



INTERFEJS WEBSERVICES

**NADAWANIE PAKIETÓW
DO SYSTEMU MKP PRZEZ INTERNET**

DOKUMENTACJA TECHNICZNA

PAŹDZIERNIK 2016

Spis treści

1. Wstęp.....	2
2. Informacje ogólne.....	2
3. Definicje.....	3
3.1. Wyliczenia.....	3
3.1.1. RodzajUsługi.....	3
3.1.2. StanPakietu.....	3
3.1.3. RodzajDokumentu.....	3
3.1.4. StanOplacenia.....	4
3.1.5. OpisStanuPakietu.....	4
3.1.6. KategoriaBledu.....	4
3.1.7. KodBledu.....	5
3.2. Struktury.....	5
3.2.1. OpisPakietu.....	5
3.2.2. OpisDokumentu.....	6
3.2.3. OpisStanuOplacenia.....	7
3.2.4. ListaZnalezionychPaketow.....	7
3.2.5. OpisBledu.....	7
4. Metody.....	8
4.1. Test współpracy z webservice.....	8
4.2. Utworzenie pakietu plikowego.....	9
4.3. Pobranie informacji o pakiecie.....	10
4.4. Usunięcie wskazanego pakietu.....	11
4.5. Zatwierdzenie zarejestrowanego pakietu.....	12
4.6. Wyszukiwanie pakietów.....	13
4.7. Pobranie listy dokumentów.....	14
4.8. Pobranie dokumentu.....	15
4.9. Pobranie dokumentu w określonym formacie.....	16
4.10. Zlecenie sprawdzenia stanu opłacenia.....	17
4.11. Sprawdzenie stanu opłacenia pakietu.....	17

1. Wstęp

Niniejszy dokument specyfikuje interfejs API usług sieciowych (WebServices) umożliwiający nadawanie pakietów do systemu MKP za pośrednictwem publicznego Internetu.

Interfejs usług sieciowych umożliwia:

- zarejestrowanie pakietu plikowego w systemie MKP,
- wyszukiwanie zarejestrowanych przez siebie pakietów,
- zatwierdzenie wybranego pakietu,
- usunięcie wybranego pakietu,
- pobieranie dokumentów związanych z pakietem wygenerowanych przez system MKP,
- sprawdzenie stanu opłacenia pakietu.

2. Informacje ogólne

Usługi dostępne są pod adresem:

<https://www.pzw.poczta-polska.pl/mkpwww/Uslugi/ObslugaPakietow.svc>

Do korzystania z usług wymagana jest rejestracja w serwisie:

<https://www.pzw.poczta-polska.pl/>

Szczegółowa specyfikacja WSDL dostępna jest pod adresem:

<https://www.pzw.poczta-polska.pl/mkpwww/Uslugi/ObslugaPakietow.wsdl>

Do dostępu do usługi wykorzystywane jest połączenie HTTPS z wiązaniem WCF „basicHttpBinding” z trybem zabezpieczeń „TransportWithMessageCredential” i trybem przekazania danych uwierzytelniających „UserName”. Nazwa użytkownika i hasło (zgodne z rejestracją w serwisie) przekazywane są w nagłówkach komunikatu SOAP w tunelu SSL/TLS.

W przypadku wykrycia nadmiernego obciążania serwera wywołaniami niniejszego API zamiast wyspecyfikowanych komunikatów mogą być zwracane niskopoziomowe błędy na poziomie protokołu HTTP lub SOAP (np. błąd uwierzytelnienia dla obciążającego użytkownika pomimo przekazania poprawnego hasła).

3. Definicje

Poniższe API wykorzystuje przestrzeń nazw komunikatów:

"<http://poczta-polska.pl/MKP/ObslugaPakietow/v1.0>"

3.1. Wyliczenia

3.1.1. RodzajUsługi

```
[DataContract]
public enum RodzajUslugi
{
    [EnumMember]
    PZW, //Ekspres pieniężny (EP-PZW)
    [EnumMember]
    PPE, //Przekaz pocztowy
}
```

3.1.2. StanPakietu

```
[DataContract]
public enum StanPakietu
{
    [EnumMember]
    NIEOBSLUGIWANY,
    [EnumMember]
    Otwarty,
    [EnumMember]
    UtworzonyCzekaNaWeryfikacjeMKP,
    [EnumMember]
    OdrzuconyZPowoduBleduWeryfikacji,
    [EnumMember]
    ZweryfikowanyOczekujacyNaZatwierdzenie,
    [EnumMember]
    Zamkniety,
}
```

3.1.3. RodzajDokumentu

```
[DataContract]
public enum RodzajDokumentu
{
    [EnumMember]
    NIEOBSLUGIWANY,
    [EnumMember]
    RaportKontroliFormalnej,
    [EnumMember]
    WykazBledow,
    [EnumMember]
    PoczтоваяKsiazkaNadawcza,
    [EnumMember]
    RaportStanuRealizacji,
    [EnumMember]
    SpecyfikacjaZwrotowZadanieZwrotu,
    [EnumMember]
    SpecyfikacjaZwrotowZwrotyPrzekazow,
```

```

    [EnumMember]
    SpecyfikacjaZwrotowPrzeterminowanie,
    [EnumMember]
    SpecyfikacjaZwrotowNadwyzka,
}

```

3.1.4. StanOplacenia

```

[DataContract]
public enum StanOplacenia
{
    [EnumMember]
    NIEOBSLUGIWANY,

    [EnumMember]
    ZlecenieOczekujeNaSprawdzenie,
    [EnumMember]
    BrakPakietu,
    [EnumMember]
    BrakWplaty,
    [EnumMember]
    WplataZgodna,
    [EnumMember]
    WplataNiezgodna,
}

```

3.1.5. OpisStanuPakietu

```

[DataContract]
public enum OpisStanuPakietu
{
    [EnumMember]
    NIEOBSLUGIWANY,

    [EnumMember]
    Zarejestrowany,
    [EnumMember]
    Oplacony,
    [EnumMember]
    Uniewazniony,
    [EnumMember]
    Przeterminowany,
}

```

3.1.6. KategoriaBledu

```

[DataContract]
public enum KategoriaBledu
{
    [EnumMember]
    NIEOBSLUGIWANY,

    [EnumMember]
    BladWywolania,
    [EnumMember]
    BladNaSerwerze,
}

```

3.1.7. KodBledu

```
[DataContract]
public enum KodBledu
{
    [EnumMember]
    NIEOBSLUGIWANY,

    [EnumMember]
    BladStrukturyKomunikatu,
    [EnumMember]
    BladPrzekroczeniaOgraniczenCzestotliwosciWywolan,

    [EnumMember]
    BledneDaneDoUtworzeniaPakietu,

    [EnumMember]
    BlednyIdentyfikatorPakietu,
    [EnumMember]
    BlednyIdentyfikatorDokumentu,

    [EnumMember]
    StanPakietuNieodpowiedniDoZadanejOperacji,

    [EnumMember]
    BrakPakietuDlaIdentyfikatora,
    [EnumMember]
    BrakDokumentuDlaIdentyfikatora,
    [EnumMember]
    BrakZadaniaSprawdzeniaStanuDlaIdentyfikatora,

    [EnumMember]
    KomunikatBiznesowy,
}
```

3.2. Struktury

3.2.1. OpisPakietu

Opisuje zarejestrowany pakiet. Część atrybutów dostępna jest tylko dla pakietów zamkniętych zgodnie z opisem poniżej.

```
[DataContract]
public class OpisPakietu
{
    [DataMember(IsRequired = true)]
    public string UIP { get; set; }

    [DataMember(IsRequired = true)]
    public RodzajUslugi Usługa { get; set; }

    [DataMember(IsRequired = true)]
    public string NazwaPliku { get; set; }

    [DataMember(IsRequired = true)]
    public StanPakietu Stan { get; set; }
}
```

```

//INFO: data bez czasu
[DataMember(IsRequired = true)]
public DateTime DataUtworzenia { get; set; }

[DataMember(IsRequired = true)]
public string NumerUmowy { get; set; }

[DataMember(IsRequired = true)]
public decimal ProcentOpustu { get; set; }

//INFO: wszystko ponizej wypełniane tylko dla stanu 'Zamkniety'

[DataMember(IsRequired = false)]
public int LiczbaPrzekazowPoprawnych { get; set; }

[DataMember(IsRequired = false)]
public decimal WartoscPrzekazowPoprawnych { get; set; }

[DataMember(IsRequired = false)]
public decimal OplataZaNadanie { get; set; }

[DataMember(IsRequired = false)]
public string NumerRachunkuBankowegoWplaty { get; set; }

[DataMember(IsRequired = false)]
public DateTime DataGranicznaWplaty { get; set; }
}

```

3.2.2. OpisDokumentu

Opisuje dokument dotyczący zarejestrowanego pakietu.

```

[DataContract]
public class OpisDokumentu
{
    [DataMember(IsRequired = true)]
    public string Identyfikator { get; set; }

    [DataMember(IsRequired = true)]
    public RodzajDokumentu Rodzaj { get; set; }

    [DataMember(IsRequired = true)]
    public DateTime DataUtworzenia { get; set; }

    [DataMember(IsRequired = true)]
    public string NazwaPliku { get; set; }
}

```

3.2.3. OpisStanuOplacenia

Opisuje stan opłacenia pakietu. Część atrybutów dostępna jest tylko dla pakietów sparowanych zgodnie z opisem poniżej.

```
[DataContract]
public class OpisStanuOplacenia
{
    [DataMember(IsRequired = true)]
    public StanOplacenia Stan { get; set; }

    [DataMember(IsRequired = false)]
    public OpisStanuPakietu OpisStanPakietu { get; set; }

    //INFO: wszystko poniżej wypełniane tylko dla stanu różnego od
    'BrakWplaty'
    [DataMember(IsRequired = false)]
    public DateTime DataSparowania { get; set; }
}
```

3.2.4. ListaZnalezionychPakietow

Zawiera listę znalezionych pakietów wraz z ewentualną informacją o przekroczeniu limitu zwracanych danych.

```
[DataContract]
public class ListaZnalezionychPakietow
{
    [DataMember(IsRequired = true)]
    public OpisPakietu[] ListaPakietow { get; set; }

    [DataMember(IsRequired = false)]
    public bool ListaPakietowOgraniczonaZPowoduPrzekroczeniaLimitu {
get; set; }
}
```

3.2.5. OpisBledu

Opisuje rodzaj zwracanego błędu. Poza komunikatem tekstowym zwracana jest ogólna kategoria błędu (błąd wywołania ze strony klienta, błąd po stronie serwera) oraz kod błędu z predefiniowanej listy.

```
[DataContract]
public class OpisBledu
{
    [DataMember(IsRequired = true)]
    public KategoriaBledu Kategoria { get; set; }

    [DataMember(IsRequired = true)]
    public KodBledu Kod { get; set; }

    [DataMember]
    public string Komunikat { get; set; }
}
```


4. Metody

4.1. Test współpracy z webservice

Metoda służy do sprawdzenia prawidłowego podłączenia do Webservice i możliwościwołania webmetody. Nie ma żadnego znaczenia merytorycznego.

Sygnatura metody:

```
[OperationContract]
[FaultContract(typeof(OpisBledu))]
// Webmetoda Test umożliwia sprawdzenie prawidłowego
// podłączenia (autoryzacji) z Webservice.
[return: MessageParameter(Name = "Odpowiedz")]
string Test(
    [MessageParameter(Name = "Pytanie")]
    string tekst);
```

Parametry wejściowe:

Parametr	Opis
Pytanie	Parametr obowiązkowy, dowolna wartość tekstowa.

Dane wyjściowe:

Nazwa	Opis
Odpowiedz	Webmetoda zwracana łańcuch znaków o wartości: „Odpowiedź: ” połączony z wartością parametru „Pytanie”.

4.2. Utworzenie pakietu plikowego

Metoda służy do utworzenia pakietu poprzez wczytanie pliku z przekazami / zleceniami.

Parametr wyjściowy – UIP - Identyfikator pakietu ma znaczenie merytoryczno-techniczne.

Jest nadawany w sytuacji, gdy wykonanie metody **UtworzPakiet** zakończyło się pomyślnie, tzn. pakiet został technicznie nadany. Nie oznacza to jednak, że pakiet jest merytorycznie poprawny.

Kontrola poprawności merytorycznej pakietu, a w szczególności zawartości pliku zostanie wykonana przez system informatyczny Poczty Polskiej w kolejnym etapie przetwarzania tego nadanego pakietu.

To, czy pakiet jest merytorycznie poprawny można zweryfikować za pomocą metody **DajPakiet**.

Maksymalna liczba przekazów jednorazowo podanych w pliku to 10 000 przekazów.

Dopuszczalnymi formatami zawartości pliku są XML i CSV.

Sygnatura metody:

```
[OperationContract]
[FaultContract(typeof(OpisBledu))]
[return: MessageParameter(Name = "UIP")]
string UtworzPakiet(
[MessageParameter(Name = "RodzajUslugi")]
RodzajUslugi rodzajUslugi,
[MessageParameter(Name = "NazwaPliku")]
string nazwaPliku,
[MessageParameter(Name = "ZawartoscPliku")]
byte[] bajtyPliku,
[MessageParameter(Name = "ZatwierdzenieAutomatyczne")]
bool zatwierdzenieAutomatyczne) ;
```

Parametry wejściowe:

Parametr	Opis
RodzajUslugi	Rodzaj usługi, dla której tworzony jest pakiet.
NazwaPliku	Nazwa pliku z przekazami / zleceniami.
BajtyPliku	Zawartość pliku.
ZatwierdzenieAutomatyczne	Żądanie automatycznego zatwierdzenia pakietu po kontroli formalnej pliku (pakiet zostanie automatycznie zatwierdzony tylko w przypadku poprawnego przejścia kontroli formalnej pliku).

Dane wyjściowe:

Nazwa	Opis
UIP	Identyfikator utworzonego pakietu.

4.3. Pobranie informacji o pakiecie

Metoda służy do pobrania informacji o zarejestrowanym pakiecie.

W sytuacji, gdy system informatyczny Poczty Polskiej nie przeprowadził jeszcze kontroli poprawności merytorycznej pakietu pole Stan ma wartość `UtworzonyCzekaNaWeryfikacjeMKP`, pozostałe wartości pól struktury `OpisPakietu` mogą mieć wartości nieokreślone.

System korzystający z Webservice powinien ponawiać wywołanie tej metody (**DajPakiet**) tak długo, aż pole Stan będzie miało wartość inną niż `UtworzonyCzekaNaWeryfikacjeMKP`, przy czym odstęp czasu pomiędzy kolejnymi wywołaniami metody powinien wynosić co najmniej kilkadziesiąt sekund.

Sygnatura metody:

```
[OperationContract]
[FaultContract(typeof(OpisBledu))]
[return: MessageParameter(Name = "Pakiet")]
OpisPakietu DajPakiet(
    [MessageParameter(Name = "UIP")]
    string UIP);
```

Parametry wejściowe:

Nazwa	Opis
UIP	Identyfikator utworzonego pakietu.

Dane wyjściowe:

Nazwa	Opis
OpisPakietu	Struktura zawierająca opis zarejestrowanego pakietu.

W przypadku podanie błędnego identyfikatora pakietu zwracany jest kod błędu:

BlednyIdentyfikatorPakietu.

W przypadku braku pakietu zwracany jest kod błędu:

BrakPakietuDlaIdentyfikatora.

4.4. Usunięcie wskazanego pakietu

Metoda służy do usuwania zarejestrowanego pakietu. Można usunąć tylko pakiet o stanie innym niż **Zamknięty**.

Sygnatura metody:

```
[OperationContract]
[FaultContract(typeof(OpisBledu))]
void UsunPakiet(
    [MessageParameter(Name = "UIP")]
    string UIP);
```

Parametry wejściowe:

Nazwa	Opis
UIP	Identyfikator utworzonego pakietu.

Dane wyjściowe:

(brak)

W przypadku podania błędnego identyfikatora pakietu zwracany jest kod błędu:

BlednyIdentyfikatorPakietu.

W przypadku gdy nie można usunąć pakietu ze względu na błędny stan, zwracany jest kod błędu:

StanPakietuNieodpowiedniDoZadanejOperacji.

W przypadku braku pakietu zwracany jest kod błędu:

BrakPakietuDlaIdentyfikatora.

4.5. Zatwierdzenie zarejestrowanego pakietu

Metoda służy do pobrania informacji o zarejestrowanym pakiecie.

Sygnatura metody:

```
[OperationContract]
[FaultContract(typeof(OpisBledu))]
void ZatwierdzPakiet(
    [MessageParameter(Name = "UIP")]
    string UIP);
```

Parametry wejściowe:

Nazwa	Opis
UIP	Identyfikator utworzonego pakietu

Dane wyjściowe:

(brak)

W przypadku podanie błędnego identyfikatora pakietu zwracany jest kod błędu:

BlednyIdentyfikatorPakietu.

W przypadku braku pakietu zwracany jest kod błędu:

BrakPakietuDlaIdentyfikatora.

W przypadku próby zamknięcia pakietu o niedozwolonym stanie zwracany jest kod błędu:

StanPakietuNieodpowiedniDoZadanejOperacji.

4.6. Wyszukiwanie pakietów

Metoda służy do wyszukiwania zarejestrowanych pakietów.

Sygnatura metody:

```
[OperationContract]
[FaultContract(typeof(OpisBledu))]
[return: MessageParameter(Name = "ListaZnalezionychPakietow")]
ListaZnalezionychPakietow ZnajdzPakiety(
    [MessageParameter(Name = "ListaStanow")]
    [Optional] StanPakietu[] listaStanow,
    [MessageParameter(Name = "DataOd")]
    [Optional] DateTime? dataOd,
    [MessageParameter(Name = "DataDo")]
    [Optional] DateTime? dataDo,
    [MessageParameter(Name = "Usługa")]
    [Optional] RodzajUslugi? usługa
);
```

Parametry wejściowe:

Parametr	Opis
ListaStanow	Opcjonalny filtr na StanPakietu .
DataOd, DataDo	Opcjonalny zakres dat, dla którego prowadzone jest wyszukiwanie.
Usługa	Opcjonalny filtr na RodzajUslugi .

Dane wyjściowe:

Nazwa	Opis
ListaZnalezionychPakietow	Struktura zawierająca listę znalezionych pakietów.

Lista zwracanych pakietów ograniczana jest na serwerze do pewnej konfigurowalnej liczby (rzędu kilkuset). W przypadku gdy lista znalezionych pakietów przekroczy ustawiony limit w zwracanym komunikacie ustawiany jest atrybut

ListaPakietowOgraniczonaZPowoduPrzekroczeniaLimitu, aby zasygnalizować, że nie zostały zwrócone kompletne wyniki i należy zawęzić kryteria.

4.7. Pobranie listy dokumentów

Metoda służy do pobrania listy dokumentów dla wskazanego pakietu.

Sygnatura metody:

```
[OperationContract]
[FaultContract(typeof(OpisBledu))]
[return: MessageParameter(Name = "ListaDokumentow")]
OpisDokumentu[] DajDokumenty(
    [MessageParameter(Name = "UIP")]
    string UIP,
    [MessageParameter(Name = "RodzajeDokumentow")]
    RodzajDokumentu[] rodzaje
);
```

Parametry wejściowe:

Parametr	Opis
UIP	Identyfikator pakietu.
Rodzaje	Opcjonalny filtr na RodzajDokumentu .

Dane wyjściowe:

Nazwa	Opis
ListaDokumentow	Struktura zawierająca listę znalezionych dokumentów.

W przypadku podanie błędnego identyfikatora pakietu zwracany jest kod błędu:

BlednyIdentyfikatorPakietu.

W przypadku braku pakietu zwracany jest kod błędu:

BrakPakietuDlaIdentyfikatora.

4.8. Pobranie dokumentu

Metoda służy do pobrania treści wskazanego dokumentu.

Sygnatura metody:

```
[OperationContract]
[FaultContract(typeof(OpisBledu))]
[return: MessageParameter(Name = "ZawartoscDokumentu")]
byte[] DajTrescDokumentu(
    [MessageParameter(Name = "IdentyfikatorDokumentu")]
    string id);
```

Parametry wejściowe:

Parametr	Opis
IdentyfikatorDokumentu	Identyfikator dokumentu.

Dane wyjściowe:

Nazwa	Opis
ZawartoscDokumentu	Zawartość pobieranego dokumentu.

4.9. Pobranie dokumentu w określonym formacie

Metoda służy do pobrania treści wskazanego dokumentu w formacie txt, csv lub xml.

Sygnatura metody:

```
[OperationContract]
[FaultContract(typeof(OpisBledu))]
[return: MessageParameter(Name = "ZawartoscDokumentu")]
byte[] DajTrescDokumentuWFormacie(
    [MessageParameter(Name = "IdentyfikatorDokumentu")]
    string id,
    [MessageParameter(Name = "FormatDokumentu")]
    string format);
```

Parametry wejściowe:

Parametr	Opis
IdentyfikatorDokumentu	Identyfikator dokumentu.
FormatDokumentu	Format, w którym ma być pobrana treść dokumentu. Możliwe wartości „txt”, „xml”, „csv”

Dane wyjściowe:

Nazwa	Opis
ZawartoscDokumentu	Zawartość pobieranego dokumentu w formacie określonym w parametrze wejściowym FormatDokumentu.

4.10. Zlecenie sprawdzenia stanu opłacenia

Metoda służy do zlecenia sprawdzenia stanu opłacenia pakietu.

Sygnatura metody:

```
[OperationContract]
[FaultContract(typeof(OpisBledu))]
void ZlecSprawdzenieStanuOplacenia(
    [MessageParameter(Name = "UIP")]
    string UIP);
```

Parametry wejściowe:

Parametr	Opis
UIP	Identyfikator pakietu.

Dane wyjściowe:

(brak)

W przypadku podanie błędnego identyfikatora pakietu zwracany jest kod błędu:

BlednyIdentyfikatorPakietu.

4.11. Sprawdzenie stanu opłacenia pakietu

Metoda służy o pobrania informacji o stanie pakietu dla złożonego wcześniej zlecenia.

Sygnatura metody:

```
[OperationContract]
[FaultContract(typeof(OpisBledu))]
[return: MessageParameter(Name = "StanOplaceniaPakietu")]
OpisStanuOplacenia DajStanOplacenia(
    [MessageParameter(Name = "UIP")]
    string UIP);
```

Parametry wejściowe:

Parametr	Opis
UIP	Identyfikator pakietu.

Dane wyjściowe:

Parametr	Opis
StanOplaceniaPakietu	Opis stanu opłacenia pakietu.

W przypadku podanie błędnego identyfikatora pakietu zwracany jest kod błędu:
BlednyIdentyfikatorPakietu.

W przypadku gdy nie znaleziono zlecenie zwracany jest kod błędu:
BrakZadaniaSprawdzeniaStanuDlaIdentyfikatora.