

# Prezentacja AMODIT WebAPI z użyciem Postman

Instrukcja konfiguracji wywołań metod AMODIT WebAPI z  
wykorzystaniem aplikacji Postman

# Aplikacja Postman

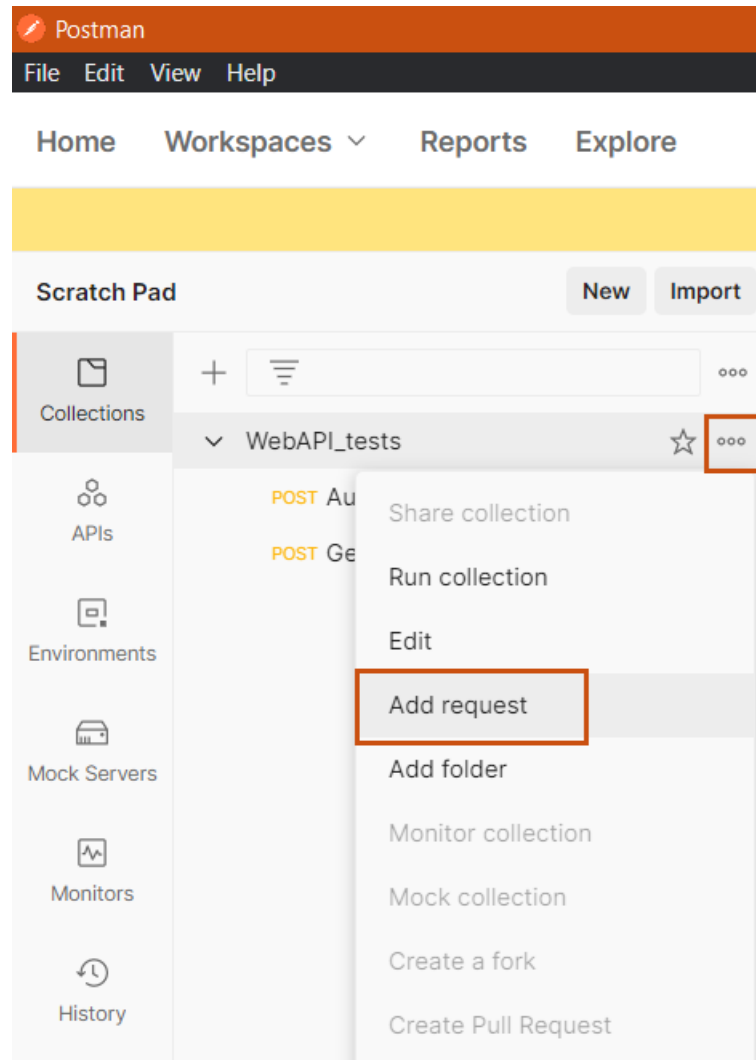
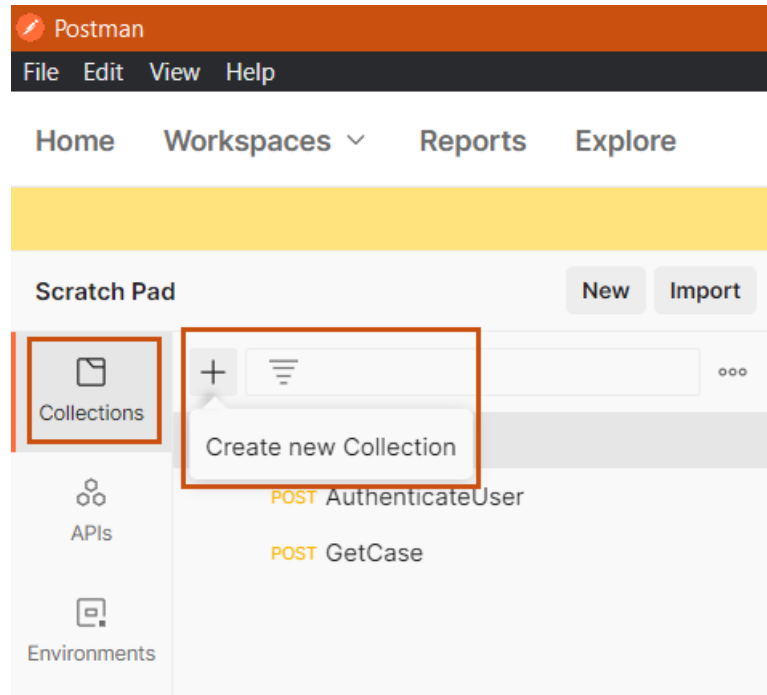
- Aplikację Postman można pobrać z tej lokalizacji:

<https://www.postman.com/downloads/>



***UWAGA!*** *Celem niniejszej prezentacji nie jest pokazanie możliwości aplikacji Postman jako takiej. Prezentacja skupia się na przedstawieniu przykładowej konfiguracji i użycia aplikacji Postman do pokazania działania AMODIT WebAPI.*

# Konfiguracja wywołania metody AMODIT WebAPI (1)



W aplikacji Postman utworzyć, o ile nie zrobiono tego wcześniej, nową kolekcję (collection) i dodać pierwsze żądanie (request), czyli wywołanie metody AMODIT WebAPI.

# Konfiguracja wywołania metody AMODIT WebAPI (2)

The screenshot shows the Postman interface with the following configuration:

- Collection:** WebAPI\_tests / UpdateCaseFields2 [1]
- Method:** POST [2]
- URL:** https://elbedemo.amodit.com/webserviceanon/amodwebapi.asmx [3]
- Headers:** Headers (8) [4]
- Header Configuration:**

Key	Value
Content-Length	0
Host	<calculated when request is sent>
User-Agent	PostmanRuntime/7.28.4
Accept	*/*
Accept-Encoding	gzip, deflate, br
Connection	keep-alive
Content-Type	text/xml; charset=utf-8 [5]
- Action:** Save [6]

[1] Nadać nazwę, np. nazwę wywoływanej metody.

[2] Ustawić typ żądania – w przypadku AMODIT WebAPI będzie to typ „POST”.

[3] Wpisać lokalizację AMODIT WebAPI.

[4] Przejsz do konfiguracji nagłówka żądania.

[5] Dodać parametr „Content-Type” z wartością jak na obrazku.

[6] Zapisać zmiany.

# Konfiguracja wywołania metody AMODIT WebAPI (3)

AMODWebAPI Usługa sieci Web x eLBedemo: Sprawa: Proces testo x +

elbedemo.amodit.com/webseviceanon/amodwebapi.asmx?op=UpdateCaseFields2 [7]

Aplikacje Do przeczytania

## AMODWebAPI

Kliknij [tutaj](#), aby uzyskać pełną listę operacji.

### UpdateCaseFields2 [8]

**Test**

Formularz testowy jest dostępny tylko dla żądań z komputera lokalnego.

**SOAP 1.1**

Poniżej zamieszczono przykładowe żądanie i odpowiedź SOAP 1.1. Zamiast symboli zastępczych należy podać rzeczywiste wartości.

```
POST /webseviceanon/amodwebapi.asmx HTTP/1.1
Host: elbedemo.amodit.com
Content-Type: text/xml; charset=utf-8
Content-Length: length
SOAPAction: "http://tempuri.org/UpdateCaseFields2"

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Header>
    <AMODWebAPIHeader xmlns="http://tempuri.org/">
      <Username>string</Username>
      <Password>string</Password>
      <AuthenticatedToken>string</AuthenticatedToken>
    </AMODWebAPIHeader>
  </soap:Header>
  <soap:Body>
    <UpdateCaseFields2 xmlns="http://tempuri.org/">
      <caseId>int</caseId>
      <fieldNames>
        <string>string</string>
        <string>string</string>
      </fieldNames>
      <fieldsValues>
        <string>string</string>
        <string>string</string>
      </fieldsValues>
    </UpdateCaseFields2>
  </soap:Body>
</soap:Envelope>
```

[9]

[7] W przeglądarce zalogować się do systemu AMODIT i przejść do specyfikacji WebAPI.

[8] Wybrać metodę, której wywołanie (żądanie) jest konfigurowane w Postman.

[9] Skopiować do schowka XML tzw. body request.

# Konfiguracja wywołania metody AMODIT WebAPI (4)

The screenshot shows the Postman interface with the following configuration:

- Method: POST
- URL: `https://elbedemo.amodit.com/webserviceanon/amodwebapi.asmx`
- Body tab selected (labeled [10])
- Editor type: raw (labeled [11])
- Format: XML (labeled [12])
- XML content (labeled [13]):

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
3   <soap:Header>
4     <AMODWebAPIHeader xmlns="http://tempuri.org/">
5       <Username>string</Username>
6       <Password>string</Password>
7       <AuthenticatedToken>string</AuthenticatedToken>
8     </AMODWebAPIHeader>
9   </soap:Header>
10  <soap:Body>
```

[10] Przejsć do zakładki „Body”.

[11] Wybrać typ edycji „raw” ...

[12] ... i format „XML”.

[13] Wkleić zawartość XML z pkt. [9]

# Konfiguracja wywołania metody AMODIT WebAPI (5)

Postman

File Edit View Help

Home Workspaces Reports Explore

Search Postman

Sign In Create Account

Working locally in Scratch Pad. Switch to a Workspace

Scratch Pad

New Import

Overview

WebAPI\_tests / UpdateCaseFields2

POST AuthenticateU... POST GetC...

POST UpdateCaseFields2

POST https://elbedemo.amodit.com/webserviceanc...

Params Authorization Headers (9) Body Pre-req

none form-data x-www-form-urlencoded raw

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
3   <soap:Header>
4     <AMODWebAPIHeader xmlns="http://tempuri.org/">
5       <Username>webapi_user</Username> [14]
6       <Password>.....</Password>
7       <AuthenticatedToken>1b4c0370-59da-48be-aba6-41b7e6a77bc1</AuthenticatedToken> [15]
8     </AMODWebAPIHeader>
9   </soap:Header>
10  <soap:Body>
```

[14] W sekcji nagłówkowej XML'a żądania wprowadzić nazwę użytkownika i login.

**UWAGA!** Musi to być użytkownik systemowy w AMODIT.

[15] Wprowadzić „AuthenticatedToken”, który można pozyskać wywołując metodę AuthenticateUser. (patrz slajdy „Autoryzacja w AMODIT WebAPI” w dalszej części prezentacji).

# Konfiguracja wywołania metody AMODIT WebAPI (6)

[16] W sekcji treści (body) XML'a wprowadzić wartości atrybutów dla wywoływanej metody (w tym przykładzie jest to metoda UpdateCaseFields2).

[17] Wywołać żądanie na serwerze AMODIT wciskając przycisk „Send”.

The screenshot shows the Postman interface with a REST client tab selected. The URL is `http://localhost:3000/webbedemo.amodit.com/webserviceanon/amodwebapi.asmx`. The request body is set to XML format. The XML content is as follows:

```
10 <soap:Body>
11   <UpdateCaseFields2 xmlns="http://tempuri.org/">
12     <caseId>792</caseId>
13     <fieldNames>
14       <string>Tekstowe (krótki tekst) 1</string>
15       <string>Tekstowe (długi tekst) 1</string>
16       <string>Numeryczne 1</string>
17     </fieldNames>
18     <fieldsValues>
19       <string>krótki tekst wstawiony przez WebAPI</string>
20       <string>i jakiś taki długi ten tekst, również wstawiony przez WebAPI</string>
21       <string>78.9</string>
22     </fieldsValues>
23   </UpdateCaseFields2>
24 </soap:Body>
25 </soap:Envelope>
```

The XML content is highlighted with a red box, and a label [16] is placed to its right. The 'Send' button is also highlighted with a red box, with a label [17] placed below it. The interface also shows a 'Cookies' section with a [17] label.



# Odczytywanie odpowiedzi i weryfikacja działania metody AMODIT WebAPI (1)

The screenshot shows the Postman interface with a REST client configured for a SOAP endpoint. The request is a POST to `https://elbedemo.amodit.com/webserviceanon/amodwebapi.asmx`. The body is a SOAP message with the following structure:

```
10 <<soap:Body>
11 <UpdateCaseFields2 xmlns="http://tempuri.org/">
12 <caseId>792</caseId>
13 <fieldNames>
  <string>Tekstowe (krótki tekst) 1</string>
  <string>Tekstowe (długi tekst) 1</string>
  <string>Numeryczne 1</string>
</fieldNames>
  <fieldValues>
  </fieldValues>
</UpdateCaseFields2>
</soap:Body>
```

The response status is `200 OK` with a time of 295 ms and a size of 638 B. The response body is a SOAP message with the following structure:

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema">
3 <soap:Body>
4 <UpdateCaseFields2Response xmlns="http://tempuri.org/">
5 <UpdateCaseFields2Result>
6 <Response>success</Response>
7 </UpdateCaseFields2Result>
8 </UpdateCaseFields2Response>
9 </soap:Body>
10 </soap:Envelope>
```

Annotations in the image include:

- [18] Wynik wywołania przykładowej metody UpdateCaseFields2. (Points to the response body content)
- [19] Status powodzenia wysłania żądania. (Points to the status bar)

# Odczytywanie odpowiedzi i weryfikacja działania metody AMODIT WebAPI (2)

Proces testowy (792) Łukasz Bott | Etap 1 | Proces testowy

SEKCJA NR 1

Tekstowe (krótki tekst) 1  
krótki tekst wstawiony przez WebAPI

Tekstowe (długi tekst) 1  
i jakiś taki długi ten tekst, również wstawiony przez WebAPI

Numeryczne 1  
78,9

plik na formularzu głównym  
 Przeciągnij plik tutaj

Kopiuj plik do tabeli

Efekt wywołania przykładowej metody UpdateCaseFields2 po stronie sprawy w AMODIT:  
[20] Wartości pól wprowadzone przez użytkownika przed wywołaniem wspomnianej metody.  
[21] Rezultat wywołania wspomnianej metody za pomocą aplikacji Postman.

Tekstowe (krótki tekst) 1: to Łukasz Bott  
jest pole z krótkim tekstem 2022-01-10 09:49  
Tekstowe (długi tekst) 1:  
Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut quis aliquet urna, ut eleifend ante. Maecenas vitae sagittis nunc. Morbi quis tincidunt libero. Sed odio odio, pulvinar fermentum sapien at, porttitor dapibus tellus. Nulla tempus nibh a elit aliquam, sit amet bibendum elit lacinia. Sed velit purus, pharetra eget malesuada non, venenatis eget magna. Mauris luctus eu elit at scelerisque. [20]  
Numeryczne 1: 1 234,56

Tekstowe (krótki tekst) 1: WebAPI User  
krótki tekst wstawiony 2022-01-10 10:42  
przez WebAPI  
Tekstowe (długi tekst) 1: i jakiś taki długi ten tekst, również wstawiony przez WebAPI  
Numeryczne 1: 78,9

[21]

# Autoryzacja w AMODIT WebAPI

The screenshot shows the Postman interface with a SOAP request for the AuthenticateUser method. The request body is an XML SOAP envelope containing a header with AMODWebAPIHeader, Username (webapi\_user), Password, and AuthenticatedToken. The response is a SOAP envelope containing an AuthenticateUserResponse with an AuthenticateUserResult attribute value.

WebAPI\_tests / AuthenticateUser

POST `https://elbedemo.amodit.com/webserviceanon/amodwebapi.asmx` **Send**

Params Authorization Headers (9) **Body** Pre-request Script Tests Settings

● none ● form-data ● x-www-form-urlencoded ● raw ● binary ● GraphQL XML **Beautify**

```
<?xml version='1.0' encoding='utf-8' ?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Header>
    <AMODWebAPIHeader xmlns="http://tempuri.org/">
      <Username>webapi_user</Username> [23]
      <Password>...</Password>
      <AuthenticatedToken></AuthenticatedToken> [24]
    </AMODWebAPIHeader>
  </soap:Header>
  <soap:Body>
    <AuthenticateUser xmlns="http://tempuri.org/" />
  </soap:Body>
</soap:Envelope>
```

Cookies Headers (7) Test Results Status: 200 OK Time: 141 ms Size: 642 B **Save Response**

```
<?xml version='1.0' encoding='utf-8' ?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <soap:Body>
    <AuthenticateUserResponse xmlns="http://tempuri.org/" [26]
      <AuthenticateUserResult>1b4c0370-59da-48be-aba6-41b7e6a77bc1</AuthenticateUserResult>
    </AuthenticateUserResponse>
  </soap:Body>
</soap:Envelope>
```

[22] Skonfigurować żądanie dla metody AuthenticateUser WebAPI.

[23] Podać login i hasło użytkownika.

[24] Nie podawać(!!!) wartości atrybutu „AuthenticatedToken” metody.

[25] Wywołać metodę klikając przycisk „Send”.

[26] Z rezultatu wywołania odczytać (skopiować) wartość atrybutu „AuthenticateUserResult”.

UWAGA! Tej właśnie wartości używa się przy wywołaniu innych metod WebAPI. Ważność tokenu autentykacji wynosi ok. 15 minut.