

Projekt z użyciem SOAP WS
Temat: System rezerwacji biletów kolejowych



NATIONAL TRAINZZ
Reservation system provided by TrainsOnline

Authors:
Adam Bajguz
Michał Kierzkowski

Białystok, 4.05.2020

Wykonujący:
Adam Bajguz
Michał Kierzkowski
Grupa PS3

Prowadzący:
dr hab. inż. Jacek Grekow

Spis treści

1. Wprowadzanie.....	3
1.1. Zakres projektowy	3
1.2. Funkcjonalności	3
1.3. Definicje, Akronimy, Skróty.....	4
1.4. Użyte technologie	6
1.4.1. Po stronie serwera.....	6
1.4.2. Po stronie klienta	10
2. Specyfikacja usługi Web Service	11
2.1. Standardy sieciowe	11
2.2. Adres usługi	11
2.3. Specyfikacja WSDL	11
2.4. Dostępne usługi	17
2.5. Opis wybranych usług	18
2.5.1. Operacja tworzenie biletu.....	18
2.5.2. Operacja pobranie listy biletów danego klienta.....	19
2.5.3. Operacja szczegóły stacji.....	20
2.5.4. Operacja usunięcie stacji	21
2.5.5. Operacja zmiana hasła	22
3. Instrukcja użytkownika aplikacji klienckiej	23

1.Wprowadzanie

Celem tego dokumentu jest przedstawienie funkcjonalności, technologii oraz zasad działania(w tym wymiany informacji) w usłudze sieciowej systemu rezerwacji biletów kolejowych oraz interfejsowym systemie klienckim.

1.1.Zakres projektowy

Zakresem projektu było stworzenie systemu, który pozwoli na przechowywanie bazy stacji kolejowych i tras, na jakich przemieszczają się pociągi, a także umożliwi zakup biletów na daną trasę oraz pobranie go w formacie PDF. W tym celu należało stworzyć dwie aplikacje jedną, jako Web Service dostarczający wszystkie funkcjonalności w API, zaś drugą, jako aplikację kliencką umożliwiającą łatwy dostęp do zasobów serwisowych.

1.2.Funkcjonalności

System posiada funkcjonalności podstawowe takie jak:

- rejestracja klienta;
- logowanie na serwer;
- wyświetlanie bazy tras kolejowych (Miasto od , Miasto do, dzień, godzina);
- wyszukiwarkę tras;
- kupno biletu;
- odbiór potwierdzenia kupna w formacie PDF;
- sprawdzenie rezerwacji na podstawie podanego numeru biletu;
- dodawanie, usuwanie, modyfikowanie tras oraz stacji (funkcjonalność dostępna na Swagger lub SOAP UI tylko dla użytkowników z prawami administratora)

a także funkcjonalności dodatkowe:

- przypominanie hasła na podany podczas rejestracji adres email;
- szczegóły danej stacji kolejowej;
- prezentowanie na mapie położenia stacji;
- autoryzacja oparta na JSON Web Token;
- generowanie kodów QR które służą do dodania wydarzenia do kalendarza o planie podróży oraz weryfikacji biletu
- dziennik logów zapisywany wewnętrznie na serwerze przy pomocy loggera Serilog, a także możliwość zdalnego przechowywania logów na platformie Sentry
- wykorzystanie algorytmu Jaro-Winkler podczas wyszukiwania

1.3. Definicje, Akronimy, Skróty

Simple Object Access Protocol (SOAP) – jest protokołem bazującym na standardzie XML. Protokół SOAP pozwala aplikacjom na komunikację przez Internet. Standard SOAP jest opracowywany przez W3C

The World Wide Web Consortium (W3C) – organizacja zajmująca się ustanawianiem standardów dla stron WWW. Publikowane przez W3C rekomendacje nie mają mocy prawnej, nakazującej ich użycie, lecz wskazują standardy dla rozwiązań technologicznych.

Web Service (WS) – podstawowa technologia wykorzystywana w architekturze SOA. Jest to usługa systemu informatycznego, która może być używana przez inne komponenty programowe, również komponenty zewnętrzne projektowanego systemu. Web Service w paradygmacie SOA stanowi najistotniejszy interfejs (ang. API – applicationprogramminginterface) projektowanego systemu. Web Service jest również wykorzystywany wewnętrznie do komunikacji pomiędzy komponentami systemu.

Web Services Description Language (WSDL) – plik WSDL to dokument XML, opisujący zbiór komunikatów SOAP.

ExtensibleMarkup Language (XML) – to uniwersalny język formalny przeznaczony do reprezentowania różnych danych w ustrukturalizowany sposób. XML jest niezależny od platformy, co umożliwia łatwą wymianę dokumentów pomiędzy różnymi systemami i rekomendowany oraz specyfikowany przez organizację W3C.

Secure Sockets Layer (SSL) – jest protokołem do zabezpieczania komunikacji odbywającej się w Internecie. Dzięki niemu, wszystkie informacje wysyłane pomiędzy przeglądarką internetową a serwerem są szyfrowane. Aby z niego skorzystać, wymagana jest instalacja certyfikatu SSL.

HTTP Strict Transport Security (HSTS) – mechanizm bezpieczeństwa sieci, który chroni strony przed atakami takimi, jak wymuszone zmniejszenie poziomu protokołu oraz przechwytywanie sesji. Dzięki niemu do serwerów można połączyć się tylko za pomocą przeglądarek, korzystających z bezpiecznych połączeń HTTPS, natomiast nigdy nie dopuszcza on połączeń na bazie niezabezpieczonego protokołu HTTP. HSTS jest uznawany za standardowy protokół przez IETF..

JSON Web Token - to internetowy standard do tworzenia, tokenów dostępu opartych na JSON. Tokeny są podpisywane przy użyciu prywatnego tajnego klucza lub klucza publicznego / prywatnego. Na przykład serwer może wygenerować token z roszczeniem „zalogowany, jako administrator” i przekazać go klientowi. Klient może następnie użyć tego tokena, aby udowodnić, że jest zalogowany, jako administrator.

Model-view-viewmodel (MVVM) to wzorzec architektoniczny oprogramowania, który ułatwia oddzielenie rozwoju graficznego interfejsu użytkownika (*widok*) - czy to poprzez język znaczników lub kod GUI - od rozwoju logiki biznesowej lub back- logika końcowa (*model*), dzięki czemu widok nie jest zależny od żadnej konkretnej platformy modelu.

Universal Windows Platform (UWP) – interfejs API stworzony przez Microsoft i po raz pierwszy użyty w systemie Windows 10. Jego celem jest ułatwienie pisania aplikacji uniwersalnych, działających zarówno na komputerach, jak i telefonach, konsolach Xbox One oraz okularach HoloLens bez potrzeby ich przepisywania dla każdego rodzaju urządzeń.

Globally unique identifier (GUID), identyfikator globalnie unikatowy – identyfikator obiektów między innymi w systemie Windows lub wszędzie, gdzie potrzebny jest unikatowy identyfikator.

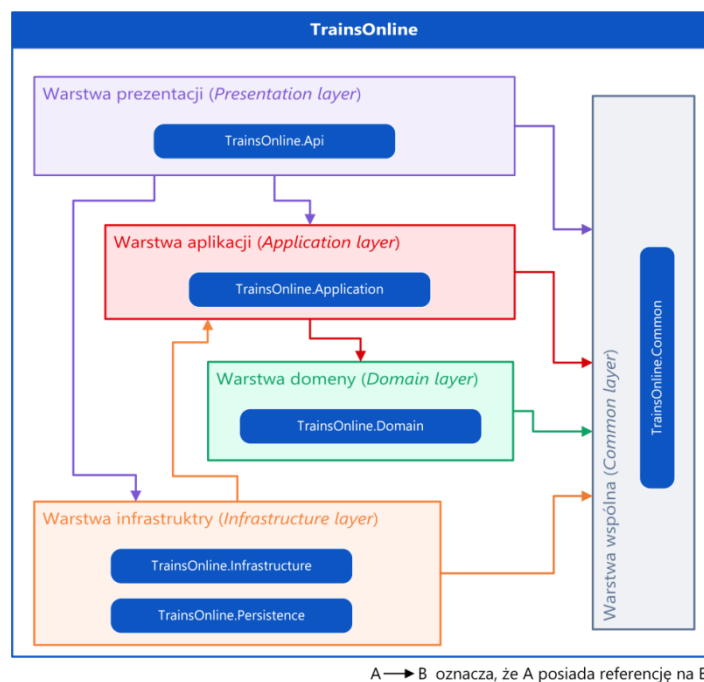
1.4. Użyte technologie

1.4.1. Po stronie serwera

Moduł TrainsOnline został zaimplementowany z użyciem podejścia określonego jako Clean Architecture, którego celem jest zapewnienie wysokiej skalowalności aplikacji i uniknięcie tworzenia monolitycznej architektury. Podejście to zrealizowano przy pomocy wzorca architektonicznego Domain Driven Design (DDD). Do stworzenia serwera aplikacji użyto: C# 8.0, NET Core 3.1.0, ASP.NET Core MVC, Microsoft SQL Server 2017. W implementacji zastosowano także następujące biblioteki:

Nazwa	Wersja
AutoMapper.Extensions.Microsoft.DependencyInjection	7.0.0
CommandLineParser	2.8.0
CSharpVitamins.ShortGuid	1.0.1
FluentValidation.AspNetCore	8.6.2
F23.StringSimilarity	3.1.0
GemBox.Document	31.0.1175
System.Drawing.Common	5.0.0-preview.3.20214.6
SoapCore	1.1.0-alpha
QRCoder	1.3.9
MediatR	8.0.1
MediatR.Extensions.Microsoft.DependencyInjection	8.0.0
Microsoft.AspNetCore.Authentication.JwtBearer	3.1.3
Microsoft.AspNetCore.Mvc	2.2.0
Microsoft.EntityFrameworkCore	3.1.3
Microsoft.EntityFrameworkCore.Design	3.1.3
Microsoft.EntityFrameworkCore.InMemory	3.1.
Microsoft.EntityFrameworkCore.SqlServer	3.1.3
Microsoft.EntityFrameworkCore.Tools	3.1.3
Microsoft.Extensions.CommandLineUtils	1.1.1
Microsoft.Extensions.Configuration.EnvironmentVariables	3.1.0
Microsoft.Extensions.Configuration.Json	3.1.0
Microsoft.NET.Test.Sdk	16.4.0
Sentry.Serilog	2.1.1
Serilog.AspNetCore	3.2.0
Serilog.Enrichers.Memory	1.0.4
Serilog.Enrichers.Process	2.0.1
Serilog.Enrichers.Thread	3.2.0-dev-00747
Serilog.Exceptions	5.4.0
Serilog.Sinks.Async	1.4.0
Serilog.Sinks.Console	3.1.1
Serilog.Sinks.File	4.1.0
Shouldly	3.0.2
Swashbuckle.AspNetCore	5.4.1

Nazwa	Wersja
Swashbuckle.AspNetCore.Newtonsoft	5.4.1
Swashbuckle.AspNetCore.ReDoc	5.4.1
Swashbuckle.AspNetCore.Annotations	5.4.1
System.IdentityModel.Tokens.Jwt	5.6.0
xunit	2.4.1
xunit.runner.visualstudio	2.4.1



Wzorec DDD wymaga również podziału projektu na następujące warstwy:

- wspólną – obejmuje wszystkie kwestie przekrojowe, czyli klasy i interfejsy wspólne dla różnych warstw;
- prezentacji – zawiera kontrolery, punkty dostępowe do REST API oraz aplikację Swagger;
- aplikacji – zawiera całą logikę. Warstwa jest zależna wyłącznie od warstwy domeny, są w niej zdefiniowane interfejsy, np. repozytorium lub jednostki pracy (Unit of Work), których implementacja znajduje się w innych warstwach. Oznacza to, że DDD wykorzystuje wzorec odwrócenia sterowania (inversion of control). Przykładowo, jeśli aplikacja musi uzyskać dostęp do usługi powiadomień, do warstwy aplikacji zostanie dodany nowy interfejs, a implementacja zostanie utworzona w ramach warstwy infrastruktury;
- domeny – zawiera dane, wyliczenia, wyjątki i logikę specyficzną dla domeny;
- infrastruktury – zawiera klasy dostępu do zasobów zewnętrznych, takich jak: baza danych, systemy plików i poczta elektroniczna. Klasy te opierają się na interfejsach zdefiniowanych w warstwie aplikacji. W warstwie znajduje się również podwarstwa Persistence, w której zawarta jest cała konfiguracja bazy danych.

W projekcie z uwagi na małą liczbę obiektów domenowych, a tym samym tabel w bazie danych, zastosowano podejście grupowania klas w:

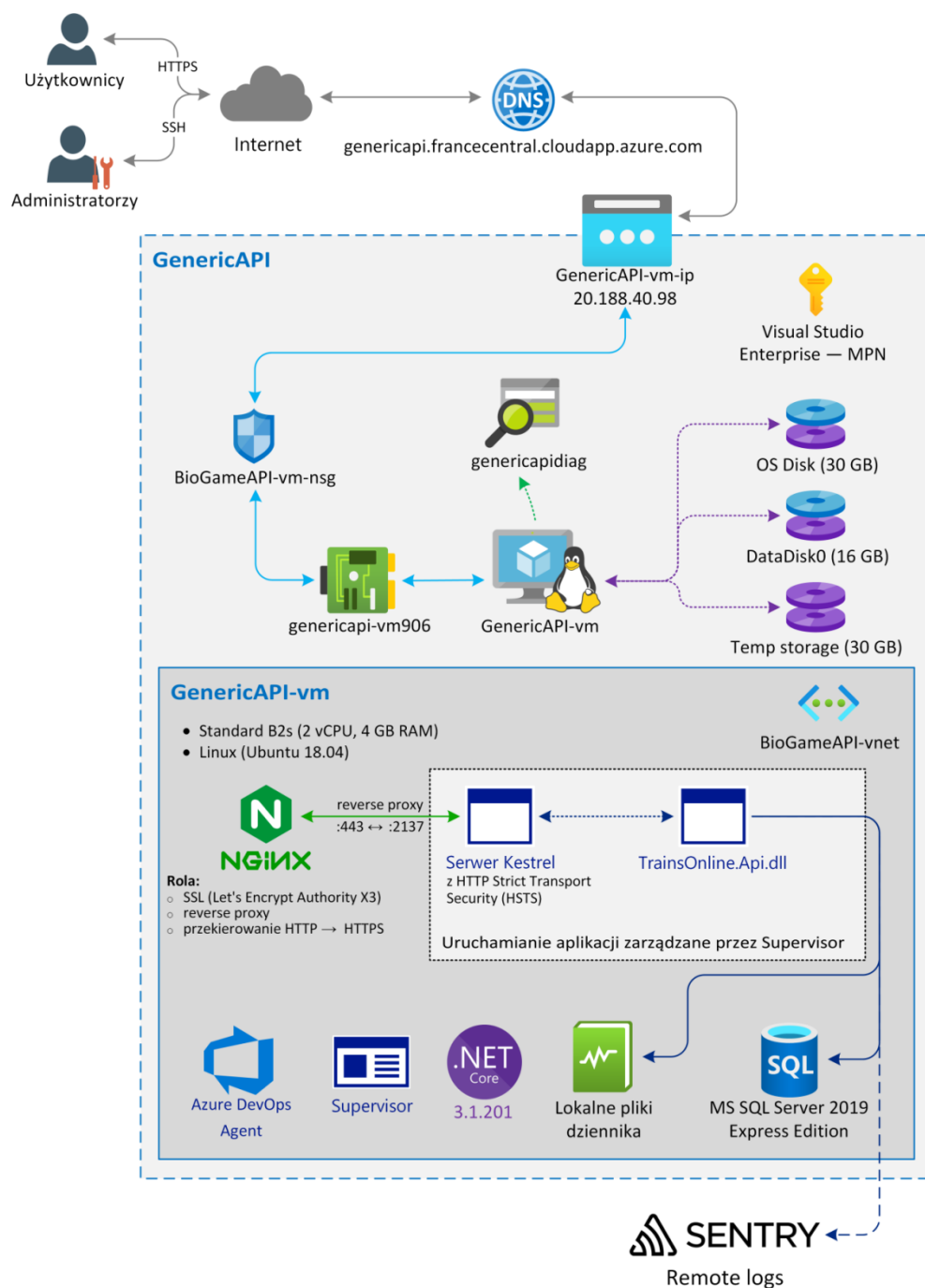
- i. projekty (biblioteki) w oparciu o warstwy określone przez DDD (package by layer),
- ii. funkcjonalności (package by feature) na poziomie folderów i przestrzeni nazw w poszczególnych warstwach.

Rozwiązanie te jest wystarczające do zapewnienia czystego kodu oraz architektury w przypadku tego modułu. Ponadto nie wymaga implementowania m.in. koncepcji ograniczonych kontekstów (bounded contexts).

Moduł TrainsOnline został wdrożony na platformie Azure na maszynie wirtualnej typu Standard B2s, wyposażonej w dwa wirtualne procesory oraz 4 GB pamięci operacyjnej. Do maszyny podpięte są dwa dyski SSD typu Premium SSD, charakteryzujące się maksymalną liczbą operacji na sekundę (IOPS) równą 120 i przepływnością maksymalną na poziomie 25 MB/s, oraz jeden dysk tymczasowy o maksymalnej wartości IOPS równej 1600 i maksymalnej przepływności 15 MB/s :

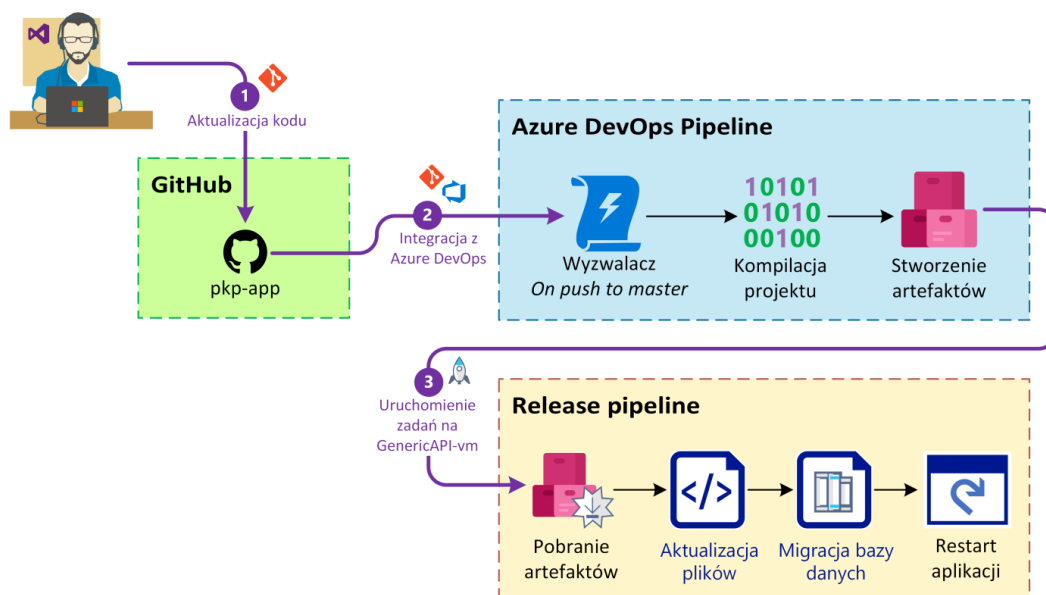
- OS Disk – dysk systemowy o pojemności 30 GB z systemem operacyjnym Ubuntu w wersji 18.04 oraz aplikacjami: .NET Core 3.1.0 Runtime, Microsoft SQL Server 2017 Express Edition, nginx, Azure DevOps Agent, supervisor;
- DataDisk0 – dysk o pojemności 16 GB zawierający aplikację "" oraz pliki dziennika aplikacji, tzw. logi;
- Temp storage – dysk o pojemności 30 GB służący do przechowywania danych tymczasowych, np. plików tymczasowych serwera SQL, który ulega wyczyszczeniu m.in. po ponownym uruchomieniu maszyny wirtualnej.

Zdalny dostęp do maszyny wirtualnej odbywa się poprzez protokół Secure Shell (SSH) z wykorzystaniem klucza RSA jako metody uwierzytelniania. Do połączenia się z maszyną wirtualną wykorzystano program Termius z pakietu GitHub Student Developer Pack. Dostarcza on wieloplatformowy terminal ze zintegrowanym klientem SSH. Dostęp przez użytkowników (klientów) do API odbywa się za pośrednictwem protokołu Hypertext Transfer ProtocolSecure (HTTPS). Działanie aplikacji można sprawdzić pod adresem <https://genericapi.francecentral.cloudapp.azure.com/soap-api>. Certyfikat na potrzeby protokołu HTTPS uzyskano za pomocą programu Certbot, urzędem certyfikacji jest Let'sEncrypt Authority X3. Aplikacja TrainsOnline uruchomiona jest na porcie 2137 za pomocą serwera Kestrel, który został użyty jako serwer graniczny. Komunikacja aplikacji z Internetem odbywa się za pomocą serwera nginx skonfigurowanego jako reverseproxy pomiędzy portami o numerach 443 oraz 2137. Aplikacja komunikuje się również z platformą Sentry, na której zapisywane są wszelkie zdarzenia błędów.



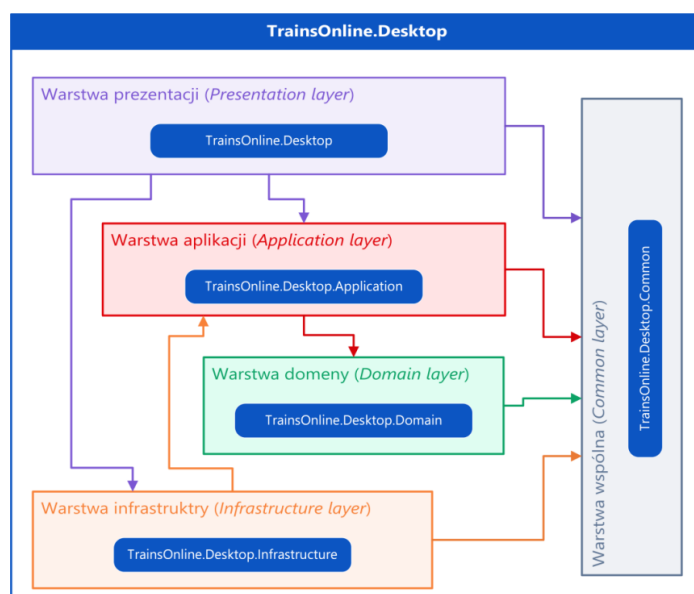
Zastosowanie ciągłej integracji (continuousintegration) oraz ciągłego dostarczania (continuousdelivery) umożliwiło łatwiejszą obsługę środowiska produkcyjnego TrainsOnline. Aktualizacje plików wykonywalnych, a także modyfikacje struktury bazy danych (migracje) odbywają się przy pomocy dwóch potoków:

1. uruchamianego na AzureDevOps w momencie pojawienia się zmian w gałęzi master repozytorium pkp-app, który buduje aplikację oraz przygotowuje paczkę z artefaktami;
2. uruchamianego na maszynie wirtualnej z poziomu AzureDevOps (z udziałem programu AzureDevOps Agent), którego zadaniem jest m.in. pobranie paczki artefaktów i aktualizacja plików.



1.4.2. Po stronie klienta

Aplikacja TrainsOnline.Desktop jest aplikacją UWP zawierającą całą logikę niezbędną do komunikacji z serwerem rezerwacji biletów kolejowych. Technologia UWP została zastosowana w celu implementacji łatwego i przyjaznego dla użytkownika interfejsu okienkowego o spójnym wyglądzie. Architektura aplikacji wykorzystuje wzorec MVVM w celu zapewnienia oddzielenia interfejsu użytkownika od logiki biznesowej. Do implementacji MVVM zastosowano bibliotekę Caliburn.Micro. Ponadto w implementacji zastosowano podejście DDD – podobnie jak w aplikacji serwerowej. Jediną różnicą jest brak warstwy Persistence.



A → B oznacza, że A posiada referencję na B

2. Specyfikacja usługi Web Service

2.1. Standardy sieciowe

Usługa „TrainsOnline” zaimplementowana została jako usługa sieciowa (Web Service) z użyciem protokołu SOAP. Usługa dostępna jest poprzez protokół HTTPS . Usługa dostępna jest pod adresem:

<https://genericapi.francecentral.cloudapp.azure.com/soap-api>, który prowadzi do strony zawierającej wszelkie dostępne na serwerze adresy, w tym adresy serwisów i ich opisów za pomocą WSDL.

Backend Api for TrainsOnline
© Adam Bajguz & Michał Kierzkowski 2020

Links:

OpenAPI specification can be found at </api/v1/swagger.json>

Swagger can be accessed through </api/index.html>

ReDoc can be accessed through </api/redoc/index.html>

REST API base url is </api>

SOAP API base url and docs </soap-api>

App health can be checked under </health>

Soap Services Endpoints List

Endpoint	Name	Description
/soap-api/authentication	Authentication	User authentication and password reset
/soap-api/route	Route	Create, update, and get route
/soap-api/station	Station	Create, update, and get station
/soap-api/ticket	Ticket	Create, update, and get ticket
/soap-api/user	User	Create, update, and get user

2.2. Adres usługi

<https://genericapi.francecentral.cloudapp.azure.com/soap-api>

2.3. Specyfikacja WSDL

WSDL serwisu obsługującego autoryzację ma 145 linii, serwisu tras kolejowych - 263, serwisu stacji kolejowych - 259, serwisu biletów - 256, a serwisu użytkowników – 314. Z tego powodu w niniejszym dokumencie przedstawiono wyłącznie przykłady okrojonych okrojone WSDL'izawierające tylko jedną metodę na raz.

- Przykładowy WSDL „Tworzenia biletu”:

```

<?xml version="1.0" encoding="UTF-8"?>
<wsdl:definitions xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/" xmlns:tns="http://tempuri.org/" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:http="http://schemas.microsoft.com/ws/06/2004/policy/http" xmlns:msc="http://schemas.microsoft.com/ws/2005/12/wsdl/contract" xmlns:wsp="http://schemas.xmlsoap.org/ws/2004/09/policy"
  xmlns:wssu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd" xmlns:wsam="http://www.w3.org/2007/05/addressing/metadata"
  xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/" targetNamespace="http://tempuri.org/" name="ITicketSoapEndpointService">
  <wsdl:types>
    <xsd:schema elementFormDefault="qualified" targetNamespace="http://tempuri.org/">
      <xsd:import namespace="http://schemas.microsoft.com/2003/10/Serialization/Arrays"/>
      <xsd:import namespace="http://schemas.datacontract.org/2004/07/System"/>
      <xsd:element name="CreateTicket">
        <xsd:complexType base="xsd:anyType" substitutionGroup="xsd:anyType">
          <xsd:sequence base="xsd:anyType" substitutionGroup="xsd:anyType">
            <xsd:element minOccurs="0" maxOccurs="1" name="ticket" type="tns:CreateTicketRequest"/>
          </xsd:sequence>
        </xsd:complexType>
      </xsd:element>
      <xsd:element name="CreateTicketResponse">
        <xsd:complexType base="xsd:anyType" substitutionGroup="xsd:anyType">
          <xsd:sequence base="xsd:anyType" substitutionGroup="xsd:anyType">
            <xsd:element minOccurs="0" maxOccurs="1" name="CreateTicketResult" type="tns:IdResponse"/>
          </xsd:sequence>
        </xsd:complexType>
      </xsd:element>
      <xsd:complexType name="CreateTicketRequest">
        <xsd:sequence base="xsd:anyType" substitutionGroup="xsd:anyType">
          <xsd:element minOccurs="1" maxOccurs="1" name="UserId" type="xsd:string"/>
          <xsd:element minOccurs="1" maxOccurs="1" name="RouteId" type="xsd:string"/>
        </xsd:sequence>
      </xsd:complexType>
      <xsd:complexType name="IdResponse">
        <xsd:sequence base="xsd:anyType" substitutionGroup="xsd:anyType">
          <xsd:element minOccurs="1" maxOccurs="1" name="Id" type="xsd:string"/>
        </xsd:sequence>
      </xsd:complexType>
    </xsd:schema>
  </wsdl:types>
  <wsdl:message name="ITicketSoapEndpointService_CreateTicket_InputMessage">
    <wsdl:part name="parameters" element="tns:CreateTicket"/>
  </wsdl:message>
  <wsdl:message name="ITicketSoapEndpointService_CreateTicket_OutputMessage">
    <wsdl:part name="parameters" element="tns:CreateTicketResponse"/>
  </wsdl:message>
  <wsdl:portType name="ITicketSoapEndpointService">
    <wsdl:operation name="CreateTicket">
      <wsdl:input message="tns:ITicketSoapEndpointService_CreateTicket_InputMessage"/>
      <wsdl:output message="tns:ITicketSoapEndpointService_CreateTicket_OutputMessage"/>
    </wsdl:operation>
  </wsdl:portType>
  <wsdl:binding name="BasicHttpsBinding" type="tns:ITicketSoapEndpointService">
    <soap:binding transport="http://schemas.xmlsoap.org/soap/http"/>
    <wsdl:operation name="CreateTicket">
      <soap:operation soapAction="http://tempuri.org/ITicketSoapEndpointService/CreateTicket" style="document"/>
      <wsdl:input>
        <soap:body use="literal"/>
      </wsdl:input>
      <wsdl:output>
        <soap:body use="literal"/>
      </wsdl:output>
    </wsdl:operation>
  </wsdl:binding>
  <wsdl:service name="ITicketSoapEndpointService">
    <wsdl:port name="BasicHttpsBinding" binding="tns:BasicHttpsBinding">
      <soap:address location="http://localhost:2138/soap-api/ticket"/>
    </wsdl:port>
  </wsdl:service>
</wsdl:definitions>

```

- Przykładowy WSDL „Pobrania listy biletów”:

This XML file does not appear to have any style information associated with it. The document tree is shown below.

```
<?xml version="1.0" encoding="utf-8"?>
<wsdl:definitions xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/" xmlns:tns="http://tempuri.org/" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:http="http://schemas.microsoft.com/ws/2004/01/policy/http" xmlns:wsa="http://schemas.xmlsoap.org/ws/2004/08/addressing" xmlns:wsu="http://schemas.xmlsoap.org/ws/2004/09/policy"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:soap12="http://schemas.xmlsoap.org/wsdl/soap-12/" xmlns:wsdl12="http://schemas.xmlsoap.org/wsdl/2003/05/"
  targetNamespace="http://tempuri.org/" name="ITicketSoapEndpointService">
  <wsdl:types>
    <xsd:schema elementFormDefault="qualified" targetNamespace="http://tempuri.org/">
      <xsd:import namespace="http://schemas.microsoft.com/2003/10/Serialization/Arrays"/>
      <xsd:import namespace="http://schemas.datacontract.org/2004/07/System"/>
      <xsd:element name="GetCurrentUserTicketsList">
        <xsd:complexType>
          <xsd:sequence>
            <xsd:element minOccurs="0" maxOccurs="1" name="GetCurrentUserTicketsListResult" type="tns:GetUserTicketsListResponse"/>
          </xsd:sequence>
        </xsd:complexType>
      </xsd:element>
      <xsd:element name="GetCurrentUserTicketsListResponse">
        <xsd:complexType>
          <xsd:sequence>
            <xsd:element minOccurs="0" maxOccurs="1" name="Tickets" nillable="true" type="tns:ArrayOfUserTicketLookupModel"/>
          </xsd:sequence>
        </xsd:complexType>
      </xsd:element>
      <xsd:complexType name="ArrayOfUserTicketLookupModel">
        <xsd:sequence>
          <xsd:element minOccurs="0" maxOccurs="unbounded" nillable="true" name="UserTicketLookupModel" type="tns:UserTicketLookupModel"/>
        </xsd:sequence>
      </xsd:complexType>
      <xsd:complexType name="UserTicketLookupModel">
        <xsd:sequence>
          <xsd:element minOccurs="1" maxOccurs="1" name="Id" type="xsd:string"/>
          <xsd:element minOccurs="1" maxOccurs="1" name="UserId" type="xsd:string"/>
          <xsd:element minOccurs="0" maxOccurs="1" name="Route" type="tns:UserTicketRouteLookupModel"/>
        </xsd:sequence>
      </xsd:complexType>
      <xsd:complexType name="UserTicketRouteLookupModel">
        <xsd:sequence>
          <xsd:element minOccurs="1" maxOccurs="1" name="Id" type="xsd:string"/>
          <xsd:element minOccurs="0" maxOccurs="1" name="From" type="tns:UserTicketRouteStationLookupModel"/>
          <xsd:element minOccurs="0" maxOccurs="1" name="To" type="tns:UserTicketRouteStationLookupModel"/>
          <xsd:element minOccurs="1" maxOccurs="1" name="DepartureTime" type="xsd:dateTime"/>
          <xsd:element minOccurs="1" maxOccurs="1" name="Duration" type="xsd:duration"/>
          <xsd:element minOccurs="1" maxOccurs="1" name="Distance" type="xsd:double"/>
          <xsd:element minOccurs="1" maxOccurs="1" name="TicketPrice" type="xsd:double"/>
        </xsd:sequence>
      </xsd:complexType>
      <xsd:complexType name="UserTicketRouteStationLookupModel">
        <xsd:sequence>
          <xsd:element minOccurs="1" maxOccurs="1" name="Id" type="xsd:string"/>
          <xsd:element minOccurs="0" maxOccurs="1" name="Name" type="xsd:string"/>
          <xsd:element minOccurs="1" maxOccurs="1" name="Latitude" type="xsd:double"/>
          <xsd:element minOccurs="1" maxOccurs="1" name="Longitude" type="xsd:double"/>
        </xsd:sequence>
      </xsd:complexType>
    </xsd:schema>
  </wsdl:types>
  <wsdl:message name="ITicketSoapEndpointService_GetCurrentUserTicketsList_InputMessage">
    <wsdl:part name="parameters" element="tns:GetCurrentUserTicketsList"/>
  </wsdl:message>
  <wsdl:message name="ITicketSoapEndpointService_GetCurrentUserTicketsList_OutputMessage">
    <wsdl:part name="parameters" element="tns:GetCurrentUserTicketsListResponse"/>
  </wsdl:message>
  <wsdl:portType name="ITicketSoapEndpointService">
    <wsdl:operation name="GetCurrentUserTicketsList">
      <wsdl:input message="tns:ITicketSoapEndpointService_GetCurrentUserTicketsList_InputMessage"/>
      <wsdl:output message="tns:ITicketSoapEndpointService_GetCurrentUserTicketsList_OutputMessage"/>
    </wsdl:operation>
  </wsdl:portType>
  <wsdl:binding name="BasicHttpsBinding" type="tns:ITicketSoapEndpointService">
    <soap:binding transport="http://schemas.xmlsoap.org/soap/http"/>
    <wsdl:operation name="GetCurrentUserTicketsList">
      <soap:operation soapAction="http://tempuri.org/ITicketSoapEndpointService/GetCurrentUserTicketsList" style="document"/>
      <wsdl:input>
        <soap:body use="literal"/>
      </wsdl:input>
      <wsdl:output>
        <soap:body use="literal"/>
      </wsdl:output>
    </wsdl:operation>
  </wsdl:binding>
  <wsdl:service name="ITicketSoapEndpointService">
    <wsdl:port name="BasicHttpsBinding" binding="tns:BasicHttpsBinding">
      <soap:address location="http://localhost:2130/soap-api/ticket"/>
    </wsdl:port>
  </wsdl:service>
</wsdl:definitions>
```

- Przykładowy WSDL „Szczegóły trasy”:

This XML file does not appear to have any style information associated with it. The document tree is shown below.

```
<?xml version="1.0" encoding="UTF-8"?>
<wsdl:definitions xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/" xmlns:tns="http://tempuri.org/" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:http="http://schemas.microsoft.com/ws/06/2004/policy/http" xmlns:msc="http://schemas.microsoft.com/ws/2005/12/wsdl/contract" xmlns:uap="http://schemas.xmlsoap.org/ws/2004/09/policy"
  xmlns:wsu="http://docs.oasis-open.org/ws-sse/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd" xmlns:wsam="http://www.w3.org/2007/05/addressing/metadata" xmlns:wsi="http://schemas.xmlsoap.org/wsdl/"
  targetNamespace="http://tempuri.org/" name="IRouteSoapEndpointService">
  <wsdl:types>
    <xsd:schema elementFormDefault="qualified" targetNamespace="http://tempuri.org/">
      <xsd:import namespace="http://schemas.microsoft.com/2001/10/Serialization/Arrays"/>
      <xsd:import namespace="http://schemas.datacontract.org/2004/07/System"/>
      <xsd:element name="GetRouteDetails">
        <xsd:complexType>
          <xsd:sequence>
            <xsd:element minOccurs="0" maxOccurs="1" name="Id" type="tns:IdRequest"/>
          </xsd:sequence>
        </xsd:complexType>
      </xsd:element>
      <xsd:element name="GetRouteDetailsResponse">
        <xsd:complexType>
          <xsd:sequence>
            <xsd:element minOccurs="0" maxOccurs="1" name="GetRouteDetailsResult" type="tns:GetRouteDetailsResponse"/>
          </xsd:sequence>
        </xsd:complexType>
      </xsd:element>
      <xsd:complexType name="IdRequest">
        <xsd:sequence>
          <xsd:element minOccurs="1" maxOccurs="1" name="Id" type="xsd:string"/>
        </xsd:sequence>
      </xsd:complexType>
      <xsd:complexType name="GetRouteDetailsResponse">
        <xsd:sequence>
          <xsd:element minOccurs="1" maxOccurs="1" name="Id" type="xsd:string"/>
          <xsd:element minOccurs="1" maxOccurs="1" name="CreatedOn" type="xsd:dateTime"/>
          <xsd:element nillable="true" minOccurs="1" maxOccurs="1" name="CreatedBy" type="xsd:string"/>
          <xsd:element minOccurs="1" maxOccurs="1" name="LastSavedOn" type="xsd:dateTime"/>
          <xsd:element nillable="true" minOccurs="1" maxOccurs="1" name="LastSavedBy" type="xsd:string"/>
          <xsd:element minOccurs="0" maxOccurs="1" name="From" type="tns:RouteStationLookupModel"/>
          <xsd:element minOccurs="0" maxOccurs="1" name="To" type="tns:RouteStationLookupModel"/>
          <xsd:element minOccurs="1" maxOccurs="1" name="DepartureTime" type="xsd:dateTime"/>
          <xsd:element minOccurs="1" maxOccurs="1" name="Duration" type="xsd:duration"/>
          <xsd:element minOccurs="1" maxOccurs="1" name="Distance" type="xsd:double"/>
          <xsd:element minOccurs="1" maxOccurs="1" name="TicketPrice" type="xsd:double"/>
        </xsd:sequence>
      </xsd:complexType>
      <xsd:complexType name="RouteStationLookupModel">
        <xsd:sequence>
          <xsd:element minOccurs="1" maxOccurs="1" name="Id" type="xsd:string"/>
          <xsd:element minOccurs="0" maxOccurs="1" name="Name" type="xsd:string"/>
          <xsd:element minOccurs="1" maxOccurs="1" name="Latitude" type="xsd:double"/>
          <xsd:element minOccurs="1" maxOccurs="1" name="Longitude" type="xsd:double"/>
        </xsd:sequence>
      </xsd:complexType>
    </xsd:schema>
  </wsdl:types>
  <wsdl:message name="IRouteSoapEndpointService_GetRouteDetails_InputMessage">
    <wsdl:part name="parameters" element="tns:GetRouteDetails"/>
  </wsdl:message>
  <wsdl:message name="IRouteSoapEndpointService_GetRouteDetails_OutputMessage">
    <wsdl:part name="parameters" element="tns:GetRouteDetailsResponse"/>
  </wsdl:message>
  <wsdl:portType name="IRouteSoapEndpointService">
    <wsdl:operation name="GetRouteDetails">
      <wsdl:input message="tns:IRouteSoapEndpointService_GetRouteDetails_InputMessage"/>
      <wsdl:output message="tns:IRouteSoapEndpointService_GetRouteDetails_OutputMessage"/>
    </wsdl:operation>
  </wsdl:portType>
  <wsdl:binding name="BasicHttpsBinding" type="tns:IRouteSoapEndpointService">
    <soap:binding transport="http://schemas.xmlsoap.org/soap/http"/>
    <wsdl:operation name="GetRouteDetails">
      <soap:operation soapAction="http://tempuri.org/IRouteSoapEndpointService/GetRouteDetails" style="document"/>
      <wsdl:input>
        <soap:body use="literal"/>
      </wsdl:input>
      <wsdl:output>
        <soap:body use="literal"/>
      </wsdl:output>
    </wsdl:operation>
  </wsdl:binding>
  <wsdl:service name="IRouteSoapEndpointService">
    <wsdl:port name="BasicHttpsBinding" binding="tns:BasicHttpsBinding">
      <soap:address location="http://localhost:2130/soap-api/route"/>
    </wsdl:port>
  </wsdl:service>
</wsdl:definitions>
```


- Przykładowy WSDL „Zmiana hasła”:

This XML file does not appear to have any style information associated with it. The document tree is shown below.

```
<?xml version="1.0" encoding="utf-8"?>
<wsdl:definitions xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/" xmlns:tns="http://tempuri.org/" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:http="http://schemas.microsoft.com/ws/06/2004/policy/http" xmlns:msc="http://schemas.microsoft.com/ws/2005/12/wsdl/contract" xmlns:wsp="http://schemas.xmlsoap.org/ws/2004/09/policy"
xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd" xmlns:wsam="http://www.w3.org/2007/05/addressing/metadata" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/"
targetNamespace="http://tempuri.org/" name="IUserSoapEndpointService">
  <wsdl:types>
    <xsd:schema elementFormDefault="qualified" targetNamespace="http://tempuri.org/">
      <xsd:import namespace="http://schemas.microsoft.com/2003/10/Serialization/Arrays"/>
      <xsd:import namespace="http://schemas.datacontract.org/2004/07/System"/>
      <xsd:element name="ChangePassword">
        <xsd:complexType>
          <xsd:sequence>
            <xsd:element minOccurs="0" maxOccurs="1" name="user" type="tns:ChangePasswordRequest"/>
          </xsd:sequence>
        </xsd:complexType>
      </xsd:element>
      <xsd:element name="ChangePasswordResponse">
        <xsd:complexType>
          <xsd:sequence>
            <xsd:element minOccurs="0" maxOccurs="1" name="ChangePasswordResult" type="tns:Unit"/>
          </xsd:sequence>
        </xsd:complexType>
      </xsd:element>
      <xsd:complexType name="ChangePasswordRequest">
        <xsd:sequence>
          <xsd:element minOccurs="1" maxOccurs="1" name="UserId" type="xsd:string"/>
          <xsd:element minOccurs="0" maxOccurs="1" name="OldPassword" type="xsd:string"/>
          <xsd:element minOccurs="0" maxOccurs="1" name="NewPassword" type="xsd:string"/>
        </xsd:sequence>
      </xsd:complexType>
      <xsd:complexType name="Unit">
        <xsd:sequence>
          <xsd:element type="xsd:string"/>
        </xsd:sequence>
      </xsd:complexType>
    </xsd:schema>
  </wsdl:types>
  <wsdl:message name="IUserSoapEndpointService_ChangePassword_InputMessage">
    <wsdl:part name="parameters" element="tns:ChangePassword"/>
  </wsdl:message>
  <wsdl:message name="IUserSoapEndpointService_ChangePassword_OutputMessage">
    <wsdl:part name="parameters" element="tns:ChangePasswordResponse"/>
  </wsdl:message>
  <wsdl:portType name="IUserSoapEndpointService">
    <wsdl:operation name="ChangePassword">
      <wsdl:input message="tns:IUserSoapEndpointService_ChangePassword_InputMessage"/>
      <wsdl:output message="tns:IUserSoapEndpointService_ChangePassword_OutputMessage"/>
    </wsdl:operation>
  </wsdl:portType>
  <wsdl:binding name="BasicHttpsBinding" type="tns:IUserSoapEndpointService">
    <soap:binding transport="http://schemas.xmlsoap.org/soap/http"/>
    <wsdl:operation name="ChangePassword">
      <soap:operation soapAction="http://tempuri.org/IUserSoapEndpointService/ChangePassword" style="document"/>
      <wsdl:input>
        <soap:body use="literal"/>
      </wsdl:input>
      <wsdl:output>
        <soap:body use="literal"/>
      </wsdl:output>
    </wsdl:operation>
  </wsdl:binding>
  <wsdl:service name="IUserSoapEndpointService">
    <wsdl:port name="BasicHttpsBinding" binding="tns:BasicHttpsBinding">
      <soap:address location="http://localhost:2138/soap-api/user"/>
    </wsdl:port>
  </wsdl:service>
</wsdl:definitions>
```

- Przykładowy WSDL „Usuwanie stacji”:

This XML file does not appear to have any style information associated with it. The document tree is shown below.

```
<?xml version="1.0" encoding="UTF-8"?>
<wsdl:definitions xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/" xmlns:tns="http://tempuri.org/" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:http="http://schemas.microsoft.com/ws/06/2004/policy/http" xmlns:msc="http://schemas.microsoft.com/ws/2005/12/wsdl/contract" xmlns:wsp="http://schemas.xmlsoap.org/ws/2004/09/policy"
  xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd" xmlns:wsam="http://www.w3.org/2007/05/addressing/metadata" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/"
  targetNamespace="http://tempuri.org/" name="IStationSoapEndpointService">
  <wsdl:types>
    <xsd:schema elementFormDefault="qualified" targetNamespace="http://tempuri.org/">
      <xsd:import namespace="http://schemas.microsoft.com/2003/10/Serialization/Arrays"/>
      <xsd:import namespace="http://schemas.datacontract.org/2004/07/System"/>
      <xsd:element name="DeleteStation">
        <xsd:complexType>
          <xsd:sequence>
            <xsd:element minOccurs="0" maxOccurs="1" name="id" type="tns:IdRequest"/>
          </xsd:sequence>
        </xsd:complexType>
      </xsd:element>
      <xsd:element name="DeleteStationResponse">
        <xsd:complexType>
          <xsd:sequence>
            <xsd:element minOccurs="0" maxOccurs="1" name="DeleteStationResult" type="tns:Unit"/>
          </xsd:sequence>
        </xsd:complexType>
      </xsd:element>
      <xsd:complexType name="IdRequest">
        <xsd:sequence>
          <xsd:element minOccurs="1" maxOccurs="1" name="Id" type="xsd:string"/>
        </xsd:sequence>
      </xsd:complexType>
      <xsd:complexType name="Unit">
        <xsd:sequence/>
      </xsd:complexType>
    </xsd:schema>
  </wsdl:types>
  <wsdl:message name="IStationSoapEndpointService_DeleteStation_InputMessage">
    <wsdl:part name="parameters" element="tns:DeleteStation"/>
  </wsdl:message>
  <wsdl:message name="IStationSoapEndpointService_DeleteStation_OutputMessage">
    <wsdl:part name="parameters" element="tns:DeleteStationResponse"/>
  </wsdl:message>
  <wsdl:portType name="IStationSoapEndpointService">
    <wsdl:operation name="DeleteStation">
      <wsdl:input message="tns:IStationSoapEndpointService_DeleteStation_InputMessage"/>
      <wsdl:output message="tns:IStationSoapEndpointService_DeleteStation_OutputMessage"/>
    </wsdl:operation>
  </wsdl:portType>
  <wsdl:binding name="BasicHttpsBinding" type="tns:IStationSoapEndpointService">
    <soap:binding transport="http://schemas.xmlsoap.org/soap/http"/>
    <wsdl:operation name="DeleteStation">
      <soap:operation soapAction="http://tempuri.org/IStationSoapEndpointService/DeleteStation" style="document"/>
      <wsdl:input>
        <soap:body use="literal"/>
      </wsdl:input>
      <wsdl:output>
        <soap:body use="literal"/>
      </wsdl:output>
    </wsdl:operation>
  </wsdl:binding>
  <wsdl:service name="IStationSoapEndpointService">
    <wsdl:port name="BasicHttpsBinding" binding="tns:BasicHttpsBinding">
      <soap:address location="http://localhost:2138/soap-api/station"/>
    </wsdl:port>
  </wsdl:service>
</wsdl:definitions>
```


2.4. Dostępne usługi

Usługi zostały podzielone na 5 grup użytkowych, reprezentowanych przez odpowiednie serwisy.

Grupa użytkowa	Dostępne usługi
Autoryzacja	<ul style="list-style-type: none">• Logowanie• Reset Hasła (krok 1 i 2)
Trasy kolejowe	<ul style="list-style-type: none">• Utwórz trasę (A)• Szczegóły trasy (U) (A)• Modyfikacja trasy (A)• Usunięcie trasy (A)• Wypisanie wszystkich tras
Stacje kolejowe	<ul style="list-style-type: none">• Utwórz stację (A)• Szczegóły stacji (U) (A)• Modyfikacja stacji (A)• Usunięcie stacji (A)• Wypisanie wszystkich stacji
Bilety	<ul style="list-style-type: none">• Tworzenie biletu (U) (A)• Szczegóły biletu (U) (A)• Tworzenie PDF biletu (U) (A)• Modyfikacja biletu (A)• Usunięcie biletu (A)• Wypisywanie wszystkich biletów danego użytkownika (U) (A)• Wypisywanie wszystkich biletów utworzonych w serwisie (A)
Użytkownik	<ul style="list-style-type: none">• Tworzenie użytkownika• Szczegóły użytkownika (U)• Modyfikacja użytkownika (U)• Usunięcie użytkownika (U)• Zmiana hasła (U)• Wypisanie wszystkich użytkowników (A)

U - Usługa dostępna dla użytkownika serwisu

A - Usługa dostępna dla administratora serwisu

2.5. Opis wybranych usług

2.5.1. Operacja tworzenie biletu

Operacja tworzenie biletu służy do dodawania wpisu o zakupie biletu do bazy danych, wpis zawiera takie dane jak identyfikator klienta, identyfikator trasy oraz datę zakupu. Aby korzystać z operacji użytkownik musi być zalogowany

Opis szczegółowy:

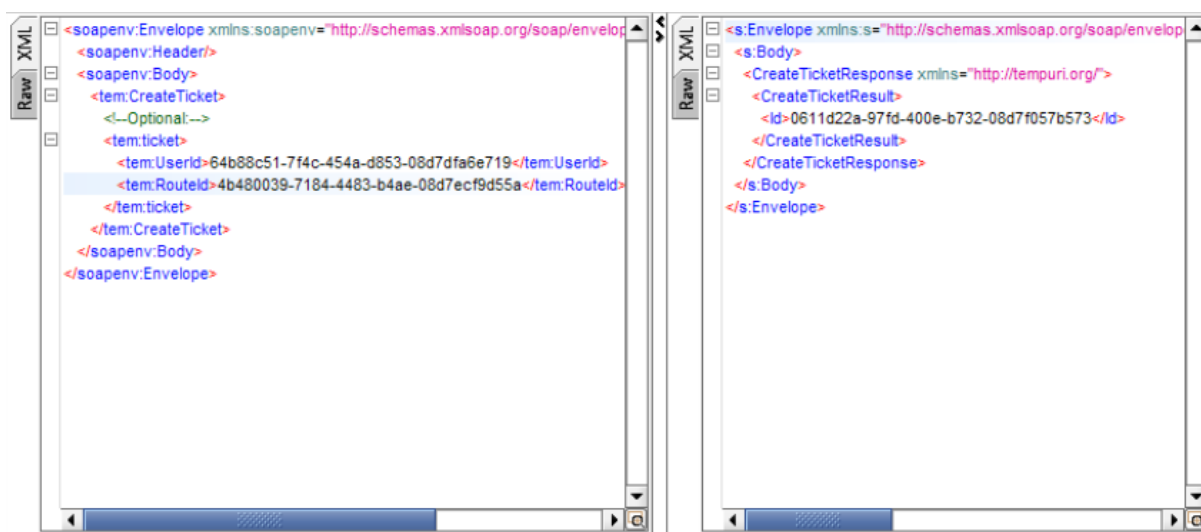
Dane wejściowe:

- **UserId** – identyfikator użytkownika – 32-znakowy ciąg heksadecymalny np. 3fa85f64-5717-4562-b3fc-2c963f66afa6, wygenerowany przy pomocy GUID
- **Routeld** – identyfikator trasy kolejowej – 32-znakowy ciąg heksadecymalny np. 3fa85f64-5717-4562-b3fc-2c963f66afa6, wygenerowany przy pomocy GUID

Dane wyjściowe:

- **TicketId** – identyfikator nowo powstałego biletu – 32-znakowy ciąg heksadecymalny np. 3fa85f64-5717-4562-b3fc-2c963f66afa6, wygenerowany przy pomocy GUID
lub
- **Komunikat o błędzie** – w razie błędnych danych lub braku uprawnień do korzystania z operacji pojawi się komunikat o błędzie

Komunikat wysyłany do usługi i zwracany z usługi:



2.5.2. Operacja pobranie listy biletów danego klienta

Operacja pobrania listy biletów danego klienta służy do wylistowania wszystkich zakupionych biletów przez klienta, wpis w liście zawiera takie dane jak identyfikator biletu, identyfikator trasy, dane stacji startowej i docelowej oraz datę zakupu biletu. Aby korzystać z operacji użytkownik musi być zalogowany.

Opis szczegółowy:

Dane wejściowe:

- **UserId** – identyfikator użytkownika – 32-znakowy ciąg heksadecymalny np. 3fa85f64-5717-4562-b3fc-2c963f66afa6, wygenerowany przy pomocy GUID

Dane wyjściowe:

- **Tickets** – lista biletów wraz z ich szczegółami lub
- **Komunikat o błędzie** – w razie błędnych danych lub braku uprawnień do korzystania z operacji pojawi się komunikat o błędzie

Komunikat wysyłany do usługi i zwracany z usługi:

The left screenshot displays the raw XML of the request:

```
<?xml version='1.0' encoding='utf-8'>
<soap:Envelope xmlns:soap='http://schemas.xmlsoap.org/soap/envelope/'>
  <soap:Header/>
  <soap:Body>
    <tem:GetCurrentUserTicketsList/>
  </soap:Body>
</soap:Envelope>
```

Below the XML view, a table shows the Authorization header:

Header	Value
Authorization	Bearer eyJhbGciOiJIU...

The right screenshot displays the raw XML of the response:

```
<?xml version='1.0' encoding='utf-8'>
<s:Envelope xmlns:s='http://schemas.xmlsoap.org/soap/envelope/' xmlns:xsd='http://www.w3.org/2001/XMLSchema'>
  <s:Body>
    <GetCurrentUserTicketsListResponse xmlns='http://tempuri.org/'>
      <GetCurrentUserTicketsListResult>
        <Tickets>
          <UserTicketLookupModel>
            <Id>40e6d541-a3bc-431c-7f68-08d7edfea8c7</Id>
            <UserId>64b88c51-7f4c-454a-d853-08d7dfa6e719</UserId>
            <Route>
              <Id>866e09db-dcc8-42e4-b4b6-08d7ecf9d55a</Id>
              <From>
                <Id>73b98be1-c55f-4c9c-f248-08d7e626224a</Id>
                <Name>Białystok</Name>
                <Latitude>53.13272222</Latitude>
                <Longitude>23.13361111</Longitude>
              </From>
              <To>
                <Id>f77e0d41-29b2-4c80-f249-08d7e626224a</Id>
                <Name>Olsztyn</Name>
                <Latitude>53.77686111</Latitude>
                <Longitude>20.48166667</Longitude>
              </To>
              <DepartureTime>2020-05-11T08:20:16.905</DepartureTime>
              <Duration>PT4H16M</Duration>
              <Distance>229</Distance>
              <TicketPrice>55</TicketPrice>
            </Route>
          </UserTicketLookupModel>
          <UserTicketLookupModel>
            <Id>138023ca-5c7e-418e-7f69-08d7edfea8c7</Id>
            <UserId>64b88c51-7f4c-454a-d853-08d7dfa6e719</UserId>
            <Route>
              <Id>866e09db-dcc8-42e4-b4b6-08d7ecf9d55a</Id>
              <From>
                <Id>73b98be1-c55f-4c9c-f248-08d7e626224a</Id>
                <Name>Białystok</Name>
                <Latitude>53.13272222</Latitude>
                <Longitude>23.13361111</Longitude>
              </From>
              <To>
                <Id>f77e0d41-29b2-4c80-f249-08d7e626224a</Id>
                <Name>Olsztyn</Name>
                <Latitude>53.77686111</Latitude>
                <Longitude>20.48166667</Longitude>
              </To>
              <DepartureTime>2020-05-11T08:20:16.905</DepartureTime>
              <Duration>PT4H16M</Duration>
              <Distance>229</Distance>
              <TicketPrice>55</TicketPrice>
            </Route>
          </UserTicketLookupModel>
        </Tickets>
      </GetCurrentUserTicketsListResult>
    </GetCurrentUserTicketsListResponse>
  </s:Body>
</s:Envelope>
```

2.5.3. Operacja szczegóły stacji

Operacja szczegóły stacji służy do wypisania wszystkich danych o wybranej stacji zawartych w bazie, szczegóły te obejmują dane takie jak nazwa, czy też położenie geograficzne stacji. Aby korzystać z operacji użytkownik musi być zalogowany.

Opis szczegółowy:

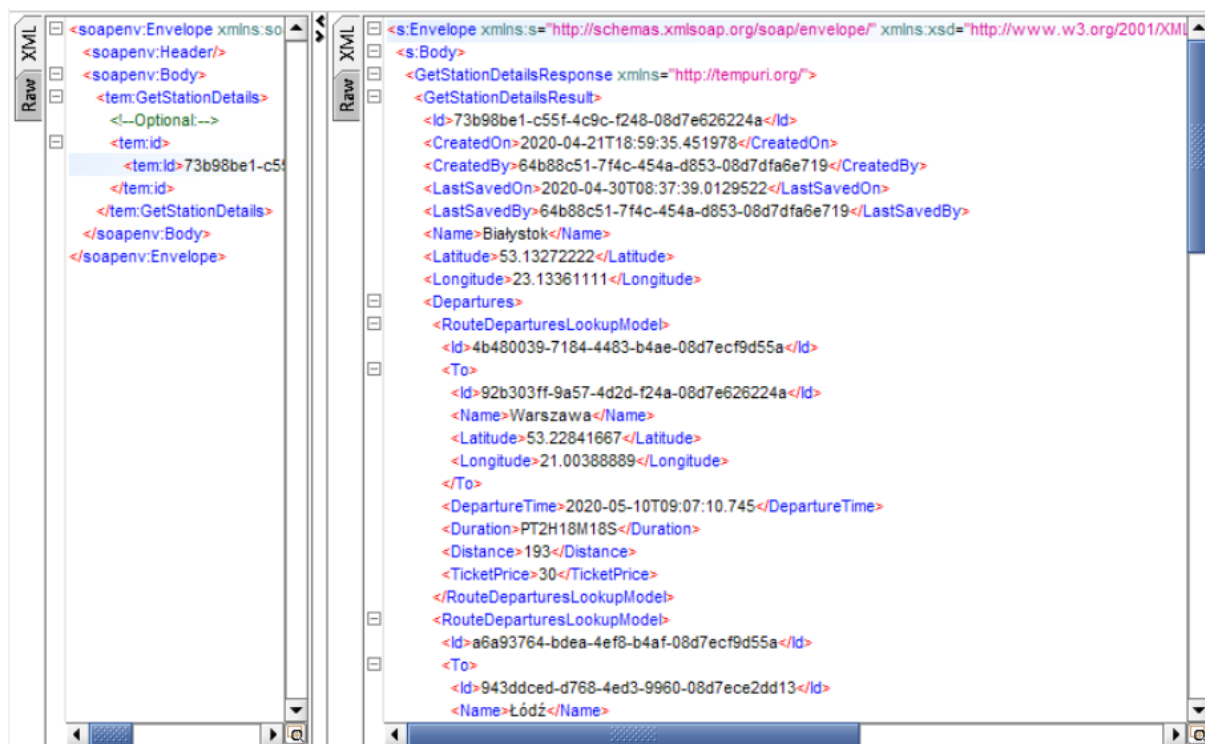
Dane wejściowe:

- **stationId** – identyfikator stacji– 32-znakowy ciąg heksadecymalny np. 3fa85f64-5717-4562-b3fc-2c963f66afa6, wygenerowany przy pomocy GUID

Dane wyjściowe:

- **Station**– wszystkie szczegóły stacji zawarte w bazie łączenie ze stacjami do których wychodzą trasy z danej stacji
lub
- **Komunikat o błędzie** – w razie błędnych danych lub braku uprawnień do korzystania z operacji pojawi się komunikat o błędzie

Komunikat wysyłany do usługi i zwracany z usługi:



2.5.4. Operacja usunięcie stacji

Operacja usunięcie stacji służy do usunięcia wszystkich danych o wybranej stacji zawartych w bazie, łącznie z trasami, które są z lub do danej stacji. Aby korzystać z operacji użytkownik musi być zalogowany i posiadać uprawnienia administratora.

Opis szczegółowy:

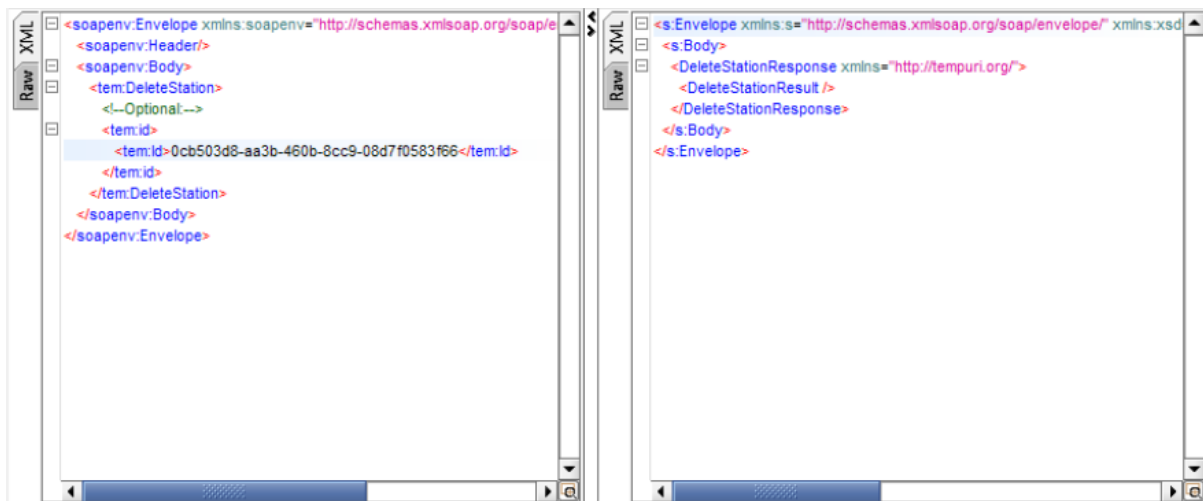
Dane wejściowe:

- **stationId** – identyfikator stacji– 32-znakowy ciągu heksadecymalnego np. 3fa85f64-5717-4562-b3fc-2c963f66afa6, wygenerowany przy pomocy GUID

Dane wyjściowe:

- Operacja udana
lub
- Komunikat o błędzie – w razie błędnych danych lub braku uprawnień do korzystania z operacji pojawi się komunikat o błędzie

Komunikat wysyłany do usługi i zwracany z usługi:



2.5.5. Operacja zmiana hasła

Operacja zmiana hasła służy do zmiany hasła przez indywidualnego użytkownika. W celu wykonania operacji użytkownik musi znać stare hasło. Aby korzystać z operacji użytkownik musi być zalogowany.

Opis szczegółowy:

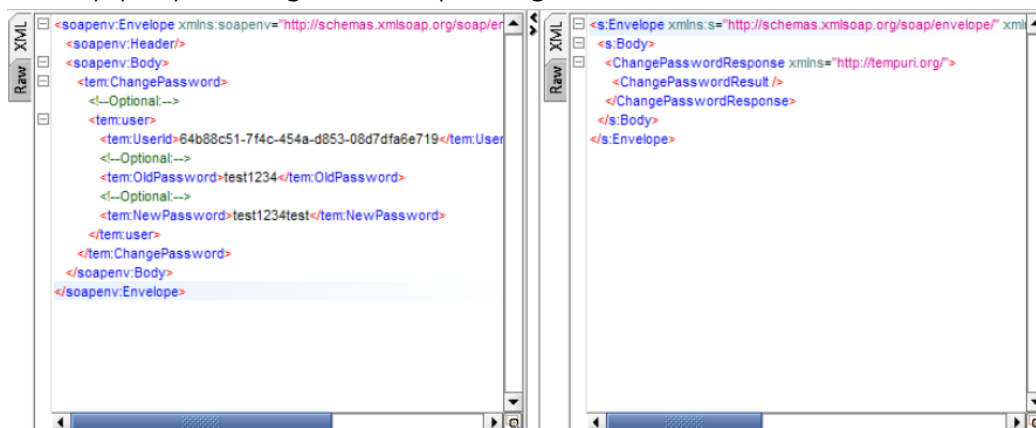
Dane wejściowe:

- **stationId** – identyfikator użytkownika– 32-znakowy ciąg heksadecymalny np. 3fa85f64-5717-4562-b3fc-2c963f66afa6, wygenerowany przy pomocy GUID
- **oldPassword** – stare hasło do konta
- **newPassword** – nowe hasło do konta

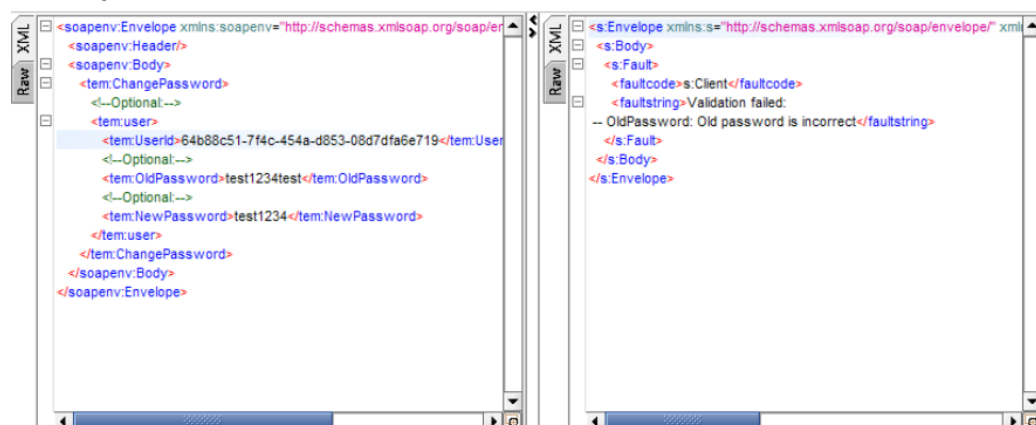
Dane wyjściowe:

- **Operacja udana**
lub
- **Komunikat o błędzie** – w razie błędnych danych lub braku uprawnień do korzystania z operacji pojawi się komunikat o błędzie

Komunikat wysyłany do usługi i zwracany z usługi:

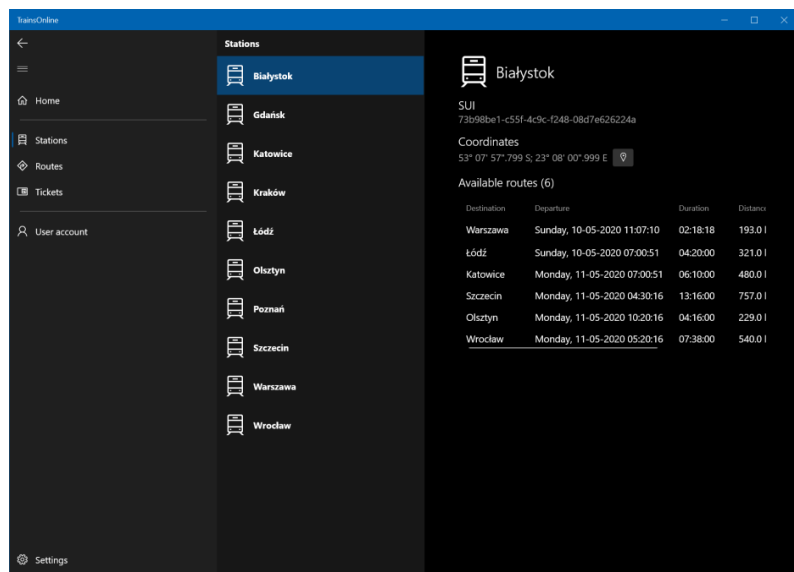
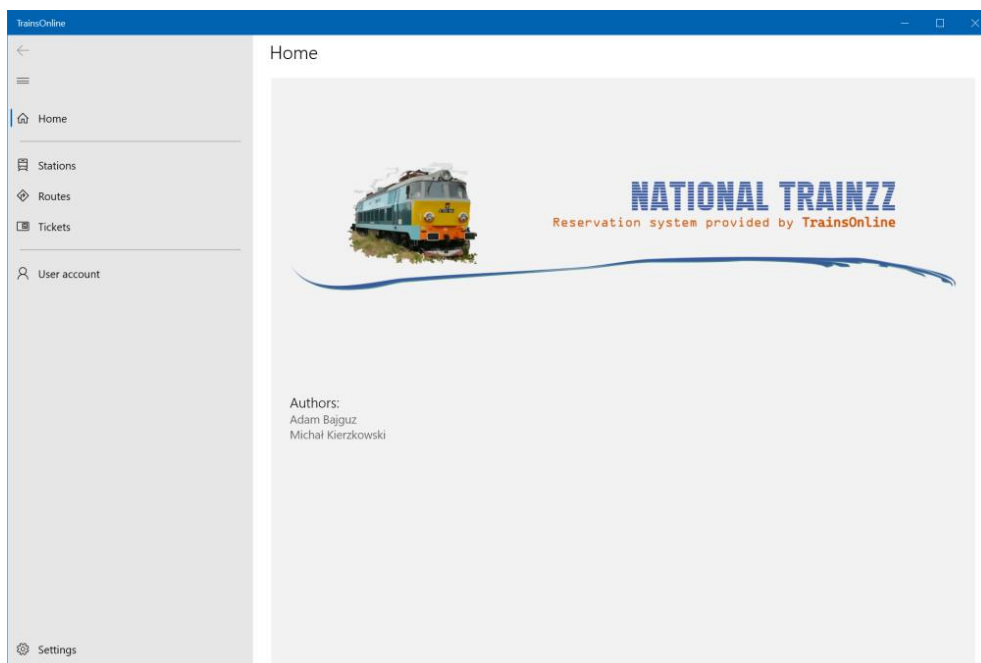


Komunikat błędu:



3. Instrukcja użytkownika aplikacji klienckiej

Program kliencki jest prostą i intuicyjną w obsłudze responsywną aplikacją okienkową powstałą w technologii Universal Windows Platform (UWP) i działa zarówno na komputerach, jak i telefonach z system Windows.



TrainsOnline

Posiada ona obsługę funkcjonalności API takich jak:

- logowanie i rejestracja;

The screenshot shows the 'Tickets' section of the TrainsOnline application. It features a sidebar with navigation links: Home, Stations, Routes, Tickets, and User account. The main content area is titled 'Tickets' and contains a message: 'Authorization is required to access this page'. Below this message are two tabs: 'Login' and 'Register'. The 'Login' tab is active, showing input fields for 'e-mail' and 'password', and a 'Login' button. The 'Register' tab is also visible, showing input fields for 'e-mail', 'password', 'name', 'surname', 'phone number', and 'address', along with a 'Register' button.

This screenshot shows the 'Register' tab of the TrainsOnline application. It includes the same sidebar as the previous screenshot. The main content area is titled 'Tickets' and contains a message: 'Authorization is required to access this page'. Below this message are two tabs: 'Login' and 'Register'. The 'Register' tab is active, showing input fields for 'e-mail', 'password', 'name', 'surname', 'phone number', and 'address', along with a 'Register' button.

- możliwość kupna biletów;

The screenshot shows the 'Routes' section of the TrainsOnline application. It features a sidebar with navigation links: Home, Stations, Routes, Tickets, and User account. The main content area is titled 'Routes' and contains a search bar with fields for 'Departure station', 'Arrival station', and 'Maximum price'. Below the search bar is a table of routes. The table has columns for 'Destination', 'Departure', 'Distance', 'Ticket price', and 'Actions'. The routes are listed under two categories: 'Departure from: Białystok (6 items)' and 'Departure from: Warszawa (4 items)'. Each route entry includes a train icon, the destination, the departure date and time, the distance, the ticket price, and icons for favorite and cart actions.

Destination	Departure	Distance	Ticket price	Actions
Departure from: Białystok (6 items)				
Warszawa	Sunday, 10-05-2020 11:07:10	193.0 km	\$30.00	
Route details: Departure from 'Białystok' at 'Sunday, 10-05-2020 11:07:10' to 'Warszawa' Arrival in 'Warszawa' at 'Sunday, 10-05-2020 01:23:28' after '02:18:18'				
Departure station coordinates: '53° 07' 57" 799 S, 23° 08' 00" 999 E				
Arrival station coordinates: '53° 13' 42" 300 S, 21° 00' 14" 000 E				
Łódź	Sunday, 10-05-2020 07:00:51	321.0 km	\$70.00	
Katowice	Monday, 11-05-2020 07:00:51	480.0 km	\$90.00	
Szczecin	Monday, 11-05-2020 04:30:16	757.0 km	\$164.00	
Olsztyn	Monday, 11-05-2020 10:20:16	229.0 km	\$55.00	
Wrocław	Monday, 11-05-2020 05:20:16	540.0 km	\$99.00	
Departure from: Warszawa (4 items)				
Katowice	Monday, 11-05-2020 03:09:16	300.0 km	\$55.00	
Olsztyn	Monday, 11-05-2020 10:00:16	200.0 km	\$40.00	
Kraków	Monday, 11-05-2020 06:00:16	290.0 km	\$42.00	
Szczecin	Monday, 11-05-2020 07:30:16	560.0 km	\$102.00	

The screenshot shows the 'Tickets' section of the TrainsOnline application. It features a sidebar with navigation links: Home, Stations, Routes, Tickets, and User account. The main content area is titled 'Tickets' and displays a grid of ticket offers. Each offer includes a train icon, the route, the departure date and time, and the ticket price. The offers are organized into three columns and three rows.

Route	Departure	Ticket price
Białystok → Olsztyn	Monday, 11-05-2020 10:20:16	\$55.00
Białystok → Olsztyn	Monday, 11-05-2020 10:20:16	\$55.00
Warszawa → Olsztyn	Monday, 11-05-2020 10:00:16	\$40.00
Białystok → Olsztyn	Monday, 11-05-2020 10:20:16	\$55.00
Warszawa → Olsztyn	Monday, 11-05-2020 10:00:16	\$40.00
Warszawa → Szczecin	Monday, 11-05-2020 07:30:16	\$102.00
Warszawa → Kraków	Monday, 11-05-2020 06:00:16	\$42.00
Białystok → Warszawa	Sunday, 10-05-2020 11:07:10	\$30.00



NATIONAL TRAINZZ TICKET

Provided by TrainsOnline

TICKET	
TID	PKP Ticket {e2214a8f-b926-40ba-4d16-08d7e6b99c7a}
TIMESTAMP	2020-04-22 12:35:16
ROUTE	Gdańsk → Poznań
DEPARTURE	2020-04-22 10:40:28
PASSENGER	
UID	{b2ac5452-e45c-4ed3-4d76-08d7dfa3a155}
NAME	string string
E-MAIL	test0@test.pl
ADDRESS	string
PHONE	string
ROUTE DETAILS	
RID	{29d3fa8b-77bf-40aa-1ccf-08d7e6a9aa76}
CREATED ON	2020-04-22 12:35:16

Departure	Arrival	Travel time	Distance	Ticket price
Gdańsk 2020-04-22 10:40:28	Poznań 2020-04-22 12:40:28	02:00:00	100 km	\$1000

MISCELLANEOUS

Calendar event QR Code:



This code allows you to add an event to calendar app on your device. The event will contain all crucial data from this ticket.

Verification QR Code:

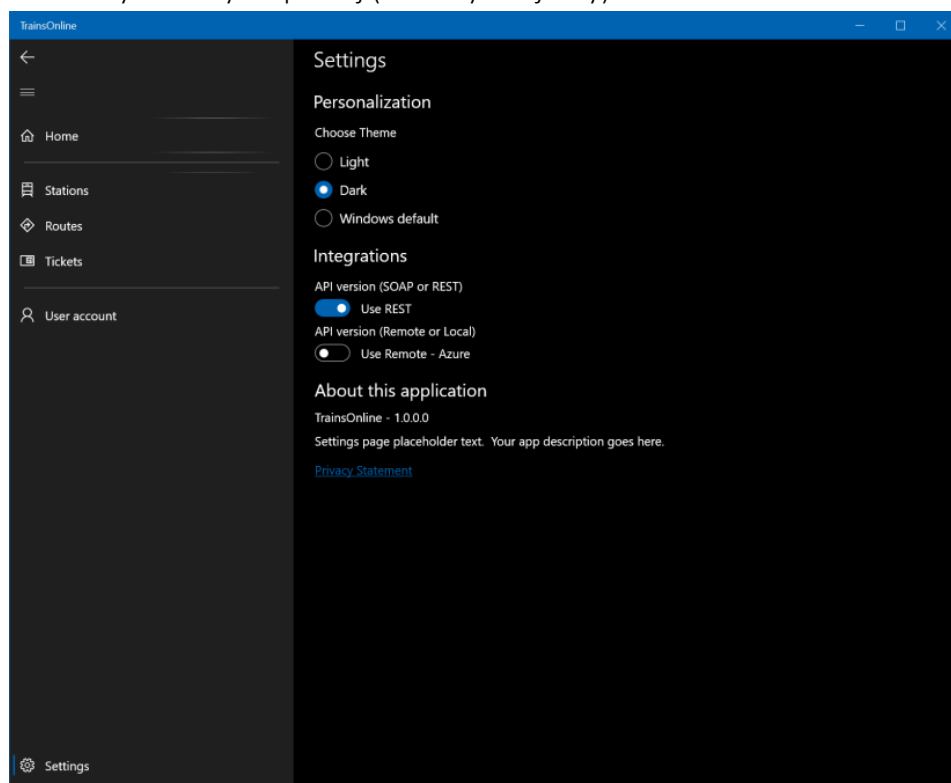


The verification QR Code allows you to verify authenticity of the ticket using a dedicated online service.

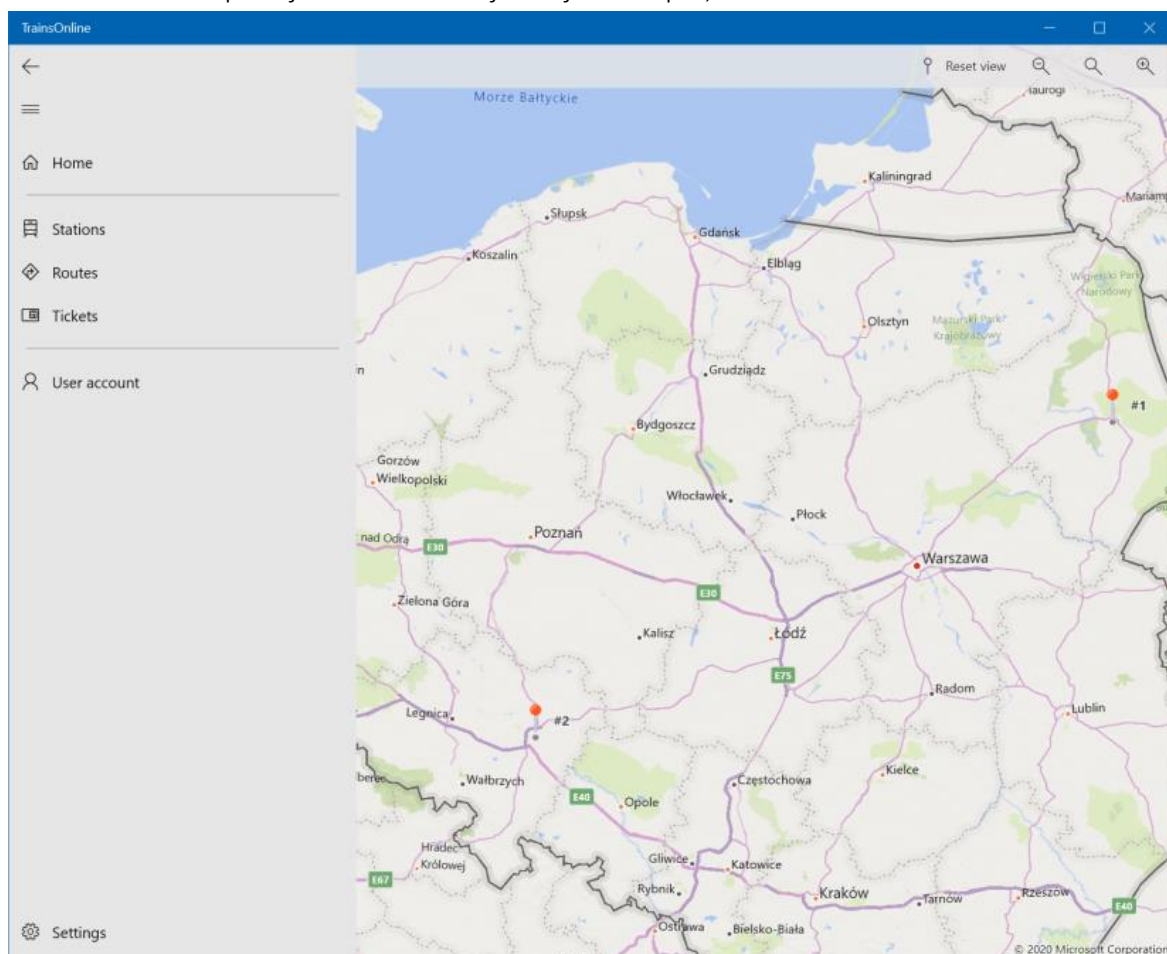
i wiele innych.

Poza spełnieniem podstawnych założeń aplikacja posiada dodatkowe funkcjonalności taki jak:

- możliwość wyboru stylu aplikacji(ciemny lub jasny)



- możliwość podejrzenia lokalizacji stacji na mapie;



- generowanie kodów QR

Calendar event QR Code:



This code allows you to add an event to calendar app on your device. The event will contain all crucial data from this ticket.

Verification QR Code:



The verification QR Code allows you to verify authenticity of the ticket using a dedicated online service.

- wykorzystania algorytmu Jaro-Winkler

← Routes

Departure station: Arrival station: Maximum price: 🔍 ✕

	Destination	Departure	Distance	Ticket price
▼	Departure from: Białystok (1 item)			
🚏	Olsztyn	Monday, 11-05-2020 10:20:16	229.0 km	\$55.00
<u>Route details:</u> Departure from 'Białystok' at 'Monday, 11-05-2020 10:20:16' to 'Olsztyn' Arrival in 'Olsztyn' at 'Monday, 11-05-2020 02:36:16' after '04:16:00'				
Departure station coordinates: '53° 07' 57".799 S; 23° 08' 00".999 E 📍				
Arrival station coordinates: '53° 46' 36".699 S; 20° 28' 54".000 E 📍				

- dziennik logów

genericapi

All Environments

Last 14 days

Issues (4)

Sort by: Last Seen

Unresolved Issues

is:unresolved

<input type="checkbox"/>	<input type="button" value="Resolve"/>	<input type="button" value="Ignore"/>	<input type="button" value="Merge"/>	<input type="button" value="More"/>	<input type="button" value="Play"/>	GRAPH:	24h	14d	EVENTS	USERS	ASSIGNEE
<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>				1	1	
HTTP "POST" "/api/user/reset-password" responded 500 in 2565...											
GENERICAPI-4 a few seconds ago Serilog.AspNetCore.RequestLoggingMiddleware											
<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>				1	1	
System.Net.Mail.SmtpException TrainsOnline.Infrastructure.Main.Ema...											
The SMTP server requires a secure connection or the client was not authenti...											
GENERICAPI-3 a few seconds ago - a few seconds old											
<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>				1	1	
Failed to determine the https port for redirect.											
GENERICAPI-2 Microsoft.AspNetCore.HttpsPolicy.HttpsRedirectionMiddleware											
<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>				2	1	
Server START: "PRODUCTION" mode enabled. --ef-migrate=False --...											
GENERICAPI-1 2 minutes ago - 2 minutes old											

```
Mon May 04 19:33:38 UTC /home/aba/DataDisk0/GenericAPI/logs/production_20200504.log generic_log: LOG
2020-05-04 19:11:40.169 +00:00 [INF] <dotnet 55795:12> (0HLVGBRSEI0J9:00000001-/api/user/reset-password) Entity Framework Core 3.1.3 initialized 'PKPAppDbContext' u
2020-05-04 19:11:41.135 +00:00 [INF] <dotnet 55795:13> (0HLVGBRSEI0J9:00000001-/api/user/reset-password) Executed DbCommand (126ms) [Parameters=[@__data_Email_0=?],
SELECT TOP(1) [u].[Id], [u].[Address], [u].[CreatedBy], [u].[CreatedOn], [u].[Email], [u].[IsAdmin], [u].[LastSavedBy], [u].[LastSavedOn], [u].[Name], [u].[Password]
FROM [Users] AS [u]
WHERE [u].[Email] = @__data_Email_0
2020-05-04 19:11:41.574 +00:00 [INF] <dotnet 55795:12> (0HLVGBRSEI0J9:00000001-/api/user/reset-password) Executed action TrainsOnline.Api.Controllers.Authentication
2020-05-04 19:11:41.575 +00:00 [INF] <dotnet 55795:12> (0HLVGBRSEI0J9:00000001-/api/user/reset-password) Executed endpoint 'TrainsOnline.Api.Controllers.Authentication
2020-05-04 19:11:41.579 +00:00 [ERR] <dotnet 55795:12> (0HLVGBRSEI0J9:00000001-/api/user/reset-password) Unhandled exception in CustomExceptionHandlerMiddleware
System.Net.Mail.SmtpException: The SMTP server requires a secure connection or the client was not authenticated. The server response was: 5.7.0 Authentication Requ
    at System.Net.Mail.MailCommand.CheckResponse(SmtpStatusCode statusCode, String response)
    at System.Net.Mail.MailCommand.EndSend(IAsyncResult result)
    at System.Net.Mail.SendMailAsyncResult.SendMailFromCompleted(IAsyncResult result)
--- End of stack trace from previous location where exception was thrown ---
    at System.Runtime.ExceptionServices.ExceptionDispatchInfo.Throw(Exception source)
    at System.Net.Mail.SendMailAsyncResult.End(IAsyncResult result)
    at System.Net.Mail.SmtpClient.SendMailCallback(IAsyncResult result)
--- End of stack trace from previous location where exception was thrown ---
    at TrainsOnline.Infrastructure.Main.Email.EmailService.SendEmail(String email, String subject, String message) in /home/vsts/work/1/s/backend/TrainsOnline.Infras
    at TrainsOnline.Application.Handlers.AuthenticationHandlers.Queries.GetResetPasswordToken.GetResetPasswordTokenQuery.Handler.Handle(GetResetPasswordTokenQuery re
    at Mediatr.Pipeline.RequestExceptionHandlerProcessorBehavior`2.Handle(TRequest request, CancellationToken cancellationToken, RequestHandlerDelegate`1 next)
    at Mediatr.Pipeline.RequestExceptionHandlerProcessorBehavior`2.Handle(TRequest request, CancellationToken cancellationToken, RequestHandlerDelegate`1 next)
    at Mediatr.Pipeline.RequestExceptionHandlerProcessorBehavior`2.Handle(TRequest request, CancellationToken cancellationToken, RequestHandlerDelegate`1 next)
    at Mediatr.Pipeline.RequestExceptionHandlerProcessorBehavior`2.Handle(TRequest request, CancellationToken cancellationToken, RequestHandlerDelegate`1 next)
    at Mediatr.Pipeline.RequestExceptionHandlerProcessorBehavior`2.Handle(TRequest request, CancellationToken cancellationToken, RequestHandlerDelegate`1 next)
    at Mediatr.Pipeline.RequestExceptionHandlerProcessorBehavior`2.Handle(TRequest request, CancellationToken cancellationToken, RequestHandlerDelegate`1 next)
    at TrainsOnline.Api.Controllers.AuthenticationController.ResetPasswordStep1(SendResetPasswordRequest request) in /home/vsts/work/1/s/backend/TrainsOnline.Api/Con
    at Microsoft.AspNetCore.Mvc.Infrastructure.ActionMethodExecutor.TaskOfIActionResultExecutor.Execute(IActionResultTypeMapper mapper, ObjectMethodExecutor executor
    at Microsoft.AspNetCore.Mvc.Infrastructure.ControllerActionInvoker.<InvokeActionMethodAsync>g__Awaited|12_0(ControllerActionInvoker invoker, ValueTask`1 actionRe
    at Microsoft.AspNetCore.Mvc.Infrastructure.ControllerActionInvoker.<InvokeNextActionFilterAsync>g__Awaited|10_0(ControllerActionInvoker invoker, Task lastTask, S
    at Microsoft.AspNetCore.Mvc.Infrastructure.ControllerActionInvoker.Rethrow(ActionExecutedContextSealed context)
    at Microsoft.AspNetCore.Mvc.Infrastructure.ControllerActionInvoker.Next(State& next, Scope& scope, Object& state, Boolean& isCompleted)
    at Microsoft.AspNetCore.Mvc.Infrastructure.ControllerActionInvoker.<InvokeInnerFilterAsync>g__Awaited|13_0(ControllerActionInvoker invoker, Task lastTask, State
    at Microsoft.AspNetCore.Mvc.Infrastructure.ResourceInvoker.<InvokeFilterPipelineAsync>g__Awaited|19_0(ResourceInvoker invoker, Task lastTask, State next, Scope s
    at Microsoft.AspNetCore.Mvc.Infrastructure.ResourceInvoker.<InvokeAsync>g__Logged|17_1(ResourceInvoker invoker)
    at Microsoft.AspNetCore.Routing.EndpointMiddleware.<Invoke>g__AwaitRequestTask|6_0(Endpoint endpoint, Task requestTask, ILogger logger)
    at Microsoft.AspNetCore.Authorization.AuthorizationMiddleware.Invoke(HttpContext context)
    at Microsoft.AspNetCore.Authentication.AuthenticationMiddleware.Invoke(HttpContext context)
    at Microsoft.AspNetCore.Diagnostics.StatusCodePagesMiddleware.Invoke(HttpContext context)
    at Microsoft.AspNetCore.ResponseCompression.ResponseCompressionMiddleware.Invoke(HttpContext context)
    at TrainsOnline.Api.CustomMiddlewares.Exceptions.CustomExceptionHandlerMiddleware.Invoke(HttpContext context) in /home/vsts/work/1/s/backend/TrainsOnline.Api/Cus
2020-05-04 19:11:41.615 +00:00 [ERR] <dotnet 55795:12> (0HLVGBRSEI0J9:00000001-/api/user/reset-password) HTTP POST /api/user/reset-password responded 500 in 2565.49
2020-05-04 19:11:41.627 +00:00 [INF] <dotnet 55795:12> (0HLVGBRSEI0J9:00000001-/api/user/reset-password) Request finished in 2611.9695ms 500 application/json
2020-05-04 19:14:52.759 +00:00 [INF] <dotnet 55795:24> (0HLVGBRSEI0JA:00000001-/api/index.html) Request starting HTTP/1.1 GET http://genericapi.francecentral.clouda
2020-05-04 19:14:52.826 +00:00 [INF] <dotnet 55795:24> (0HLVGBRSEI0JA:00000001-/api/index.html) HTTP GET /api/index.html responded 200 in 63.544454 ms
2020-05-04 19:14:52.826 +00:00 [INF] <dotnet 55795:24> (0HLVGBRSEI0JA:00000001-/api/index.html) Request finished in 67.5295ms 200 text/html; charset=utf-8
2020-05-04 19:14:52.930 +00:00 [INF] <dotnet 55795:23> (0HLVGBRSEI0JB:00000001-/api/v1/swagger.json) Request starting HTTP/1.1 GET http://genericapi.francecentral.c
2020-05-04 19:14:52.933 +00:00 [INF] <dotnet 55795:24> (0HLVGBRSEI0JC:00000001-/api/favicon-32x32.png) Request starting HTTP/1.1 GET http://genericapi.francecentral
2020-05-04 19:14:52.934 +00:00 [INF] <dotnet 55795:23> (0HLVGBRSEI0JB:00000001-/api/v1/swagger.json) Successfully validated the token.
2020-05-04 19:14:52.944 +00:00 [INF] <dotnet 55795:24> (0HLVGBRSEI0JC:00000001-/api/favicon-32x32.png) Sending file. Request path: '/favicon-32x32.png'. Physical pa
2020-05-04 19:14:52.945 +00:00 [INF] <dotnet 55795:24> (0HLVGBRSEI0JC:00000001-/api/favicon-32x32.png) HTTP GET /api/favicon-32x32.png responded 200 in 10.885127 ms
2020-05-04 19:14:52.945 +00:00 [INF] <dotnet 55795:24> (0HLVGBRSEI0JC:00000001-/api/favicon-32x32.png) Request finished in 11.6189ms 200 image/png
2020-05-04 19:14:53.124 +00:00 [INF] <dotnet 55795:23> (0HLVGBRSEI0JB:00000001-/api/v1/swagger.json) HTTP GET /api/v1/swagger.json responded 200 in 193.99577 ms
2020-05-04 19:14:53.125 +00:00 [INF] <dotnet 55795:23> (0HLVGBRSEI0JB:00000001-/api/v1/swagger.json) Request finished in 195.0167ms 200 application/json; charset=utf
2020-05-04 19:15:00.574 +00:00 [INF] <dotnet 55795:23> (0HLVGBRSEI0JD:00000001-/api/user/reset-password) Request starting HTTP/1.1 POST http://genericapi.francecent
2020-05-04 19:15:00.581 +00:00 [INF] <dotnet 55795:23> (0HLVGBRSEI0JD:00000001-/api/user/reset-password) CORS policy execution failed.
L10857 99% 0 hits ??:View Help
Press e/E to move forward/backward through error messages
```

- wysyłanie emaili

