

Platforma Elektronicznego Fakturowania

PEFexpert
Platforma Elektronicznego Fakturowania

edi son **SOFTIQ**

Logowanie

W celu zarejestrowania się na platformie wymagane będzie podanie:

[Zaloguj](#)

Rejestracja

W celu zarejestrowania się na platformie wymagane będzie podanie:

danych podmiotu
danych osoby zarządzającej profilem podmiotu

W trakcie rejestracji nadany zostanie Ci ADRES PEF służący do adresowania dokumentów
Swoją tożsamość potwierdzisz z wykorzystaniem profilu zaufanego (wyłączone w wersji testowej)

[Zarejestruj nowe konto](#)

PRZEWODNIK IMPLEMENTACYJNY PEFEXPERT DLA DOSTAWCÓW APLIKACJI I USŁUG IT W ZAKRESIE API INTEGRACYJNEGO

Gliwice, 01.04.2019

SPIS TREŚCI

1. WSTĘP	3
2. DOKUMENTACJA TECHNICZNA	3
3. ŚRODOWISKA	3
4. UŻYTKOWNIK API	4
5. KONCEPCJA	4
6. AUTORYZACJA UŻYTKOWNIKA API	5
7. PRZYKŁADY WYWOŁAŃ I ODPOWIEDZI	6

1. WSTĘP

Niniejszy dokument zawiera opis interfejsu API do komunikacji Oprogramowania Użytkowników Platformy PEF z Systemem PEF. Powstał w celu ułatwienia procesu integracji nowych systemów z Systemem PEF.

Na początku omówiona zostanie koncepcja działania interfejsu. Następnie określona zostanie jego architektura oraz protokół komunikacji. Ważnym elementem opisu API jest jego techniczna dokumentacja przygotowana w popularnym formacie OpenAPI 3.0 (wspieranym przez narzędzia Swagger, usprawniające pracę z interfejsami w architekturze REST).

2. DOKUMENTACJA TECHNICZNA

Tak jak wspomniano we wstępie, dokumentacja techniczna przygotowana została w formacie OpenAPI 3.0. Można pobrać ją ze strony brokera (<https://brokerpefexpert.efaktura.gov.pl/>) lub za pośrednictwem Platformy PEF (<https://efaktura.gov.pl/dokumentacja>).

API platformy PEF jest zrealizowane z wykorzystaniem architektury REST. Protokołem wykorzystywanym do komunikacji pomiędzy Oprogramowaniem Użytkowników Platformy PEF a samym Systemem PEF jest protokół HTTPS.

3. ŚRODOWISKA

Broker PEFexpert udostępnia dwa środowiska:

- Produkcyjne:
 - Aplikacja WEB: <https://brokerpefexpert.efaktura.gov.pl>
 - API: <https://api.brokerpefexpert.efaktura.gov.pl>
- Integracyjne (przeznaczone do testów):
 - Aplikacja WEB: <https://int-brokerpefexpert.efaktura.gov.pl>
 - API: <https://api.int-brokerpefexpert.efaktura.gov.pl>

W obrębie jednego środowiska (produkcyjnego lub integracyjnego) odpowiednie kanały dostępu do platformy PEF (tzn. aplikacja webowa oraz API) są zsynchronizowane.

4. UŻYTKOWNIK API

Pierwszym krokiem w procesie integracji jest stworzenie specjalnego konta systemowego z dostępem do API Platformy PEF. W tym celu Super Użytkownik musi stworzyć nowego „Użytkownika API” poprzez platformę WEB. Proces ten jest bliźniaczo podobny do tworzenia konta standardowego Użytkownika – należy również pamiętać o nadaniu Użytkownikowi API dostępu do określonych adresów PEPPOL. Login Użytkownika API będzie służył w dalszej części jako „client id”, natomiast hasło to „client secret”. Super Użytkownik może również opcjonalnie ograniczyć dostęp do interfejsu API dla danego Użytkownika API z dokładnością do pojedynczych adresów IP.

5. KONCEPCJA

Koncepcja API opiera się na wykorzystaniu kolejki FIFO zawierającej komunikaty, które są pobierane i przetwarzane przez Oprogramowanie Użytkownika Platformy PEF. Każdy Użytkownik API ma swoją, niezależną od innych, kolejkę. Integrujące się systemy pobierają poszczególne komunikaty metodą „pull”, cyklicznie odpytując kolejkę (odstępów pomiędzy kolejnymi zapytaniami mogą być różne). W danej chwili czasu kolejka zwraca tylko jeden komunikat (ten, który pojawił się w kolejce jako pierwszy) oraz dodatkową informację o wielkości kolejki (tzn. liczbie komunikatów do odczytu, które znajdują się w kolejce). Informacja o wielkości kolejki służy umożliwieniu integrującym się systemom dostosowania częstotliwości odpytywania kolejki. Na kolejce mogą znajdować się następujące typy komunikatów:

- **Komunikat o otrzymaniu nowego dokumentu (`documentReceivedMessage`)** – wiadomość oznacza, że w systemie pojawił się nowy dokument adresowany do jednego z adresów PEPPOL przypisanych do Użytkownika API. Komunikat nie zawiera treści dokumentu – należy pobrać ją z wykorzystaniem niezależnej metody;
- **Komunikat o zmianie statusu otrzymanego dokumentu (`receivedDocumentStatusChangedMessage`)** – wiadomość oznacza, że otrzymany wcześniej dokument zmienił status (na odebrany lub odczytany). Użytkownik API powinien zaktualizować status dokumentu w swoim systemie przed oznaczeniem komunikatu jako przetworzony;
- **Komunikat o zmianie statusu wysłanego dokumentu (`sentDocumentStatusChangedMessage`)** – wiadomość oznacza, że dokument, którego nadawcą jest jeden z adresów PEPPOL przypisanych do konta Użytkownika API zmienił status (na przesłany, odebrany lub odczytany). Użytkownik API powinien zaktualizować status dokumentu w swoim systemie przed oznaczeniem komunikatu jako przetworzony;
- **Komunikat o błędzie w trakcie wysyłki dokumentu (`sentDocumentErrorMessage`)** – wiadomość oznacza, iż proces wysyłki dokumentu przez Użytkownika API nie powiódł się. Szczegółowy opis przyczyny niepowodzenia znajduje się w treści wiadomości – częste

powody nieudanej wysyłki to np. błędnie wskazany adresat lub dokument nieprzechodzący walidacji biznesowych;

- **Komunikat informujący o fakcie wysłania dokumentu innym kanałem (documentSentFromOtherSourceMessage)** – komunikat służy do synchronizacji informacji pomiędzy kanałem API a aplikacją webową oraz desktopową. Oznacza, że za pośrednictwem systemu wysłano dokument (poprzez aplikację webową lub desktopową), którego nadawcą jest adres PEPPOL przypisany do Użytkownika API. W ramach przetwarzania komunikatu należy pobrać treść takiego dokumentu i zapisać ją w systemie.

Do wysyłania dokumentów została wydzielona osobna metoda API.

Oprogramowanie Użytkownika Platformy PEF powinno pobierać kolejne komunikaty z kolejki, przetwarzać je we własnym zakresie, a następnie oznaczać komunikaty jako przetworzone (ten krok jest niezbędny – w przypadku jego pominięcia na kolejce wciąż widoczna będzie jedna, ta sama wiadomość). Takie oznaczenie jest realizowane jako osobna metoda API do usuwania komunikatów z kolejki. Co ważne – usunięcie z kolejki komunikatu o otrzymaniu nowego dokumentu jest jednoznaczne z oznaczeniem takiego dokumentu jako odebrany.

Komunikaty przechowywane w kolejce mają małą objętość, dlatego też treść dokumentów nie jest ich elementem. Do pobierania treści dokumentów (plików XML) służy osobna metoda.

Przychodzące dokumenty mogą w dalszej kolejności zostać odczytane przez Użytkownika końcowego (np. na formatce w Aplikacji Użytkownika Platformy PEF) – w API przewidziano możliwość oznaczania dokumentu przychodzącego jako odczytany, a służy temu niezależna wydzielona metoda.

6. AUTORYZACJA UŻYTKOWNIKA API

Do autoryzacji Użytkowników API wykorzystano OAuth 2.0 Client Credentials Flow, przeznaczony do autoryzacji pomiędzy systemami.

Dane niezbędne do pomyślnej autoryzacji generowane są przez Superużytkownika. Użytkownik API posługuje się danymi:

- Client ID – odpowiednik standardowego loginu;
- Client Secret – odpowiednik standardowego hasła.

Powyższe dane służą do uzyskiwania Access Tokena (dalej zwany AT), który służy do autoryzacji użytkowników w systemie. Aby otrzymać AT należy wykonać zapytanie pod odpowiedni adres metodą POST, podając parametr o kluczu „grant_type” i wartości „client_credentials”. Dane można przekazać na dwa sposoby: pierwszy polega na przekazaniu ich w nagłówku „Authorization”, którego zawartość powinna zawierać zakodowaną za pomocą Base64 parę id:secret poprzedzoną

słowem „Basic”, drugi natomiast polega na przekazaniu ich w ciele zapytania w wartościach parametrów „client_id” oraz „client_secret”¹.

Jeżeli użytkownik istnieje i został uwierzytelniony, otrzymuje odpowiedź² w formacie JSON z polami:

- access_token – AT, służący do autoryzacji zapytań użytkownika w innych serwisach;
- token_type – pole określające typ AT; zawsze jest ustawione na „bearer”, czyli na okaziciela;
- expires_in – pole określające po jakim czasie AT traci ważność i nie można się już nim posługiwać (czas ten podawany jest w sekundach i wynosi 2 minuty);
- refresh_token – pole zawierające Refresh Token (dalej zwany RT).

Po otrzymaniu takiej odpowiedzi użytkownik API może wykorzystywać AT w celu autoryzacji w systemie. Powinien dodawać nagłówek o nazwie „Authorization”, w którym należy umieścić wygenerowany AT poprzedzony słowem „Bearer”. Tylko zapytania z takim nagłówkiem będą poprawnie obsłużone przez wszystkie serwisy.

W momencie wygaśnięcia AT³ użytkownik traci możliwość autoryzacji w systemie i musi wygenerować go ponownie.

7. PRZYKŁADY WYWOŁAŃ I ODPOWIEDZI

Przykładowe wywołania metod API zostały przygotowane w formie interaktywnej – jako zestaw dwóch plików do importu w popularnym narzędziu Postman. W skład zestawu wchodzi:

- Plik z konfiguracją środowiska;
- Plik z definicją kolekcji zapytań – wraz z przykładowymi dokumentami (treści dokumentów należy dostosować do swoich potrzeb biznesowych, dostarczone dane są demonstracyjne). Adresy PEPPOL w treściach dokumentów należy dostosować do takich, do których dany Użytkownik API ma dostęp.

1 Szkielet zapytania oparty na podstawie dokumentacji OAuth2: <https://www.oauth.com/oauth2-servers/access-tokens/client-credentials/>

2 Odpowiedź generowana na podstawie dokumentacji OAuth2: <https://www.oauth.com/oauth2-servers/access-tokens/access-token-response/>

3 Użyty schemat to AT o krótkim czasie życia– zgodnie z dokumentacją OAuth2: <https://www.oauth.com/oauth2-servers/access-tokens/access-token-lifetime/#short-token-long-refresh>