

WYKAZ UMÓW – INSTRUKCJA DLA PRZEDSIĘBIORCÓW TURYSTYCZNYCH

Wywołanie usług API poprzez klienta usług REST (na przykładzie aplikacji Bruno)

Obligatoryjne komponenty do wywołania usług TFG API:

1. Konto techniczne wystawione przez TFG dla Przedsiębiorcy Turystycznego (nazwa użytkownika i hasło)
2. Certyfikat przekazany na adres mailowy z wniosku o konto techniczne
3. Hasło do archiwum ZIP z certyfikatem, wysłane SMSem na numer telefonu z wniosku o konto techniczne
4. IP publiczne/lub zakres IP z jakiego będą wywoływane usługi musi być tożsame z tym zgłoszonym na wniosku o konto techniczne
5. Aplikacja do wywołania usług API – na potrzeby dokumentu użyto bezpłatnej aplikacji Bruno ([Bruno - The Git-Native API Client](#))

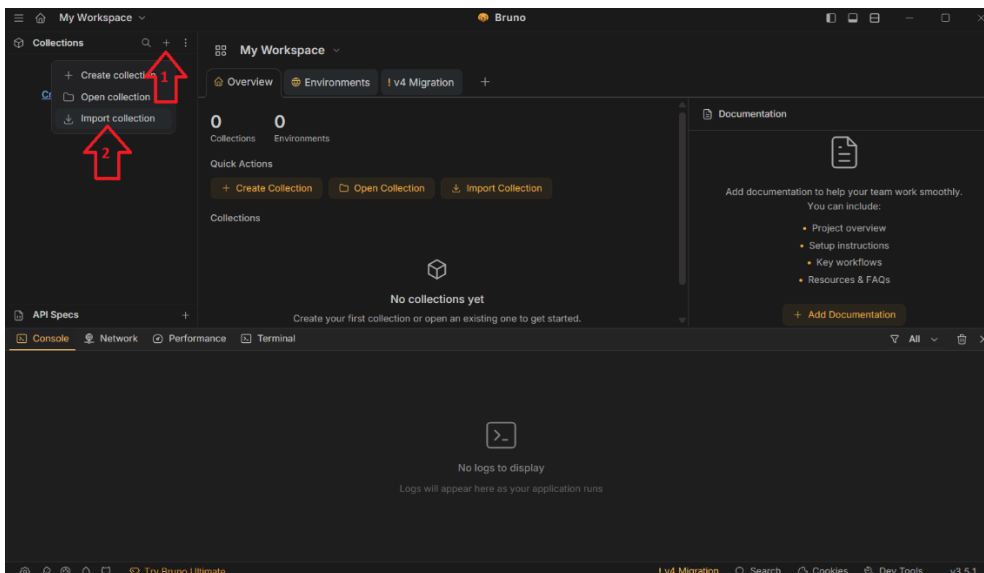
Pliki użyte do stworzenie projektu dostępowego do API TFG w aplikacji Bruno znajdują się na stronie TFG: tfg.ufg.pl > [Baza Wiedzy](#) > [Jak korzystać z Portalu TFG](#):

1. Specyfikacja YAML API TFG - [OpenApi swagger umowy i zasilenia](#)
2. Pięć przykładów operacji załączników JSON dla API TFG- [Przykładowe pliki JSON](#)

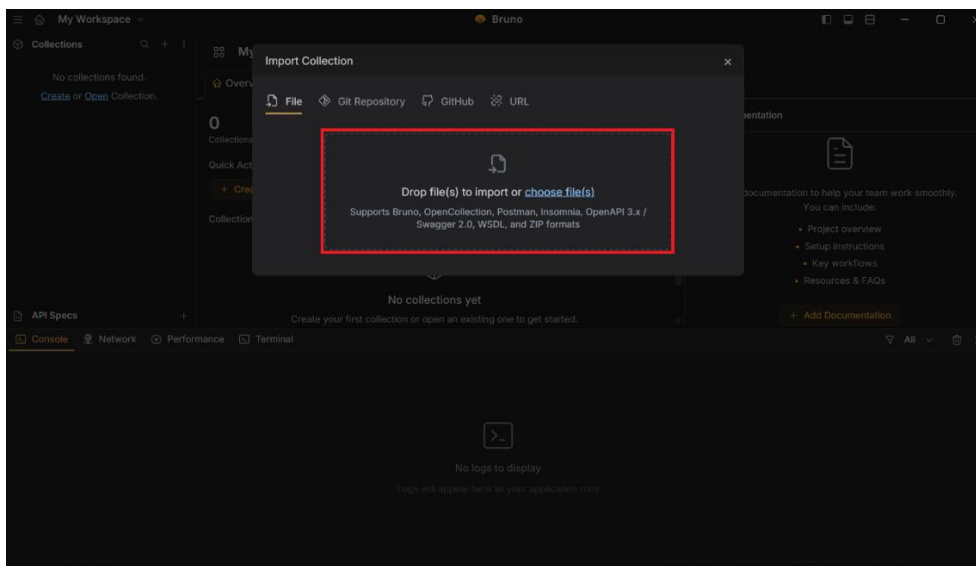
Instrukcja tworzenia kolekcji usług w aplikacji Bruno:

Należy uruchomić aplikację Bruno. Po uruchomieniu aplikacji Bruno należy wywołać w kolejności jak na poniższym printscreenie:

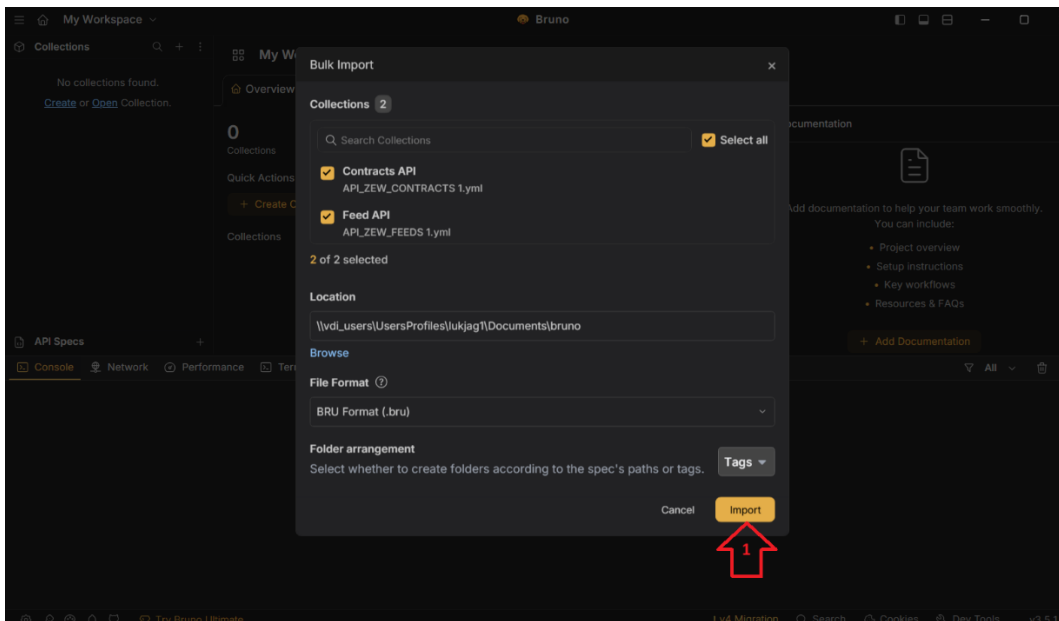
1. Symbol „+”
2. Import collection



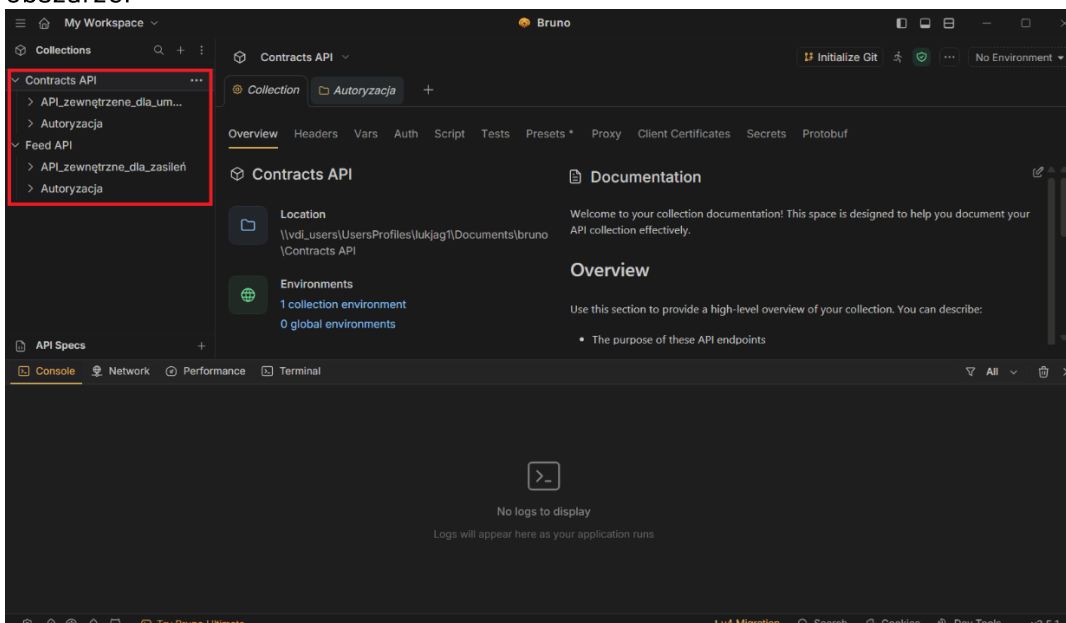
Następnie wyświetli się okno jak na poniższym printscr. W obszar zaznaczony na czerwono należy wrzucić specyfikacje plików YAML API TFG (link do specyfikacji [OpenApi swagger umowy i zasilenia](#)).



W kolejnym etapie należy kliknąć w przycisk Import wskazany strzałką 1 na poniższym printscr.



Zaimportowane kolekcje powinny pojawić się w lewym menu jak na printscr poniżej w zaznaczonym obszarze:



Następnie należy zaimportować certyfikaty co ważne dla obydwu kolekcji Contracts API i Feed API. W archiwum certyfikatów wysłanym przez UFG znajdują się certyfikaty w formatach .p12, .jks i .pem. Na potrzeby aplikacji Bruno użyjemy formatu pem. W jednym pliku XXXX.pem znajdują się zarówno certyfikaty jak i klucz prywatny. Ponieważ jednak aplikacja Bruno wymaga oddzielnych plików dla certyfikatów i klucza należy je rozdzielić tworząc dwa oddzielne pliki:

1. Wszystkie certyfikaty z pliku XXXX.pem pomiędzy sekcjami:
-----BEGIN CERTIFICATE----- i -----END CERTIFICATE-----

Plik certyfikatów powinien mieć poniższą strukturę:

```
-----BEGIN CERTIFICATE-----
c3NkZnNkZnNkZnNmc2ZzZGZzZANkZnNkZnNmc2ZkZnNmc2ZzZGZzZANkZnNmc2Zt
<wiele linii ciągów BASE64>
ZXdyZXJ3ZWN4dnhjZzQ1MzY0NWZkZ2RmdmRmaHRoanJ0Njc1Njc1Njc1NjczNDUz
-----END CERTIFICATE-----
-----BEGIN CERTIFICATE-----
bWMyWnpaRlp6WkFOa1puTmtabk5tYzJaa1puTm1jMlp6WkdaelpBTmtabk5tYzJa
<wiele linii ciągów BASE64>
G5oalp6UTFNelkwTldaa1oyUm1kbVJtYUhSb2FuSjBOamMxTmpjMU5qYzFOamN6T
-----END CERTIFICATE-----
-----BEGIN CERTIFICATE-----
c3NkZnNkZnNkZnNmYzJaelpHWnpaQU5rWm5Oa1puTm1jMlprWm5ObWMyWnpaRlp6
<wiele linii ciągów BASE64>
WlhkeVpYSjNaV040ZG5oalp6UQ1MzY0NWZkZ2RmdmRmaHRoanJ0Njc1Njc1Njc1N
-----END CERTIFICATE-----
```

2. Klucz prywatny pomiędzy sekcjami:
-----BEGIN RSA PRIVATE KEY----- i -----END RSA PRIVATE KEY-----

Plik z kluczem prywatnym RSA powinien mieć poniższą strukturę:

```
-----BEGIN RSA PRIVATE KEY-----
Proc-Type: 4, ENCRYPTED
DEK-Info: AES-256-CBC, A86517C359AA0CE79A0A9DF3CCC3D200

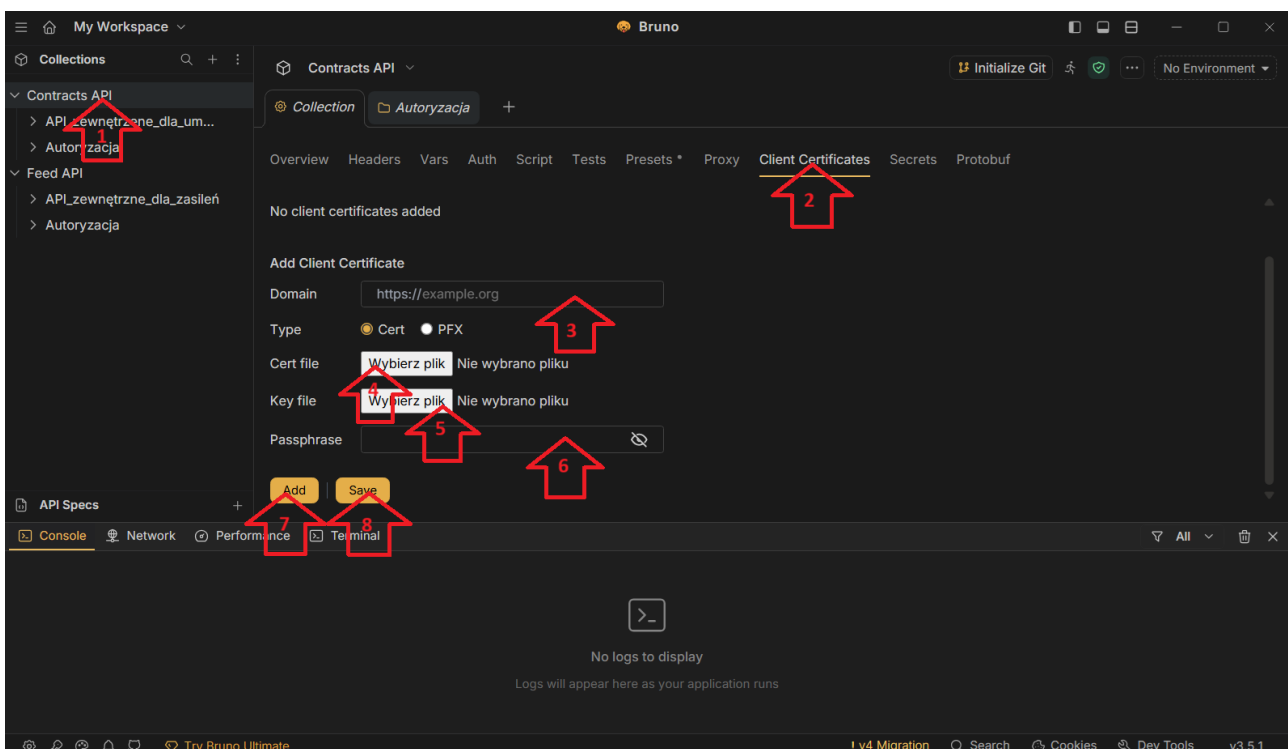
YzNOa1puTmtabk5rWm5ObWMyWnpaRlp6WkFOa1puTmtabk5tYzJaa1puTm1jMlp6
<wiele linii ciągów BASE64>
WlhkeVpYSjNaV040ZG5oalp6UTFNelkwTldaa1oyUm1kbVJtYUhSb2FuSjBOamMx
-----END RSA PRIVATE KEY-----
```

Tak przygotowane pliki można zaimportować do aplikacji Bruno w poniższy sposób. Klikamy w aplikacji Bruno i wykonujemy poniższe kroki (punkt 1,2,3,4,5,6,7 jednoznacznie mapuje się z poniższym printscr):

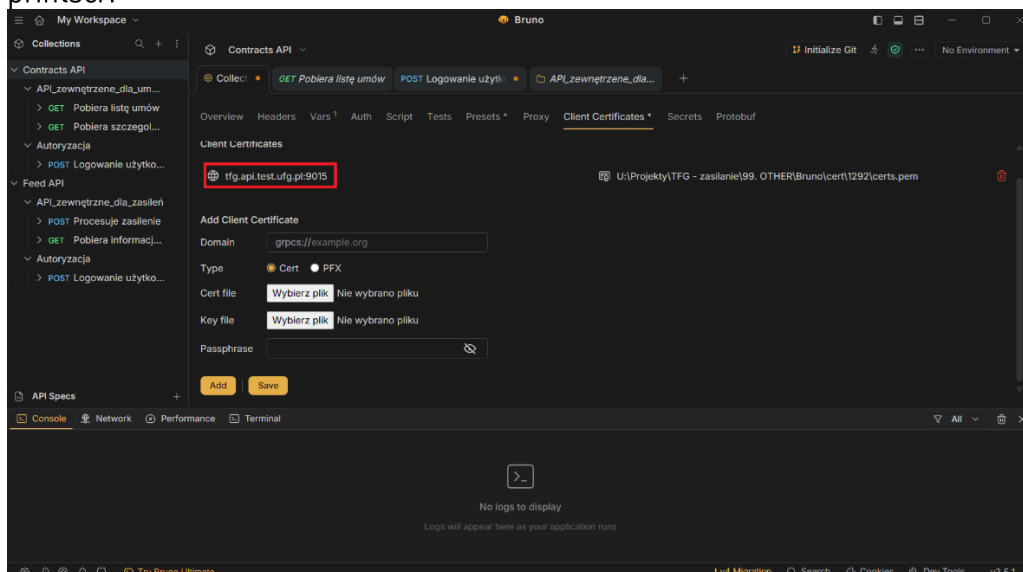
1. Klikamy na kolekcję „Contracts API”
2. Klikamy na „Client Certificates”
3. W polu „Domain” wpisujemy tfg.api.test.ufg.pl:9015 (w przypadku środowiska testowego)
4. Klikamy na przycisk „Cert file” i wybieramy plik z certyfikatami stworzony w poprzednim kroku
5. Klikamy na przycisk „Key file” i wybieramy plik z kluczem RSA stworzonym w poprzednim kroku
6. W pole „Passphrase” wpisujemy hasło do importu certyfikatu otrzymane na adres mailowy wpisany do wniosku o konto techniczne. **UWAGA:** jeżeli w to pole wpisujemy błędne hasło to aplikacja Bruno tego nie zweryfikuje a wywołanie usług nie będzie działać (podczas wywołania usługi pojawi się błąd OPENSSL_internal:SSLV3_ALERT_HANDSHAKE_FAILURE)
7. Klikamy na przycisk „Add”
8. Klikamy na przycisk „Save”

UWAGA: operacje należy powtórzyć dla drugiej kolekcji „Feed API”. Jeżeli tego nie wykonamy usługi z tej kolekcji nie potężą się z API TFG.

UWAGA: jeżeli przy imporcie certyfikatu do Bruno w pole „Passphrase” nie zostanie wpisane poprawne hasło to aplikacja tego nie zweryfikuje, a wywołanie usług nie będzie działać.



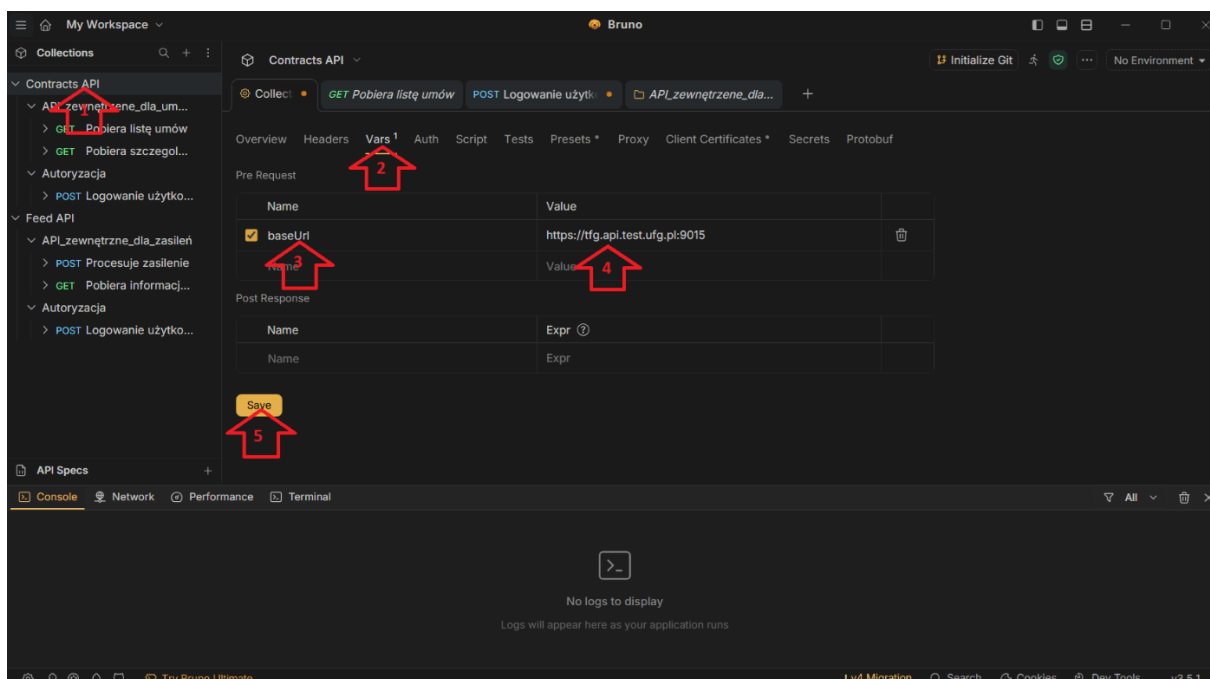
Po zaimportowaniu certyfikatu powinien pojawić się wpis zaznaczony na czerwono w poniższym printscr:



Uzupełnienie parametru baseUrl dla projektu. To ostatni krok przed wywoływaniem usług. Należy go uzupełnić zgodnie z poniższą instrukcją (mapowanie punktów jest bezpośrednio na poniższy printscr):

1. Klikamy na nagłówek kolekcji „Contracts API”
2. Klikamy na kontrolkę „Vars”
3. Klikamy na pustą komórkę w kolumnie „Name” i wpisujemy baseUrl (na poniższym printscr wartość jest już uzupełniona)
4. Klikamy na pustą komórkę w kolumnie „Value” i wpisujemy <https://tfg.api.test.ufg.pl:9015> (na poniższym printscr wartość jest już uzupełniona)
5. Klikamy „Save”

UWAGA: operacje trzeba powtórzyć również dla kolekcji „Feed API”. Jeżeli tego nie wykonamy usługi z kolekcji Feed API nie wywołają usług TFG API.



Po wykonaniu wszystkich kroków opisanych powyżej aplikacja Bruno umożliwi wywoływanie usług TFG API.

Poniżej przedstawiono dwie ścieżki wywołania usług:

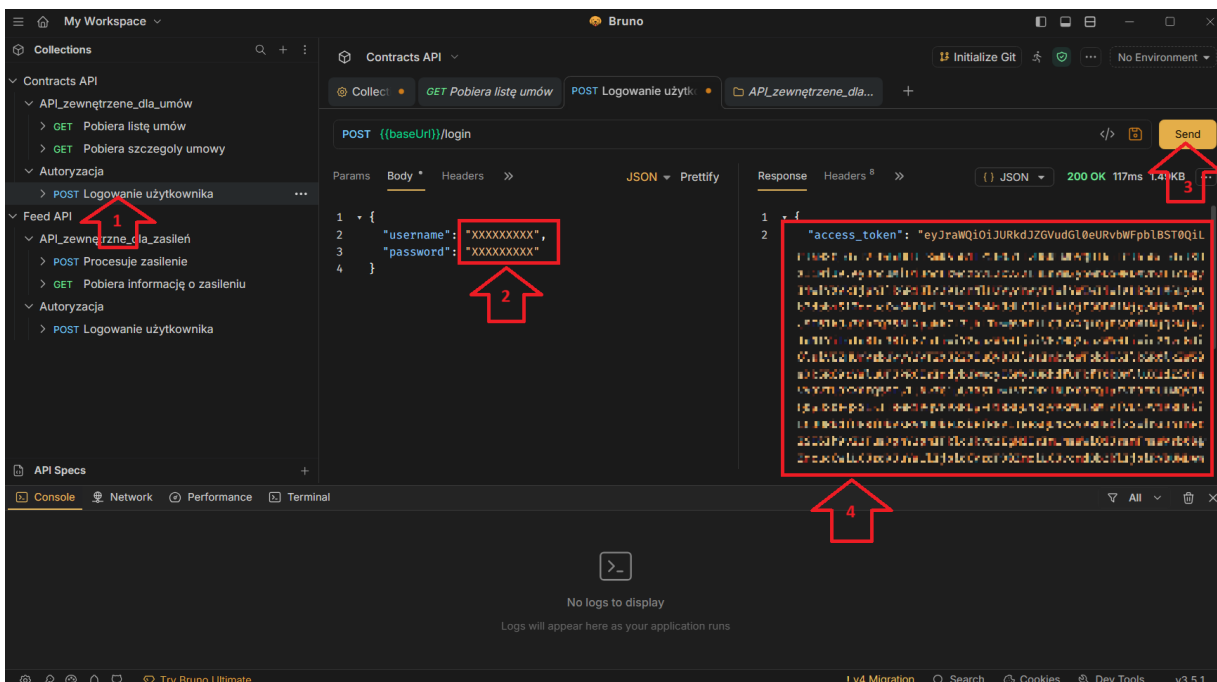
1. Pobranie wykazu umów
2. Zasilanie wykazem umów

Pobranie wykazu umów:

W Pierwszej kolejności należy wygenerować access token. W aplikacji Bruno należy wykonać następujące kroki:

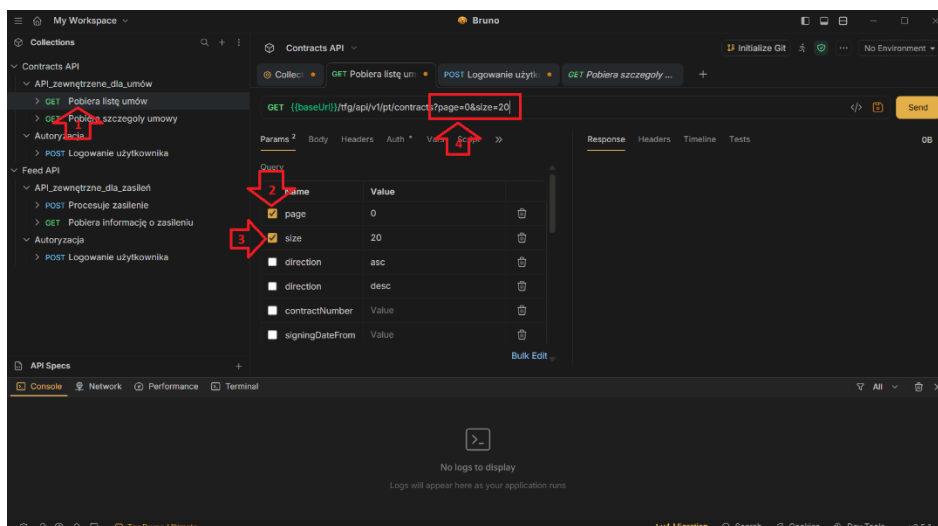
1. Wybrać z kolekcji „Contracts API” pozycję „Autoryzacja” -> „POST Logowanie użytkownika”
2. Uzupełnić nazwę użytkownika i hasło dla konta technicznego (w miejscu znaków XXXXXXXXX)
3. Kliknąć przycisk „Send”
4. W odpowiedzi powinien zostać zwrócony „access_token” – należy go skopiować – cały ciąg BASE64 bez znaków cudzysłowów na końcach (”).

UWAGA: Jeżeli w odpowiedzi usługi pojawi się błąd „Invalid client credentials” będzie to oznaczać błędnego użytkownika lub hasło, inne błędy mogą oznaczać błąd konfiguracji projektu w aplikacji Bruno.



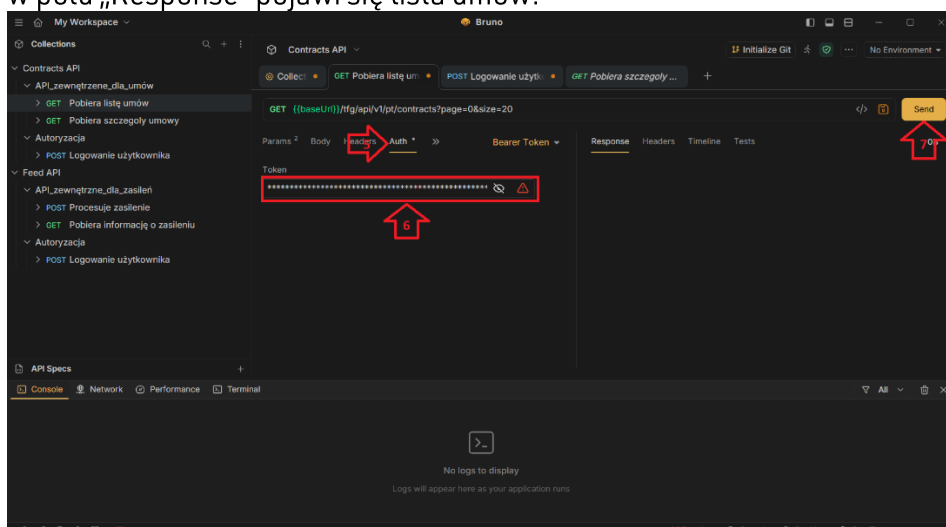
Pobranie wykazu umów. Należy wykonać następujące kroki aby pobrać wykaz umów:

1. Wybrać z kolekcji „Contracts API” pozycję „API_zewnętrzne_dla_umów” -> „GET Pobierz listę umów”
2. Należy odznaczyć i zaznaczyć tick „page” (należy to wykonać ponieważ niektóre wersje Bruno nie widzą parametrów po imporcie kolekcji z plików YAML)
3. Należy odznaczyć i zaznaczyć tick „size”
4. Zweryfikować w pasku adresu pojawiły się parametry na zasadzie ciągu znaków „?page=0&size=20”



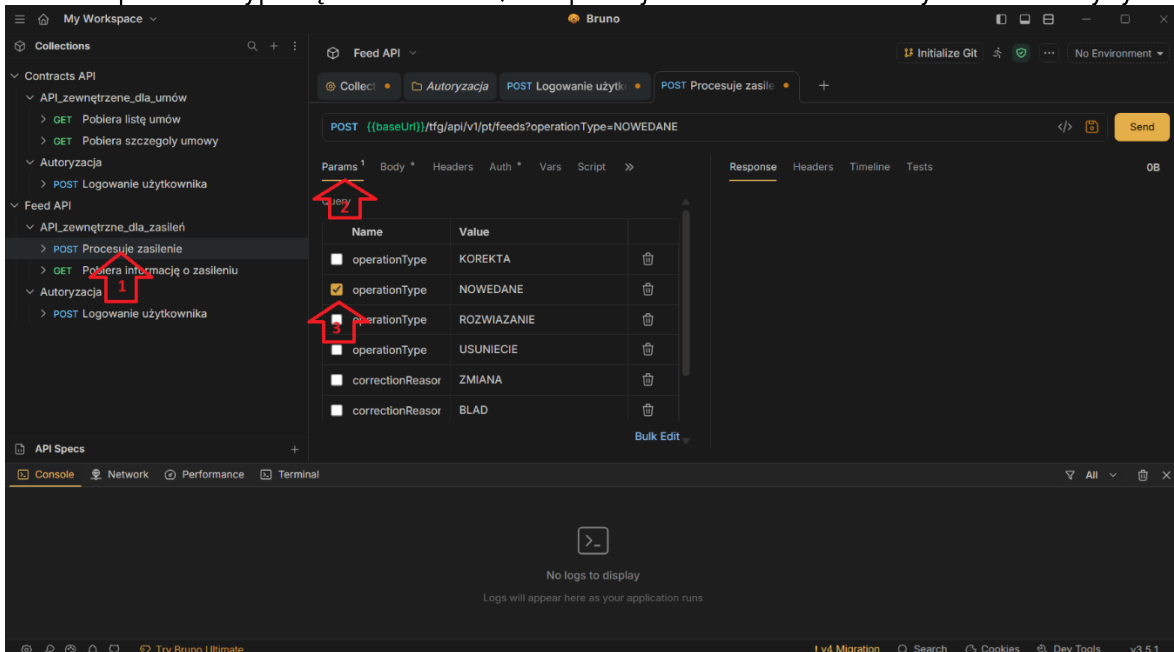
5. Należy kliknąć na kontrolkę „Auth”
6. W pole należy wkleić wygenerowany access token z usługi „Logowanie użytkownika”
7. Kliknąć na przycisk „Send”

Jeżeli wszystko zostało poprawnie skonfigurowane oraz został poprawnie wklejony access token to w polu „Response” pojawi się lista umów.

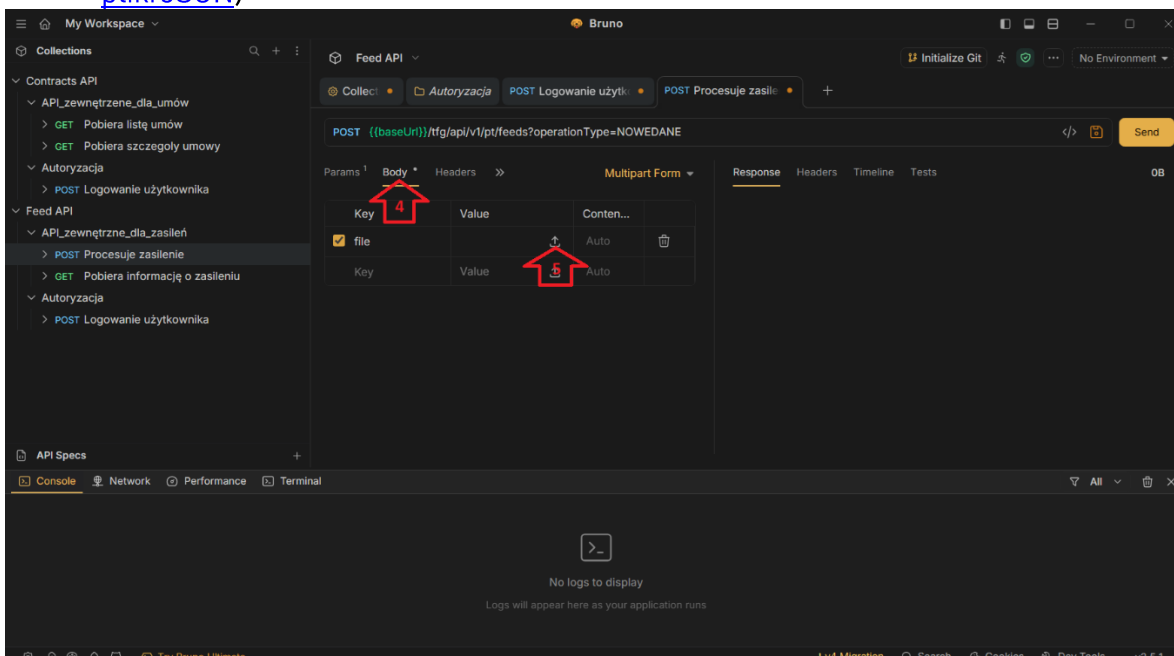


Wysyłka wykazu umów. Należy wykonać następujące kroki aby wysłać wykaz umów:

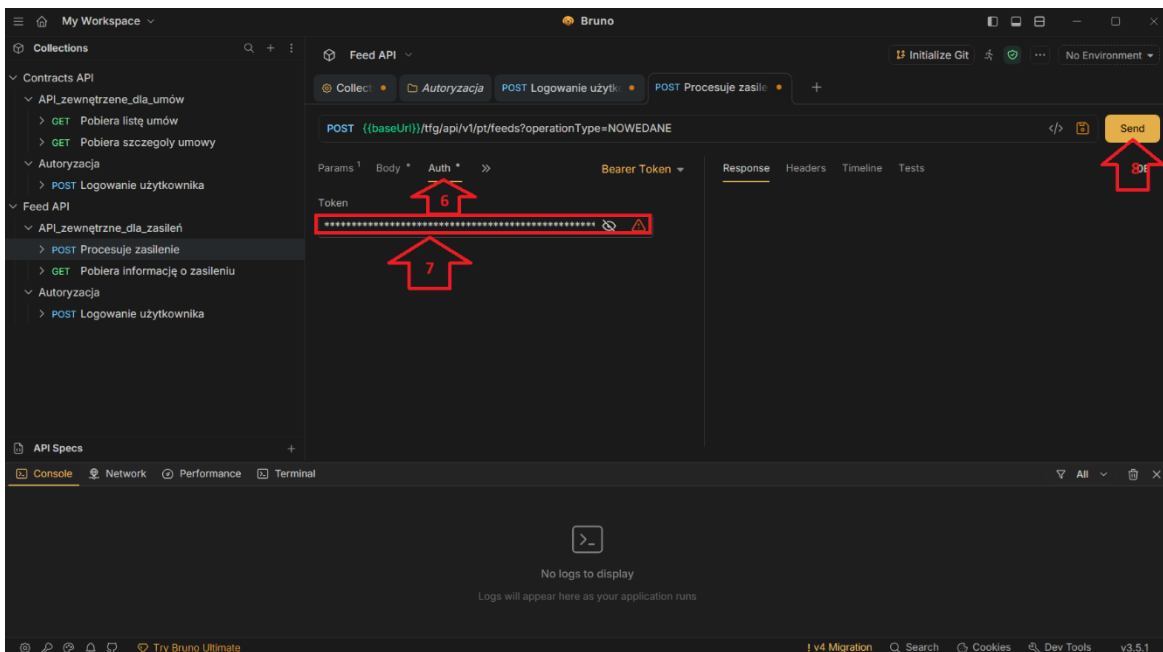
1. Wybrać z kolekcji „Feed API” pozycję „API_zewnętrzne_dla_zasilen” -> „POST Procesuje zasilenie
2. Wybrać kontrolkę „Params”
3. Zaznaczyć tick przy operationType=NOWEDANE oraz zweryfikować czy pozostałe operationType są odznaczone (dla operacji NOWEDANE może być zaznaczony tylko jeden)



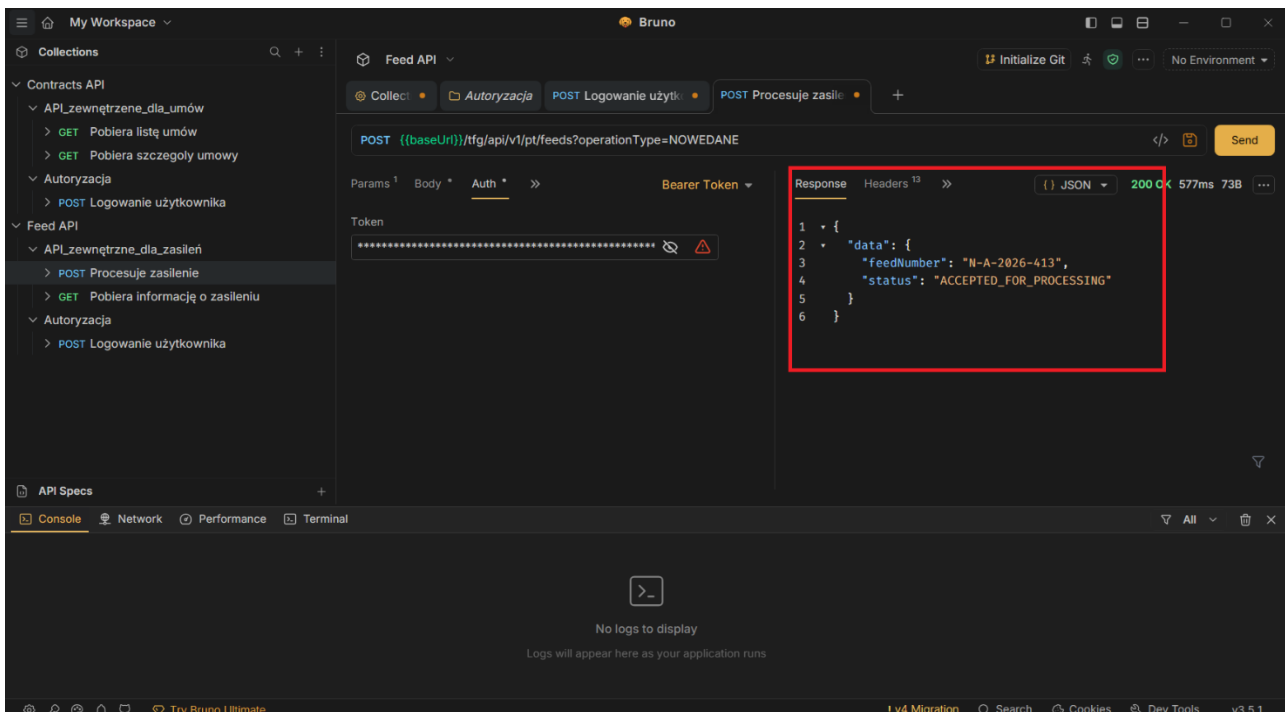
4. Wybrać kontrolkę „Body”
5. Wybrać symbol strzałki w górę i załączyć plik JSON z nowym wykazem umów ([Przykładowe pliki JSON](#))



6. Wybrać kontrolkę „Auth”
7. Wkleić access token w pole „Token”
8. Kliknąć przycisk „Send”



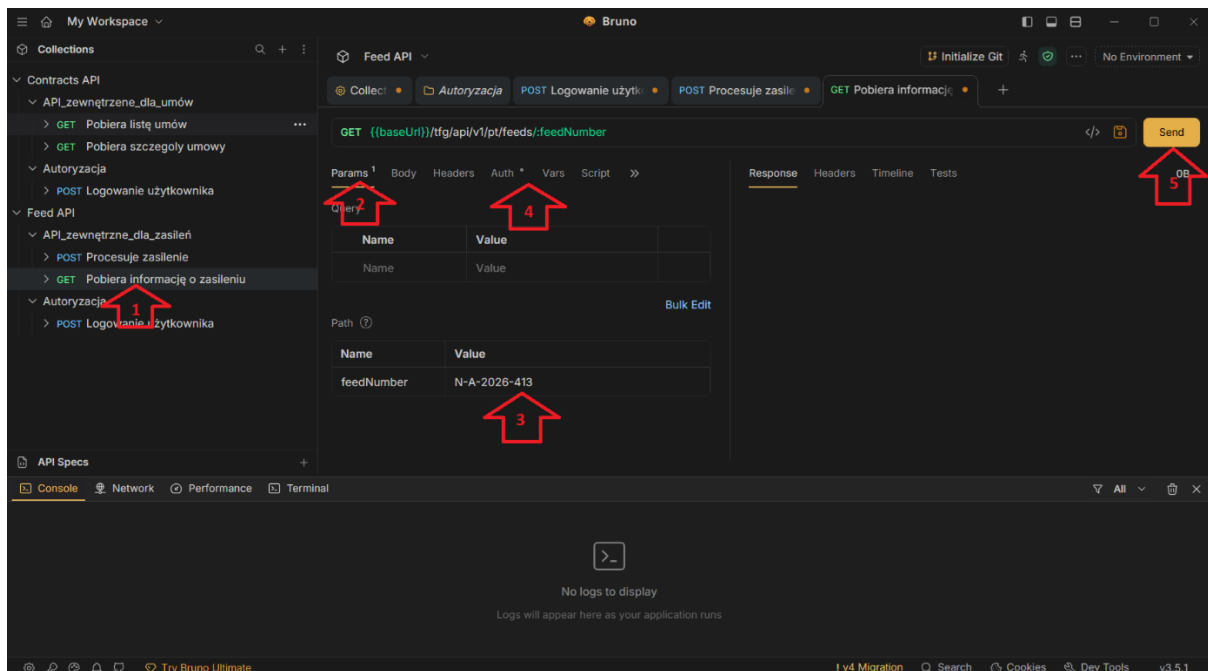
Jeżeli wszystkie kroki konfiguracyjne zostały wykonane poprawnie oraz został wklejony do żądania poprawny access token wtedy w polu „Response” pojawi się odpowiedź jak niżej:



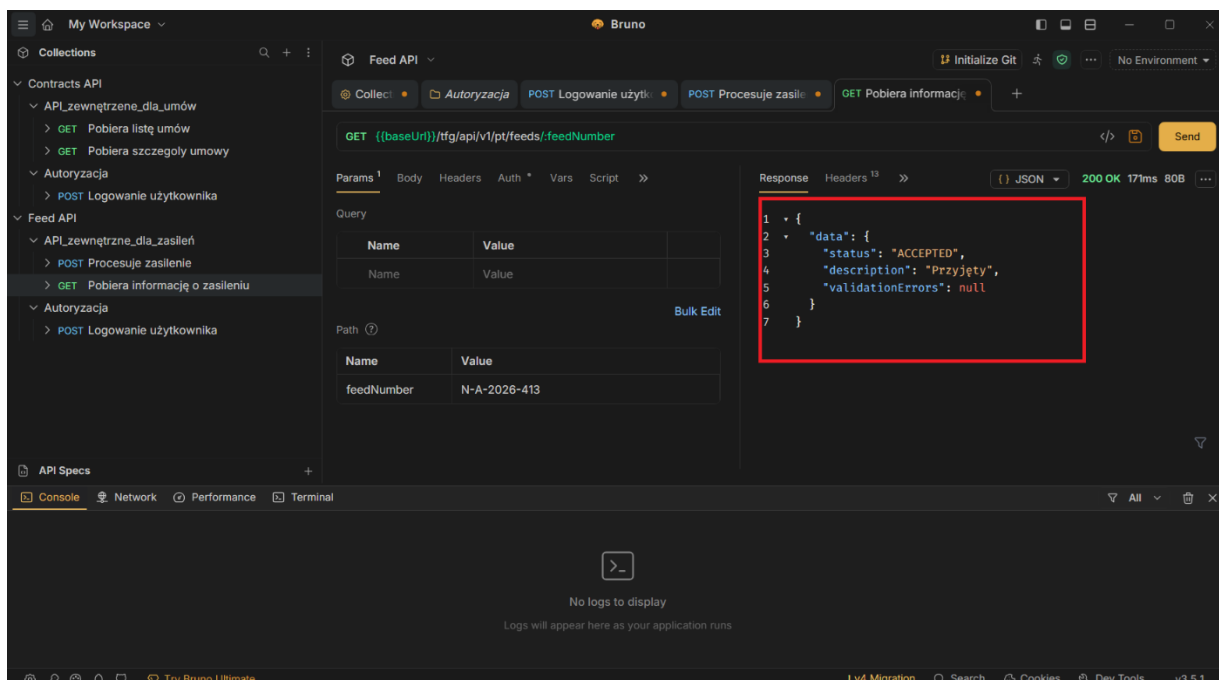
Pole „feedNumber” w powyższym printscr należy skopiować. Posłuży do weryfikacji przetwarzania poprzez wywołanie usługi pobrania informacji o zasileniu.

Należy wykonać następujące kroki aby pobrać informację o zasilaniu:

1. Wybrać z kolekcji „Feed API” pozycję „API_zewnętrzne_dla_zasileń” -> „GET Pobiera informacje o zasilaniu”
2. Kliknąć w kontrolkę „Params”
3. W polu „Value” parametru „feedNumber” wkleić wartość skopiowaną z odpowiedzi wywołania poprzedniej usługi
4. Kliknąć w kontrolkę „Auth” i wkleić w pole Token access token (jak w poprzednich wywołaniach)
5. Kliknąć przycisk „Send”



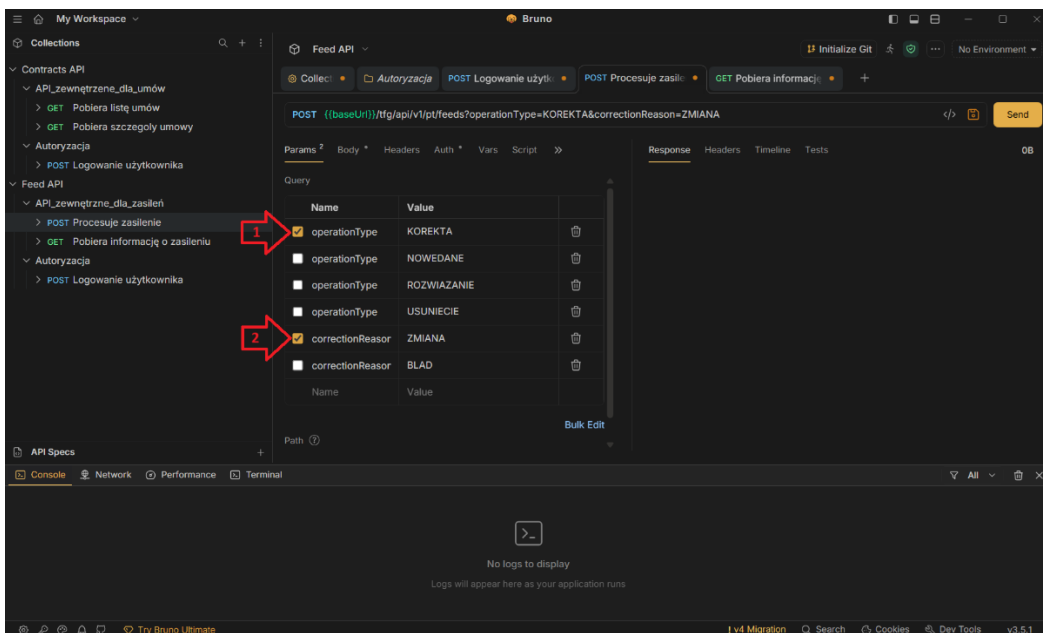
Poniżej przedstawiono odpowiedź wywołania usługi pobierającej informację o zasilaniu:



Pozostałe operacje wysyłki danych do API TFG wykonuje się w sposób analogiczny wybierając odpowiedni operationType z listy parametrów specyfikacji usługi oraz dodając plik JSON w odpowiedniej strukturze zgodnie z przykładami w [Przykładowe pliki JSON](#)

Dodatkowo w przypadku operacji operationType KOREKTA należy obligatoryjnie wybrać jedną z wartości correctionReason ZMIANA lub BLAD (jak na poniższych dwóch printscr):

1. Zaznaczyć tick operationType=KOREKTA
2. Zaznaczyć tick correctionReason=ZMIANA



1. Zaznaczyć tick operationType=KOREKTA
3. Zaznaczyć tick correctionReason=BLAD

