

1. Podłączenie parkingu do ITS w celu monitorowania zajętości miejsc postojowych

Obecnie system ITS obsługuje następujące kanały komunikacji z operatorami parkingów:

- otwarty interfejs API RESTful do zastosowania przez dowolnego operatora przy integracji systemu parkingowego
- interfejs oparty o technologię Webservice/SOAP (metod alternatywna - archiwalna) dla operatorów parkingów, którzy nie posiadają możliwości komunikacji za pomocą rozwiązania RESTful.

2. Instrukcja interfejsu API

Mechanizm funkcji API zrealizowany jest z wykorzystaniem technologii RESTful z zastosowaniem formatu JSON do opisu operacji oraz parametrów. Jako protokół transportowy może być wykorzystywany HTTP (transmisja nieszyfrowana) lub HTTPS (transmisja szyfrowana).

Usługi świadczone przez platformę API zostały opisane w postaci kart usług.

2.1. Autoryzacja – logowanie

Nazwa	Logowanie klienta <ul style="list-style-type: none">• Używany do pozyskania tokena autoryzacji umożliwiającego prowadzenie dalszej komunikacji
URL	/v1/client/login/json
Metoda	POST
URL Nagłówki	---
Parametry	Przykład: <pre>{ user: [string], pass: [string] }</pre> Przykład: <pre>{ user: "test_user",</pre>

	<pre>pass: "test_pass" }</pre> <p>Parametry:</p> <table border="1"> <tr> <td>user</td> <td>Nazwa użytkownika dostarczona przez operatora systemu</td> </tr> <tr> <td>pass</td> <td>Hasło użytkownika dostarczone przez operatora systemu</td> </tr> </table>	user	Nazwa użytkownika dostarczona przez operatora systemu	pass	Hasło użytkownika dostarczone przez operatora systemu
user	Nazwa użytkownika dostarczona przez operatora systemu				
pass	Hasło użytkownika dostarczone przez operatora systemu				
<p>Odpowiedź (sukces)</p>	<p>Odpowiedź:</p> <p>Code: 200 OK</p> <p>Content:</p> <pre>{ token: "36f10fc7a18c8cc477499d476ffdaffdcf640062", token_expiration_date: "2017-03-15 11:17:25.858938" }</pre> <p>Parametry:</p> <p>token: token autoryzacji sesji użytkownika token_expiration_date: data wygaśnięcia tokenu</p>				
<p>Odpowiedź (błąd)</p>	<p>Odpowiedź:</p> <p>Code: 403 FORBIDDEN</p> <p>Content:</p> <pre>{ message : "Wrong User" } { message : "Wrong Password" }</pre> <p>Code: 400 BAD REQUEST</p> <p>Content:</p> <pre>{ message: { <*param>: "Missing required parameter in the JSON body" } } { message: "Failed to decode JSON object: ..." }</pre> <p>*param: name of parameter missing in json body</p> <p>Code: 415 Unsupported Media Type</p> <p>Content:</p> <pre>{ message: "Invalid Content-Type" }</pre>				
<p>Przykład wywołania</p>	<pre>curl -v -H "content-type: application/json" -H "content-encoding: utf-8" -X POST -d '{"user": "test_user", "pass": "test_pass"}' http://<ip>/v1/client/login/json</pre>				

Komentarz	<ul style="list-style-type: none"> - Kodowanie znaków UTF-8. - W przypadku wysłania żądania "login" przez zalogowanego użytkownika, data wygaśnięcia tokenu zostanie przesunięta o 24 godziny.
------------------	--

2.2. Autoryzacja – wylogowanie

Nazwa	Wylogowanie klienta <ul style="list-style-type: none"> • Używany do unieważnienia tokena autoryzacji 		
URL	<code>/v1/client/logout/json</code>		
Metoda	POST		
URL Nagłówki	Content-Type: application/json Token: <session token> User: <user name>		
Parametry	<p>Przykład:</p> <pre>{ user: [string] }</pre> <p>Przykład:</p> <pre>{ user: "test_user" }</pre> <p>Parametry:</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 20%; padding: 5px;"><code>user</code></td> <td style="padding: 5px;">nazwa użytkownika dostarczona przez operatora systemu</td> </tr> </table>	<code>user</code>	nazwa użytkownika dostarczona przez operatora systemu
<code>user</code>	nazwa użytkownika dostarczona przez operatora systemu		
Odpowiedź (sukces)	<p>Odpowiedź:</p> <p>Code: 200 OK Content:</p> <pre>{ reply: "Logged out" }</pre>		

<p>Odpowiedź (błąd)</p>	<p>Odpowiedź:</p> <p>Code: 401 Unauthorized Content: <pre>{ message: "User not authorized" } { message: "Invalid token" } { message: "Session token Expired" }</pre></p> <p>Code: 415 Unsupported media type Content: <pre>{ message : "Invalid content-type" }</pre></p> <p>Code: 50X Server error Content: <pre>{ message : "Server error" } { message : "Database error" }</pre></p> <p>Code: 400 BAD REQUEST Content: <pre>{ message: { <*param>: "Missing required parameter in the JSON body" } } { message: "Failed to decode JSON object: ..." }</pre></p> <p><i>*param: name of parameter missing in json body</i></p>
<p>Przykład wywołania</p>	<pre>curl -v -H "content-type: application/json" -H "content-encoding: utf-8" -X POST -d '{"user": "test_user"}' http://<ip>/v1/client/logout/json</pre>
<p>Komentarz</p>	<ul style="list-style-type: none"> - Kodowanie znaków UTF-8. - Pole "user" w treści zapytania musi zgadzać się z nazwą zalogowanego użytkownika wysyłającego zapytanie.

2.3. Rejestracja informacji o stanie wolnych miejsc na parkingu

<p>Nazwa</p>	<p>Dodaj informacje o stanie wolnych miejsc na parkingu</p> <ul style="list-style-type: none"> • Używany do cyklicznej notyfikacji systemu o bieżącej zajętości miejsc na parkingach objętych monitorowaniem wolnych miejsc.
---------------------	--

URL	<code>/v2/parking/occupancy/json</code>						
Metoda	POST						
URL Nagłówki	Content-Type: application/json Token: <session token> User: <user name>						
Parametry	<p>Przykład:</p> <pre>{ parkingID: [integer], name: [string], category: [string], type: [string], capacity: [string], trend: [string], freePlaces: [integer], countCarIn: [integer], countCarOut: [integer], time: [datetime], measureTime: [datetime], forecastFreePlaces: [integer], information: [string] }</pre> <p>Przykład:</p> <pre>{ parkingId: "1103", name: "Hala Stulecia - parking", category: "ZAMKNIETY", type: "CALODOBOWY", capacity: "150", trend: "BEZ ZMIAN", freePlaces: "128", countCarIn: "3", countCarOut: "3", time: "2017-01-23T13:22:12.153", measureTime: "2017-01-23T13:17:12.153", forecastFreePlaces: "128", information: "Dodatkowe istotne informacje" }</pre> <p>Parametry:</p> <table border="1"> <tr> <td>parkingId</td> <td>Identyfikator parkingu w systemie ITS, nadawany przez operatora systemu</td> </tr> <tr> <td>name</td> <td>Nazwa parkingu (maksymalnie 100 znaków)</td> </tr> <tr> <td>category</td> <td>Kategoria dostępności miejsc parkingowych. Przyjmowane parametry:</td> </tr> </table>	parkingId	Identyfikator parkingu w systemie ITS, nadawany przez operatora systemu	name	Nazwa parkingu (maksymalnie 100 znaków)	category	Kategoria dostępności miejsc parkingowych. Przyjmowane parametry:
parkingId	Identyfikator parkingu w systemie ITS, nadawany przez operatora systemu						
name	Nazwa parkingu (maksymalnie 100 znaków)						
category	Kategoria dostępności miejsc parkingowych. Przyjmowane parametry:						

	<p>OTWARTY - plac parkingowy niezadaszony</p> <p>ZAMKNIETY - parking kubaturowy zadaszony</p> <p>DLA_ABONENTOW - parking dostępny wyłącznie dla osób posiadających wykupiony abonament</p>
type	<p>Typ parkingu, określony poprzez parametry:</p> <p>CALODOBOWY - parking dostępny całodobowo</p> <p>OGRANICZONY - parking ograniczony czasowo</p>
capacity	Ogólna ilość miejsc parkingowych dostępnych do zaparkowania (również miejsc objętych abonamentami)
trend	<p>Trend zmian zajętości miejsc parkingowych.</p> <p>Przyjmowane parametry:</p> <p>WZRATAJACY - zwiększająca się zajętość miejsc parkingowych</p> <p>MALEJACY - zmniejszająca się zajętość miejsc parkingowych</p> <p>BEZ ZMIAN - nie zmieniała się zajętość miejsc</p> <p>N/A - trend nie może być określony</p>
freePlaces	Aktualna liczba wolnych miejsc parkingowych - dostępnych do zaparkowania, z wyłączeniem miejsc abonamentowych
countCarIn	Ilość pojazdów wjeżdżających w czasie od początku pomiaru do zarejestrowania pomiaru
countCarOut	Ilość pojazdów wyjeżdżających w czasie od początku pomiaru do zarejestrowania pomiaru
time	Czas zarejestrowania pomiaru UTC w formacie YYYY-MM-DDTHH:MM:SS.SSS
meassureTime	Czas początku pomiaru UTC w formacie YYYY-MM-DDTHH:MM:SS.SSS
forecastFreePlaces	Prognoza ilości wolnych miejsc parkingowych na godzinę do przodu
information	Dodatkowe istotne informacje, które należy przekazać dla osób planujących parkowanie (maksymalnie 1000 znaków)
Odpowiedź (sukces)	<p>Odpowiedź:</p> <p>Code: 200 OK</p> <p>Content: { id : 1111111 }</p> <p>Parametry:</p>

	<p>id: unikalny identyfikator zarejestrowanej lokalizacji w repozytorium danych</p>
<p>Odpowiedź (błąd)</p>	<p>Odpowiedź:</p> <p>Code: 401 Unauthorized Content: <pre>{ message: "User not authorized" } { message: "Invalid token" } { message: "Session token Expired" }</pre></p> <p>Code: 415 Unsupported media type Content: { message : "Invalid content-type" }</p> <p>Code: 50X Server error Content: <pre>{ message : "Server error" } { message : "Database error" }</pre></p>
<p>Przykład wywołania</p>	<pre>curl -X POST -H "token: 36f...062" -H "user: test_user" -H "content-type: application/json" -d "{parkingId: "1103", ...}" http://<ip>:<port>/v2/parking/occupancy/json</pre>
<p>Komentarz</p>	<ul style="list-style-type: none"> - Kodowanie znaków UTF-8 - Przekazywane informacje muszą być z gradacją 5 minutową