acknowledged	HTTP202 (partial create)
1a	Negatywna weryfikacja informatyczna.	Zasób zgłoszenia nie jest tworzony nie posiada stanu	HTTP4XX ErrorRepresentation
2	Pozytywny wynik weryfikacji formalnej.	inprogress	Notyfikacja TroubleTicketStatusChangeNotification
2a	Negatywna weryfikacja formalna.	rejected	Notyfikacja TroubleTicketStatusChangeNotification
3	Wniosek OPL o zmianę terminu	pending	Notyfikacja TroubleTicketInformationRequiredNotification
3a	Potwierdzenie zmiany terminu przez OA	inprogress	HTTP200 (successfull patch) z ewentualnym w przypadku błędu lub niemożności zmiany terminu ErrorRepresentation
4	Wniosek OA o zmianę terminu	inprogress	HTTP200 (successfull patch) z ewentualnym w przypadku błędu lub niemożności zmiany terminu ErrorRepresentation

5	OA anuluje zgłoszenie asysty	cancelled	OA tworzy zadanie anulowania asysty technicznej - CancelTechnicalAssistTicketTask . Po zakończeniu zadania przekazywana jest notyfikacja CancelTechnicalAssistTicketStatusChangeNotification informująca o zakończeniu zadania oraz notyfikacja TroubleTicketStatusChangeNotification o zmianie stanu zasobu asysty. W przypadku błędów związanych z utworzeniem zadania przekazywany jest ErrorRepresentation
6	OPL powiadamia OA o konieczności usunięcia usterki w ramach asysty	held	Notyfikacja TroubleTicketStatusChangeNotification W sytuacji gdy OPL nie zna daty realizacji naprawy uszkodzenia gdy zostanie ona ustalona wysyłana jest następna notyfikacja informująca o ustalonej dacie. TroubleTicketChangeNotification
6a	OPL powiadamia o usunięciu usterki w ramach asysty	inprogress	Notyfikacja TroubleTicketStatusChangeNotification
7	Realizacja skuteczna.	resolved	Notyfikacja TroubleTicketResolvedNotification
8	Zamknięcie zgłoszenia	closed	Notyfikacja TroubleTicketStatusChangeNotification

1. Utworzenie zgłoszenia asysty technicznej w OPL

Poniższa tabelka zawiera listę atrybutów zgłoszenia modyfikowanych w trakcie wykonania operacji na zasobie. Część atrybutów określa operator, pozostałe OPL. Należy pamiętać o polach wymaganych wynikających z modelu danych.

Atrybuty niewymagane których operator nie wypełnia i nie są zamieszczone w tabelce nie powinny być przesyłane. Pola nieużywane i niewymagane nie są opisane w tabelce.

Atrybuty	Wymagane	Wartość domyślna	Reguła
externalId	Nie		Identyfikator własny operatora, pole nie jest przetwarzane przez OPL
ticketType	Tak		technicalAssist
description	Tak		Opis swobodny zgłoszenia
name	Nie		Nazwa zgłoszenia
creationDate			Wypełnianie przez OPL

Atrybuty	Wymagane	Wartość domyślna	Reguła
lastUpdate			Wypełnianie przez OPL
severity			Wypełniane przez OPL
priority			Wypełniane przez OPL
requestedResolutionDate			Wypełniane przez OPL
expectedResolutionDate			Wypełniane przez OPL
resolutionDate			Wypełnianie przez OPL
status			Wypełnianie przez OPL
@type	Tak	AssistTicket	Pole techniczne określające typ obiektu, tutaj AssistTicket
@baseType	Nie	TroubleTicket	Pole techniczne określające typ bazowy – TroubleTicket.
relatedEntity	Tak		Lista referencji do obiektów reprezentujących instancje produktów w roli service które są przedmiotem asysty. Dopuszczalne jest wyspecyfikowanie dwóch produktów: ACCESS oraz BITSTREAML2 . Nieistniejące produkty, nieprzypisane do właściwego operatora spowodują błąd związany z weryfikacją informatyczną (niewłaściwa wartość w polu).
statusChange			Wypełnianie przez OPL, lista zmian statusów zasobu
note	Nie		
relatedParty	Tak		Lista obiektów zawierająca referencje do obiektów typu Organization lub wartości typu Person. Obiekty z listy są w relacji ze zgłoszeniem asysty: <ul style="list-style-type: none"> operatora (referencja do obiektu Organization w "role": "owner"),

Atrybuty	Wymagane	Wartość domyślna	Reguła				
			<ul style="list-style-type: none">• abonenta (wartość jako obiekt Person)• partnera technicznego operatora wartość jako obiekt Person w "role": "technicalPartnerOA"). Partner techniczny musi posiadać numer telefonu number. <p>Tylko lista z wymienionymi powyżej obiektami jest dopuszczalna. Nieprawidłowo wypełniona lista spowoduje błąd weryfikacji informatycznej (niewłaściwa wartość w polu).</p>				
channel			Wypełniane przez OPL				
pduCheck	Nie	false	Sprawdzono usługi na PDU: true, false				
netCheck	Nie	false	Sprawdzono infrastrukturę w lokalu abonenta: true, false				
hardwareCheck	Nie	false	Sprawdzono sprzęt u abonenta: true, false				
appointmentTimeSlot	Tak		<p>Slot czasowy podany w postaci dat z czasem czyli daty początku slotu startDateTime i końca endDateTime. Sloty należy podawać zgodnie ze schematem:</p> <table><tr><td>Pn. – Pt. od 8:00 do 10:00</td></tr><tr><td>Pn. – Pt. Od 10:00 do 12:00</td></tr><tr><td>Pn. – Pt. Od 12:00 do 14:00</td></tr><tr><td>Pn. – Pt. Od 14:00 do 16:00</td></tr></table> <p>Slot jest określany datami od do tj:</p> <p>"appointmentTimeSlot": { "@type": "TimeSlot",</p>	Pn. – Pt. od 8:00 do 10:00	Pn. – Pt. Od 10:00 do 12:00	Pn. – Pt. Od 12:00 do 14:00	Pn. – Pt. Od 14:00 do 16:00
Pn. – Pt. od 8:00 do 10:00							
Pn. – Pt. Od 10:00 do 12:00							
Pn. – Pt. Od 12:00 do 14:00							
Pn. – Pt. Od 14:00 do 16:00							

Atrybuty	Wymagane	Wartość domyślna	Reguła
			<pre>"startDateTime": "2018-09-10T08:00:00Z", "endDateTime": "2018-09-10T10:00:00Z" }</pre>

Przykład:

POST https://host:port/tmf/troubleTicketManagement/v1/troubleTicket
Content-Type: application/json; charset=UTF-8

```
{
  "name": "Zgłoszenie asysty TM",
  "externalId": "21308090",
  "ticketType": "technicalAssist",
  "description": "Wymagana asysta techniczna w miejscu",
  "appointmentTimeSlot": {
    "@type": "TimeSlot",
    "startDateTime": "2018-05-01T08:00",
    "endDateTime": "2018-05-01T10:00"
  },
  "pduCheck": true,
  "netCheck": false,
  "hardwareCheck": true,
  "@type": "AssistTicket",
  "relatedEntity": [
    {
      "id": "1234567890",
      "role": "service",
      "@referredType": "Product"
    },
    {
      "id": "1234567890",
      "role": "service",
      "@referredType": "Product"
    }
  ],
  "note": [
    {
      "@type": "Note",
      "date": "2018-05-01T00:00",
      "author": "Janusz Nowak",
      "text": "Proszę o szybka interwencję w tej sprawie"
    }
  ],
  "relatedParty": [
    {
      "id": "4",
      "name": "T-Mobile",
      "role": "owner",
      "@referredType": "Organization"
    },
    {
      "@type": "Person",
      "name": "Jan Kowalski",

```

```

        "role": "technicalPartnerOA",
        "number": "+4850607025"
    },
    {
        "@type": "Person",
        "name": "Jan Nowak",
        "role": "customer",
        "number": "+4850607025"
    }
]
}

```

2. Informacja o pozytywnym wyniku weryfikacji formalnej

```

POST https://host:port/notifications
Content-Type: application/json; charset=UTF-8

{
  "eventId": "00001",
  "eventTime": "2015-11-16T16:42:25-04:00",
  "eventType": "TroubleTicketStatusChangeNotification",
  "event": {
    "troubleTicket": {
      "@type": "AssistTicket",
      "status": "inprogress",
      "statusChangeReason": "Zgłoszenie procesowane",
      --Tutaj znajduje się pozostała reprezentacja zasobu zgłoszenia
    }
  }
}

```

2a. Informacja o negatywnym wyniku weryfikacji formalnej

```

POST https://host:port/notifications
Content-Type: application/json; charset=UTF-8

{
  "eventId": "00001",
  "eventTime": "2015-11-16T16:42:25-04:00",
  "eventType": "TroubleTicketStatusChangeNotification",
  "event": {
    "troubleTicket" : {
      "@type": "AssistTicket",
      "additionalState": {
        "@type": "Rejection",
        "description": "Powód odrzucenia zgłoszenia ze słownika
RR-FIA",
        "code": "1", # ze słownika RR-FIA
      },
      "status": "rejected",
      --Tutaj znajduje się pozostała reprezentacja zasobu zgłoszenia
    }
  }
}

```

3. Przekazanie informacji o zmianie terminu przez OPL do OA

```

POST https://host:port/notifications
Content-Type: application/json; charset=UTF-8

{
  "eventId": "00001",
  "eventTime": "2015-11-16T16:42:25-04:00",
  "eventType": "TroubleTicketInformationRequiredNotification",
  "resourcePath": "TroubleTicketManagement/v1/TroubleTicket/3180",
  "fieldPath": "accept=appointmentTimeSlot",
  "event": {
    "troubleTicket": {
      "@type": "AssistTicket",
      "appointmentTimeSlot" {
        "@type": "TimeSlot",
        "startDateTime": "2018-08-09T14:00",
        "endDateTime": "2018-08-09T16:00"
      },
      "status": "pending"
      -- Tutaj znajduje się pozostała reprezentacja zasobu zgłoszenia
    }
  }
}

```

3a. Potwierdzenie przez OA zmiany terminu. OA potwierdza termin wykonując PATCH na zgłoszeniu ze slotem przesyłanym w notyfikacji.

```

PATCH /tmf/troubleTicketManagement/v1/troubleTicket/3180
Content-Type: application/merge-patch+json; charset=UTF-8
If-Match: e6c19694e0a94702c50ea93d0455ec83

{
  "id": "3180",
  "appointmentTimeSlot" {
    "@type": "TimeSlot",
    "startDateTime": "2018-08-09T14:00",
    "endDateTime": "2018-08-09T16:00"
  },
  "@type": "AssistTicket"
}

```

4. OA przesyła wniosek o zmianę terminu.

```

PATCH /tmf/troubleTicketManagement/v1/troubleTicket/3180
Content-Type: application/merge-patch+json; charset=UTF-8
If-Match: e6c19694e0a94702c50ea93d0455ec83

{
  "id": "3180",
  "appointmentTimeSlot" {
    "@type": "TimeSlot",
    "startDateTime": "2018-08-09T12:00",
    "endDateTime": "2018-08-09T14:00"
  },
  "@type": "AssistTicket"
}

```

5. OA tworzy zadanie anulowania zgłoszenia asysty.

```

POST
https://host:port/tmf/troubleTicketManagement/v1/cancelAssistTicketTask
Content-Type: application/json; charset=UTF-8

{
  "@type": "CancelAssistTicketTask",
  "description": "Opis powodu anulowania", #dodatkowy opis
  "assistTicket": {
    id: "123456",
    "@referredType": "AssistTicket"
  }
}

```

OPL przekazuje informacje o zakończeniu zadania anulowania zgłoszenia asysty.

```

POST https://host:port/notifications
Content-Type: application/json; charset=UTF-8

{
  "eventId": "00001",
  "eventTime": "2015-11-16T16:42:25-04:00",
  "eventType": "CancelAssistTicketTaskStateChangeNotification",
  "event": {
    "cancelAssistTicketTask": {
      "@type": "CancelAssistTicketTask",
      "exitCode": "1", #kod ze słownika
      "exitCodeDescription": "opis ze słownika FAULT-CANCEL-REASON ",
      "state": "done",
      -- Tutaj znajduje się pozostała reprezentacja zasobu zadania
    }
  }
}

```

OPL przekazuje informacje zmianie stanu zasobu asysty technicznej spowodowanego zadaniem anulowania asysty.

```

POST https://host:port/notifications
Content-Type: application/json; charset=UTF-8

{
  "eventId": "00001",
  "eventTime": "2015-11-16T16:42:25-04:00",
  "eventType": "TroubleTicketStatusChangeNotification",
  "event": {
    "troubleTicket": {
      "@type": "AssistTicket",
      "additionalState": {
        "code": "15001",
        "@type": "Cancel",
        "description": "swobodny opis powodu odrzucenia skopiowany
z zadania"
      },
      "status": "cancelled",
      -- Tutaj znajduje się pozostała reprezentacja zasobu zadania
    }
  }
}

```

6. OPL powiadamia OA o konieczności usunięcia usterki.

```

POST https://host:port/notifications
Content-Type: application/json; charset=UTF-8

{
  "eventId": "00001",
  "eventTime": "2015-11-16T16:42:25-04:00",
  "eventType": "TroubleTicketStatusChangeNotification",
  "event": {
    "troubleTicket": {
      "@type": "AssistTicket",
      "requestedResolutionDate": "2015-11-16T00:00:00",
      "expectedResolutionDate": "2015-11-16T00:00:00",
      "status": "held",
      -- Tutaj znajduje się pozostała reprezentacja zasobu zgłoszenia
    }
  }
}

```

OPL powiadamia OA o spodziewanej dacie usunięcia uszkodzenia.

```

POST https://host:port/notifications
Content-Type: application/json; charset=UTF-8

{
  "eventId": "00001",
  "eventTime": "2015-11-16T16:42:25-04:00",
  "eventType": "TroubleTicketChangeNotification",
  "event": {
    "troubleTicket": {
      "@type": "AssistTicket",
      "id": "3180",
      "expectedResolutionDate": "2015-11-17T00:00:00"
    }
  }
}

```

6a. OPL powiadamia o pozytywnym zakończeniu uszkodzenia.

```

POST https://host:port/notifications
Content-Type: application/json; charset=UTF-8

{
  "eventId": "00001",
  "eventTime": "2015-11-16T16:42:25-04:00",
  "eventType": "TroubleTicketStatusChangeNotification",
  "event": {
    "troubleTicket": {
      "@type": "AssistTicket",
      "resolutionDate": "2015-11-17T00:00:00",
      "status": "inprogress",
      -- Tutaj znajduje się pozostała reprezentacja zasobu zgłoszenia
    }
  }
}

```

7. Przesłanie informacji o skutecznej realizacji asysty.

```
POST https://host:port/notifications
Content-Type: application/json; charset=UTF-8

{
  "eventId": "00001",
  "eventTime": "2015-11-16T16:42:25-04:00",
  "eventType": "TroubleTicketResolvedNotification",
  "event": {
    "troubleTicket": {
      "@type": "AssistTicket",
      "additionalState": {
        "@type": "Exit",
        "description": "opis kodu wyjścia ze słownika EC-FIA",
        "code": "13004", # ze słownika EC-FIA
      },
      "resolutionDate": "2019-01-30T08:00:00",
      "status": "resolved",
      "statusChangeReason": "Zgłoszenie rozwiązane",
      --Tutaj znajduje się pozostała reprezentacja zasobu zgłoszenia
    }
  }
}
```

8. Przesłanie informacji o zamknięciu zgłoszenia asysty.

```
POST https://host:port/notifications
Content-Type: application/json; charset=UTF-8

{
  "eventId": "00001",
  "eventTime": "2015-11-16T16:42:25-04:00",
  "eventType": "TroubleTicketStatusChangeNotification",
  "event": {
    "troubleTicket": {
      "@type": "AssistTicket",
      "additionalState": {
        "@type": "Exit",
        "description": "opis kodu wyjścia ze słownika EC-FIA",
        "code": "13004", # ze słownika EC-FIA
      },
      "statusChangeReason": "Zgłoszenie zakończone",
      "status": "closed",
      --Tutaj znajduje się pozostała reprezentacja zasobu zgłoszenia
    }
  }
}
```

Słowniki

RR-FIA – kody odrzucenia dla uszkodzeń, asysty i awarii masowej od OA

Code	Description	Use
------	-------------	-----

1	Istnieje niezakończone zgłoszenie Uszkodzenia / Asysty dla danej Usługi	Uszkodzenie/Asysta techniczna
2	Istnieje niezakończone zgłoszenie Awarii Masowej dla danej usługi	Uszkodzenie/Asysta techniczna
9	Zgłoszenie Asysty nie spełnia warunków klasyfikujących do Asysty – przekroczony termin	Asysta techniczna
10	Zgłoszenie Asysty nie spełnia warunków klasyfikujących do Asysty – ilość zgłoszeń poprzedzających	Asysta techniczna
11	Zgłoszenie Asysty nie spełnia warunków klasyfikujących do Asysty – niewłaściwe kody wyjścia poprzednich zgłoszeń	Asysta techniczna
20	Błędny termin dostępności do lokalu klienta	Uszkodzenie
21	Istnieje niezakończona praca planowa dla danej usługi	Uszkodzenia/Asysta techniczna
1001	Niepoprawne idPDU	Awaria masowa od OA
1002	Niepoprawny vlan2pdu	Awaria masowa od OA
1003	Mało dokładny opis Awarii	Awaria masowa od OA

EC-FIA – kody wyjścia dla uszkodzeń, reklamacji i asysty

Code	Description	Use
12001	OPL Ok. Jest synchronizacja modemu (sprawdzenie)	Uszkodzenie
12002	Wymiana urządzenia OPL	Uszkodzenie
12003	OPL poprawiono (opis)	Uszkodzenie
12004	Wymiana urządzenia OA	Uszkodzenie
12005	Dewastacja przyłącza/ instalacji u Abonenta	Uszkodzenie
12006	Brak uszkodzenia	Uszkodzenie
12007	Brak Abonenta w lokalu	Uszkodzenie
12008	Uszkodzenie u klienta	Uszkodzenie
12009	Wymiana ONT i CPE	Uszkodzenie
12010	Obniżona prędkość	Uszkodzenie
12011	Wystąpienie "siły wyższej", awarie długotrwałe nie możliwe do usunięcia ze względu na warunki atmosferyczne (powódzie, burze, niskie	Uszkodzenie

	temperatury, konieczność uzyskania zgody zarządu dróg, właściciela posesji itp.)	
12012	Brak możliwości technicznych	Uszkodzenie
12013	Usługa objęta Pracą Planową	Uszkodzenie
12014	Usługa objęta Awarią PDU/Masową	Uszkodzenie
12015	Przywrócona opcja sprzed modyfikacji	Uszkodzenie
12016	Problem z modemem klienta	Uszkodzenie
12017	Linia nie spełnia parametru przepływu	Uszkodzenie
12019	Przywrócona opcja po obniżeniu	Uszkodzenie
12020	Uszkodzenie trudnousewualne	Uszkodzenie
13001	Nieprawidłowość po stronie OA	Asysta techniczna
13002	Nieprawidłowość po stronie OPL	Asysta techniczna
13003	Nieprawidłowość po stronie Abonenta	Asysta techniczna
13004	Podczas Asysty nie stwierdzono nieprawidłowości	Asysta techniczna
13005	Podczas Asysty stwierdzono niezgodność danych konfiguracyjnych usługi	Asysta techniczna
13006	Przedstawiciel OA nie stawił się na Asystę	Asysta techniczna
13007	Przedstawiciel TP nie stawił się na Asystę	Asysta techniczna
13008	Nieprawidłowość po stronie OA oraz po stronie OPL	Asysta techniczna
14001	Rozpatrzone pozytywnie	Reklamacja
14002	Rozpatrzone negatywnie	Reklamacja
14003	Konieczność przesunięcia daty rozpatrzenia reklamacji	Reklamacja
14004	Reklamacja częściowo uznana	Reklamacja
14005	OPL nie jest stroną	Reklamacja

FAULT-CANCEL-REASON – kody anulowania uszkodzeń/asyst kierowane do operatora

Code	Description
1	Anulowanie zgłoszenia wykonane
2	Anulowanie zgłoszenia nie jest możliwe na tym etapie

CANCEL-REASON-CODE – kody anulowania uszkodzeń/asyst podawane przez operatora

Code	Description
15001	Klient odwołał uszkodzenie
15002	Klient zrezygnował z usługi

COMPLAINT-SUBJECT – przedmiot reklamacji

Code	Description
2	Negatywny wynik weryfikacji technicznej
3	Brak/ niewłaściwa realizacja zamówienia,
4	Odmowa realizacji zamówienia przez TP,
5	Brak anulowania zamówienia przez TP
7	Abonent reklamuje brak aktywnej usługi
8	Abonent reklamuje konieczność płacenia abonamentu w związku z czasowym dostępem do usług
9	Usługa zrealizowana w innym terminie niż wymagana data usługi
10	Abonent uważa, iż sposób wykonywania usługi telekomunikacyjnej jest niewłaściwy
11	Abonent reklamuje czas i wystąpienia połączeń
12	Abonent kwestionuje opłatę za usługę, z której nie korzystał
13	Podwójne fakturowanie
14	Brak formalnego uzasadnienia nierozpoczęcia świadczenia usługi
15	Nieskuteczne procesowanie rozwiązania awarii po stronie TP
16	Przerwa w działaniu usługi

SYMPTOMS – symptom uszkodzenia

Code	Description	Use
11001	ZAKOŃCZENIE SIECIOWE - NIE DZIAŁA/ŹLE DZIAŁA	Uszkodzenie
11002	USZKODZENIE SIECI INSTALACYJNEJ	Uszkodzenie
11003	DEKODER - BRAK KOMUNIKACJI Z MODEMEM	Uszkodzenie
11004	BRAK SYGNAŁU OPTYCZNEGO do ONT	Uszkodzenie
11005	ONT - NIE DZIAŁA/ ŹLE DZIAŁA	Uszkodzenie
11006	BRAK STABILNOŚCI POŁĄCZENIA, KRÓTKIE SESJE	Uszkodzenie

11007	NISKI TRANSFER	Uszkodzenie
11008	BRAK SYNCHRONIZACJI	Uszkodzenie
11009	BRAK POŁĄCZENIA JEST SYNCHRONIZACJA	Uszkodzenie
11010	USZKODZENIE PO MIGRACJI	Uszkodzenie
20001	BRAK DZIAŁANIA USŁUG	Awaria masowa od OA
20002	SKOKI MOCY OPTYCZNEJ	Awaria masowa od OA
20003	SPADEK/NIESTABILNE SESJE	Awaria masowa od OA
20004	BRAK SESJI	Awaria masowa od OA
20005	ZANIK TRANSMISJI	Awaria masowa od OA
20006	ZANIK KOMUNIKACJI	Awaria masowa od OA
20007	ALARMY LOS	Awaria masowa od OA