



Pošta Slovenije d.o.o.
Slomškov trg 10
2500 Maribor
e-pošta: info@posta.si
www.posta.si

eSpremnica.Provider.Webapi

Struktura podatkovnega modela za elektronsko izmenjavo podatkov

Maribor, November 2019

Kazalo

Kazalo.....	2
Opis dokumenta	4
eSpremnica.Provider.WebApi	5
Vstopna točka.....	5
Objekti za klic in odgovor	6
ProviderRequest	6
Authentication.....	6
Data.....	7
EnumOddajaRequestDataType.....	7
ProviderResponse	8
EnumResposeCode.....	8
Value object	9
ProcessStatus.....	9
EnumTransactionCode	9
Error.....	10
eOddaja pošiljk.....	11
R_GetGuid – Pridobivanje žetona za posredovanje podatkov.....	11
C_PostData – Oddaja podatkov o pošiljkah.....	11
R_GetStatus – Poizvedba o statusu obdelave oddanih pošiljk.....	11
eOddaja Entitete.....	12
eOddajaData	12
eOddajaParcel.....	13
EnumParcelContent.....	13
eOddajaService.....	14
eOddajaAddressee.....	14
eOddajaCN23.....	15
eOddajaInternational	16
EnumNonDeliveryInstruction.....	17

EnumInternationalCategoryType.....	17
eOddajaUPN.....	18
EnumUPNType.....	18
CN23 servis	19
Pomoč	19
C_Document – Oddaja podatkov o CN23 pošiljki.....	19
R_Report – Pridobivanje PDF-dokumenta – obrazca o CN23 pošiljki.....	19
CN23-entitete.....	20
CN23Document.....	20
Consignee.....	21
Shipment.....	22
Category.....	23
AmountTotal	23
AmountCOD.....	24
Item.....	25
Article.....	26
Amount.....	26
Implementacija C#	27
Pomožne metode.....	27
Decompress.....	27
Post.....	27
Post_Document().....	28
Get_Report().....	32
Implementacija REST client.....	33
Post_Document.....	33
Get_Report().....	34
Kazalo tabel	35
Kazalo metod	36
Kazalo implementacij	36

Opis dokumenta

Dokument opisuje strukturo podatkovnega modela za elektronsko izmenjavo podatkov z informacijskim sistemom Pošte Slovenije (v nadaljevanju PS). Omogoča izmenjavo podatkov o pošiljkah za domači in mednarodni promet ter pridobivanje ustreznih obrazcev CN23 v PDF-obliki.

Datum	Avtor	Opomba
11. 3. 2019	Andrej Majcen	Priprava dokumentacije

Tabela 1 – Spremembe dokumenta

eSpremnica.Provider.WebApi

WebApi je namenjen za:

- oddajo podatkov o pošilkah,
- izmenjavo podatkov o domačih in mednarodnih pošilkah (CN23).

Za pravilno implementacijo je treba zagotoviti naslednje:

- komunikacija deluje samo preko HTTPS;
- izmenjava podatkov je omogočena samo registriranim uporabnikom – komitentom;
- za vzpostavitev komunikacije s servisom je treba zagotoviti avtorizacijo med sistemoma (BasicAuth). Prijavne podatke posreduje skrbnik pogodbe na strani Pošte Slovenije;
- odgovor (Response) je komprimiran string. Uporabljena je GZip-oblika s 4 zlogi odmika.

Vstopna točka

Test: <https://tstePortal.posta.si/Services/eSpremnica.Provider/WebApi/api/>

Produkcija: <https://ePortal.posta.si/Services/eSpremnica.Provider/WebApi/api/>

Objekti za klic in odgovor

ProviderRequest

Objekt je namenjen za posredovanje informacij na WebApi servis.

Element	Tip	Opis	Obvezen
Authentication	Authentication	Avtentikacija uporabnika	O
Guid	String	Žeton za oddajo podatkov	O
Data	Object	Tip odvisen od klica webAPI-ja	O
Route	String		N
DataType	EnumOddajaRequestDataType	Tip oddaje podatkov	PO

Tabela 2 - ProviderRequest

C# implementacija

```
public class ProviderRequest
{
    public Authentication Authentication {get;set; }
    public string Guid {get;set; }
    public object Data { get;set; }
    public string Route { get; set; }
    public EnumOddajaRequestDataType DataType {get;set; }
}
```

Implementacija 1 - ProviderRequest

Authentication

Objekt je namenjen avtentikaciji uporabnika, ki želi izmenjavo podatkov s sistemom PS.

Element	Tip	Opis	Obvezen
KomitentId	String	(*)	O
Pogodbald	String	(*)	O
Podruznicald	String	(*)	O
Postald	string	(*)	O

Tabela 3 – Authentication

(*) Podatke za prijavo uporabnika posreduje PS.

C# implementacija

```
public class Authentication
{
    public string PostaId { get; set; }
    public int PodruznicaId { get; set; }
    public int PogodbaId { get; set; }
    public int KomitentId { get; set; }
}
```

Implementacija 2 - Authentication

Data

Objekt Data je namenjen posredovanju podatkov v sistem PS. Njegov tip je odvisen od klica spletnega servisa.

Servis	Tip objekta	Obvezen
eOddaja	eOddajaData	<input type="radio"/>
CN23	CN23Document	<input type="radio"/>

Tabela 4 – Data

EnumOddajaRequestDataType

Tip oddanih podatkov, uporablja se samo pri spletnem servisu eOddaja.

Element	Tip	Opis
CSV	int;2	CSV-oblika
XML	int;3	XML-oblika
JSON	int;4	JSON-oblika

Tabela 5 – EnumeOddajaRequestDataType

ProviderResponse

Objekt se vrača po klicu WebApi servisa s potrebno vsebino. V primeru uspešne poizvedbe (Code=200) se podatki nahajajo v atributu Value. V nasprotnem primeru ta atribut vsebuje vsebino o napaki.

Element	Tip	Opis	Obvezen
Code	EnumResponseCode	Status o poizvedbi	O
Value	object	Vsebina odgovora	O
Route	string	Klicana pot	P

Tabela 6 – ProviderResponse

C# implementacija

```
public class ProviderResponse
{
    public EnumResponseCode Code { get; set; }
    public object Value { get; set; }
    public string Route { get; set; }
}
```

Implementacija 3 – ProviderResponse

EnumResponseCode

Vrednost	Opis
200	OK
400	BadRequest
401	Unauthorized
403	Forbidden
408	TimeOut
412	InvalidArguments
500	InternalServerError
501	NotImplemented
550	ValidationError
560	NoData
600	TryAgain

Tabela 7 – EnumResponseCode

Value object

Value je odvisen od klica spletnega servisa. V primeru Uspešne poizvedbe (200) vsebuje objekt, drugače opis napake, ki je tipa string.

Servis	Klic	Tip objekta	Opis
eOddaja	R_GetGuid	string	Vsebuje vrednost GUID
eOddaja	C_PostData	string	Vsebuje EnumResultCode
eOddaja	R_GetStatus	ProcessStatus	Vsebuje objekt ProcessStatus
CN23	C_Document	CN23DocumentValue	Vsebuje objekt CN23DocumentValue
CN23	R_Report	string	Vsebuje .pdf generiran CN23 obrazec v base64 zapisu

Tabela 8 – Value

ProcessStatus

Objekt je namenjen za posredovanje informacij o statusu obdelave posredovanih pošiljk.

Element	Tip	Opis	Obvezen
ResultCode	EnumTransactionCode	Tip	
Errors	List<Error>	Seznam morebitnih napak	
ProcessNumber	int	Številka oddajnega popisa	

Tabela 9 – ProcessStatus

EnumTransactionCode

Seznam statusov obdelave podatkov

Element	Tip	Opis	Obvezen
InTransmission	int;0	Podatki se še posredujejo	
Transmitted	int;1	Podatki so posredovani	
InProgress	int;2	Oddani podatki so v stanju obdelave	
ProcessingWithoutErrors	int;3	Podatki so obdelani brez napak	
ProcessingWithErrors	int;4	Podatki so obdelani z napako	
InvalidInputData	int;5	Posredovani podatki so napačni	
NoData	int;6	Za zahtevan žeton ni podatkov	

Tabela 10 - EnumTransactionCode

Error

Objekt je namenjen za posredovanje informacij o napakah, če so se pojavile pri obdelavi posredovanih pošiljk, samo kadar je ResultCode = 4.

Element	Tip	Opis	Obvezen
RowNumber	int	Številka napake	
Description	string	Opis morebitnih napak v slovenskem jeziku	
DescriptionEN	string	Opis morebitnih napak v angleškem jeziku	

Tabela 11 - Error

C# implementacija

```
public class Error
{
    public int RowNumber{ get; set; }
    public string Description{ get; set; }
    public string DescriptionEN{ get; set; }
}
```

Implementacija 4 - Error

CN23DocumentValue

Objekt vrača identifikacijsko številko dokumenta in sprejemno številko pošiljke.

Element	Tip	Opis	Obvezen
ShortNumber	int	Tip	
ParcelNumber	string	Številka črtne kode	

Tabela 12 – CN23DocumentValue

C# implementacija

```
public class CN23DocumentValue
{
    public int ShortNumber { get; set; }
    public string ParcelNumber { get; set; }
}
```

Implementacija 5 - CN23DocumentValue

eOddaja pošiljk

Za uspešno oddajo podatkov je treba naprej pridobiti unikatno GUID-število.

R_GetGuid – Pridobivanje žetona za posredovanje podatkov

Namenjena za pridobivanje unikatnega žetona – (GUID-a).

URL: **/eOddaja/R_GetGuid**

Metoda : **POST**

Parametri: **ProviderRequest**

Odgovor: **ProviderResponse**

V primeru uspešne poizvedbe `ProviderResponse.Code=200` se v `ProviderResponse.Value` nahaja unikatni žeton.

C_PostData – Oddaja podatkov o pošiljkah

Namenjena za oddajo podatkov o pošiljkah.

URL: **/eOddaja/C_PostData**

Metoda : **POST**

Parametri: **ProviderRequest**

Odgovor: **ProviderResponse**

V primeru uspešne poizvedbe `ProviderResponse.Code=200`.

R_GetStatus – Poizvedba o statusu obdelave oddanih pošiljk

Namenjena za pridobivanja statusa obdelave oddanih podatkov o pošiljkah. Metodo pokličite z zakasnitvijo 15~30 sekund šele po uspešnem klicu metode `C_PostData`. Proces lahko ponavljate v ne prepogostih intervalih dokler je `EnumTransactionCode` enak 1.

URL: **/eOddaja/R_GetStatus**

Metoda : **POST**

Parametri: **ProviderRequest**

Odgovor: **ProviderResponse**

V primeru uspešne poizvedbe `ProviderResponse.Code=200` se v `ProviderResponse.Value` nahaja objekt `ProcessStatus`.

eOddaja Entitete

eOddajaData

Objekt je namenjen za posredovanje informacij o pošiljki.

Element	Tip	Opis	Obvezen
RecordNumber	int	Številka zapisa	PO
ParcelType	int	Vrsta pošiljke (*)	O
Parcels	List<eOddajaParcel>	Podatki pošiljke	O
CommonServices	List<eOddajaService>	Skupne storitve pošiljke	N
Addressee	eOddajaAddress	Naslovník	O
UPN	eOddajaUPN	Podatki za UPN	N
Comment1	string	Opomba 1	N
Comment2	string	Opomba 2	N

Tabela 13 – eOddajaData

(*) Podatki, ki jih posreduje PS

C# implementacija

```
public class eOddajaData
{
    public int RecordNumber { get; set; }
    public int ParcelType { get; set; }
    public List<eOddajaParcel> Parcels { get; set; }
    public List<eOddajaService> CommonServices { get; set; }
    public eOddajaAddress Addressee { get; set; }
    public eOddajaUPN Upn { get; set; }
    public string Comment1 { get; set; }
    public string Comment2 { get; set; }
}
```

Implementacija 6 - eOddajaData

eOddajaParcel

Objekt je namenjen za posredovanje osnovnih podatkov o pošiljki.

Element	Tip	Opis	Obvezen
ParcelNumber	string	Sprejemna številka	O
Weight	decimal	Teža pošiljke	PO
Volume	decimal	Volumen	PO
AdditionalServices	eOddajaService	Dodatne/posamične storitve	N
CN23	eOddajaCN23	Podatki za potrebe CN23	N
International	eOddajaInternational	Podatki za mednarodni promet	N
ParcelContent	EnumParcelContent	Tip vsebine	PO

Tabela 14 – eOddajaParcel

C# implementacija

```
public class eOddajaParcel
{
    public string ParcelNumber { get; set; }
    public decimal Weight { get; set; }
    public decimal Volume { get; set; }
    public List<eOddajaService> AdditionalServices { get; set; }
    public List<eOddajaCN23> Cn23 { get; set; }
    public eOddajaInternational International { get; set; }
    public EnumParcelContent ParcelContent { get; set; }
}
```

Implementacija 7 – eOddajaParcel

EnumParcelContent

Element	Tip	Opis	Obvezen
DOCUMENTS	Int;0	Vsebuje dokumente	
GOODS	Int;1	Pošilja se blago	

Tabela 15 – EnumParcelContent

eOddajaService

Objekt je namenjen za posredovanje podatkov o skupnih in dodatnih storitvah pošiljke.

Element	Tip	Opis	Obvezen
Code	string	Šifra storitve	O
Amount	decimal	Vrednost	PO

Tabela 16 – eOddajaService

C# implementacija

```
public class eOddajaService
{
    public string Code { get; set; }
    public decimal Amount { get; set; }
}
```

Implementacija 8 – eOddajaService

eOddajaAddressee

Objekt je namenjen za posredovanje podatkov o naslovniku pošiljke.

Element	Tip	Opis	Obvezen
AddresseeName1	string	Naziv	O
AddresseeName2	string	Dodaten naziv	N
AddresseeAddress	string	Naslov	N
AddresseePostId	string	Poštna številka	O
AddresseePost	string	Naziv pošte	O
AddresseeCountryId	string	Koda države	O
AddresseePhone	string	Telefonska številka	PO
AddresseeEMail	string	Elektronski naslov	N

Tabela 17 – eOddajaAddressee

C# implementacija

```
public class eOddajaAddressee
{
    public string AddresseeName1 { get; set; }
    public string AddresseeName2 { get; set; }
    public string AddresseeAddress { get; set; }
    public string AddresseePostId { get; set; }
    public string AddresseePost { get; set; }
    public string AddresseeCountryId { get; set; }
    public string AddresseePhone { get; set; }
    public string AddresseeEmail { get; set; }
}
```

Implementacija 9 – eOddajaAddressee

eOddajaCN23

V objektu se posredujejo podatki za potrebe opreme pošiljke z obrazcem CN23. Če je naslovnik v EU, je lahko celoten objekt null.

Element	Tip	Opis	Obvezen
Content	string	Vsebina	O
Quantity	decimal	Količina	O
NetWeight	decimal	Neto teža	O
Value	decimal	Vrednost	O
Tariff	string	Tarifna številka	O
CountryOfOrigin	string	Država porekla	O

Tabela 18 – eOddajaCN23

C# implementacija

```
public class eOddajaCN23
{
    public string Content { get; set; }
    public decimal Quantity { get; set; }
    public decimal NetWeight { get; set; }
    public decimal Value { get; set; }
    public string Tariff { get; set; }
    public string CountryOfOrigin { get; set; }
}
```

Implementacija 10 – eOddajaCN23

eOddajaInternational

Objekt je namenjen posredovanju podatkov za pošiljko v mednarodnem prometu. Če objekta ne potrebujete, je lahko null.

Element	Tip	Opis	Obvezen
NonDeliveryInstruction	EnumNonDeliveryInstruction	Navodila za nevročljivost	O
CategoryExplanation	String	Razlaga, če je izbrana vrsta pošiljke pod Ostalo	P
CategoryType	EnumInternationalCategoryType	Vrsta pošiljke	O
ImporterReference	String	Davčna številka (pravne osebe)	P
ImporterPhone	String	Telefonska številka	P
ImporterEmail	String	Elektronski naslov	N
Comment	String	Komentar	N
License	String	Licenca	N
Certificate	String	Certifikat	N
Invoice	String	Račun	N
CodForeign	Decimal	Vrednost valute	P
CodForeighCurrency	String	Valuta	P

Tabela 19 - eOddajaInternational

C# implementacija

```
public class eOddajaInternational
{
    public EnumNonDeliveryInstruction nonDeliveryInstruction { get; set; }
    public EnumInternationalCategoryType categoryType { get; set; }
    public string categoryExplanation { get; set; }
    public string importerReference { get; set; }
    public string importerPhone { get; set; }
    public string importerEmail { get; set; }
    public string comments { get; set; }
    public string license { get; set; }
    public string certificate { get; set; }
    public string invoice { get; set; }
    public decimal codForeign { get; set; }
    public string codForeighCurrency { get; set; }
}
```

Implementacija 11 – eOddajaInternational

EnumNonDeliveryInstruction

Seznam statusov za nevročljivost

Element	Tip	Opis	Obvezen
RETURN_TO_SENDER	int;1	Pošiljka se vrne pošiljatelju	
TREAT_AS_ABANDONED	int;2	Pošiljka je zavrnjena	

Tabela 20 - EnumNonDeliveryInstruction

EnumInternationalCategoryType

Seznam statusov za kategorijo

Element	Tip	Opis	Obvezen
GIFT	int;1	Darilo	
DOCUMENT	int;2	Dokument	
COMERCIALSAMPLE	int;3	Komercialni vzorci	
RETURNEDGOODS	int;4	Vračilo blaga	
SALEOFGOODS	int;5	Prodaja blaga	
OTHER	int;6	Ostalo	

Tabela 21 - EnumInternationalCategoryType

eOddajaUPN

Objekt je namenjen za posredovanje podatkov o plačilnem instrumentu UPN. V primeru, da UPN-a ni, je lahko objekt prazen.

Element	Tip	Opis	Obvezen
UPNType	string	Vrsta vplačilnega dokumenta	P
UPNPurpose1	string	Namen 1	P
UPNPurpose2	string	Namen 2	P
UPNModel	string	Model	P
UPNReference	string	Referenca	P
UPNIBAN	string	Številka TRR	P

Tabela 22 - eOddajaUPN

C# implementacija

```
public class eOddajaUPN
{
    public EnumUPNType UPNtype { get; set; }
    public string UPNpurpose1 { get; set; }
    public string UPNpurpose2 { get; set; }
    public string UPNmodel { get; set; }
    public string UPNreference { get; set; }
    public string UPNiban { get; set; }
}
```

Implementacija 12 - eOddajaUPN

EnumUPNType

Seznam tipov UPN

Element	Tip	Opis	Obvezen
UPN_WITH_OCR	int;2	UPN z OCR-kodo	
UPN_WITHOUT_OCR	int;3	UPN brez OCR-kode	
HUB	int;4	HUB	
UPN_WITH_QR	int;5	UPN z QR-kodo	
UPN_WITHOUT_QR	int;6	UPN brez QR-kode	

Tabela 23 - EnumUPNType

CN23 servis

Pomoč

Namenjena za prikaz podatkovnega modela in metod za potrebe izvedbe implementacije.

URL: /

Metoda : **GET**

C_Document – Oddaja podatkov o CN23 pošiljki

Namenjena za oddajo podatkov o CN23-pošiljki.

URL: **/CN23/C_Document**

Metoda : **POST**

Parametri: **CN23Request**

Odgovor: **CN23Response**

V primeru uspešne poizvedbe CN23Response.Code=200 se v CN23Response.Value nahaja kratka številka oddaje podatkov.

R_Report – Pridobivanje PDF-dokumenta – obrazca o CN23 pošiljki

Namenjena za pridobivanje CN23-obrazca v PDF-obliki.

URL: **/CN23/R_Report**

Metoda : **POST**

Parametri: **CN23Request**

Odgovor: **CN23Response**

V primeru uspešne poizvedbe CN23Response.Code=200 se v CN23Response.Value nahaja PDF-dokument, zapisan v obliki BASE64.

CN23-entitete

CN23Document

Objekt je namenjen za posredovanje CN23 podatkov o pošiljki.

Element	Tip	Opis	Obvezen
parcelType	Int	Tip pošiljke	P
docNumber	String(13)	Črna koda	P
consignee	Consignee	Naslovník	O
shipment	Shipment	Podatki o postavkah	O

Tabela 24 – CN23Document

C# implementacija

```
public class CN23Document
{
    public int parcelType{ get; set; }
    public string docNumber { get; set; }
    public Consignee consignee { get; set; }
    public Shipment shipment { get; set; }
}
```

Implementacija 13 – CN23Document

Consignee

Objekt je namenjen za posredovanje informacij o naslovniku CN23 pošiljke. Podatki objekta so obvezni.

Element	Tip	Opis	Obvezen
name	String(50)	Ime in priimek	O
surname	String(50)	Dodaten naziv	P
organizationName	String(50)	Naziv organizacije	P
street	String(45)	Ulica	O
houseNumber	String(8)	Hišna številka	O
post	String(9)	Poštna številka	O
city	String(35)	Naziv pošte	O
countryCode	String(2)	Koda države	O
phone	String(50)	Telefon	P
email	String(250)	Elektronski naslov	P

Tabela 25 - Consignee

C# implementacija

```
public class Consignee
{
    public string name { get; set; }
    public string surname { get; set; }
    public string organizationName { get; set; }
    public string street { get; set; }
    public string houseNumber { get; set; }
    public string post { get; set; }
    public string city { get; set; }
    public string countryCode { get; set; }
    public string phone { get; set; }
    public string email { get; set; }
}
```

Implementacija 14- Consignee

Shipment

Objekt je namenjen za posredovanje informacij o postavkah CN23 pošiljke. Podatki objekta so obvezni.

Element	Tip	Opis	Obvezen
items	Item[]	Postavka	P
category	Category	Kategorija	P
licence	String(35)	Številka licence	P
certificate	String(35)	Številka certifikata	P
invoice	String(35)	Številka računa	P
comment	String(250)	Opomba	P
grossWeightTotal	Decimal	Skupna teža	O
amountTotal	Amounttotal	Skupna vrednost	O
amountCOD	Amountcod	Skupna vrednost odkupnine Koda za nevročljivost	O
nonDeliveredStatusCode	String	(ReturnToSender, CancelShipment)	O
articles	Articles[]	SI	O

Tabela 26 - Shipment

C# implementacija

```
public class Shipment
{
    public Item[] items { get; set; }
    public Category category { get; set; }
    public string licence { get; set; }
    public string certificate { get; set; }
    public string invoice { get; set; }
    public string comment { get; set; }
    public decimal grossWeightTotal { get; set; }
    public Amounttotal amountTotal { get; set; }
    public Amountcod amountCOD { get; set; }
    public string nonDeliveredStatusCode { get; set; }
    public Article[] articles { get; set; }
}
```

Implementacija 15 – Shipment

Category

Objekt je namenjen za posredovanje informacij o tipu pošiljke. Podatki objekta so obvezni.

Element	Tip	Opis	Obvezen
categoryType	String	Tip (Gift, Document, ComercialSample, ReturnedGoods, SaleOfGoods, Other)	O
explanation	String(250)	Opomba postavke	P

Tabela 27 - Category

C# implementacija

```
public class Category
{
    public string categoryType { get; set; }
    public string explanation { get; set; }
}
```

Implementacija 16 – Category

AmountTotal

Objekt je namenjen posredovanju informacij o skupnem znesku CN23 pošiljke.

Element	Tip	Opis	Obvezen
amountValue	Int	Skupni znesek	P
currencyCode	String(3)	ISO-kratka denarne valute (EUR)	P

Tabela 28 – AmountTotal

C# implementacija

```
public class Amounttotal
{
    public int amountValue { get; set; }
    public string currencyCode { get; set; }
}
```

Implementacija 17 - AmountTotal

AmountCOD

Objekt je namenjen za posredovanje informacij o odkupninah.

Element	Tip	Opis	Obvezen
amountValue	Int	Skupni znesek	P
currencyCode	String(3)	ISO-kratice denarne valute (EUR)	P

Tabela 29 - AmountCOD

C# implementacija

```
public class Amountcod
{
    public int amountValue { get; set; }
    public string currencyCode { get; set; }
}
```

Implementacija 18 - AmountCOD

Item

Objekt je namenjen za posredovanje informacij o posamezni postavki CN23 pošiljke. Podatki objekta so obvezni.

Element	Tip	Opis	Obvezen
goodsDescription	String(250)	Opis postavke	O
quantity	Int	Količina	O
weight	Decimal	Teža	O
amount	Amount	Vrednost	O
originCountryCode	String(2)	Država porekla	O
tariffNumber	String(8)	Tarifna številka	P

Tabela 30 - Item

C# implementacija

```
public class Item
{
    public string goodsDescription { get; set; }
    public int quantity { get; set; }
    public decimal weight { get; set; }
    public Amount amount { get; set; }
    public string originCountryCode { get; set; }
    public string tariffNumber { get; set; }
}
```

Implementacija 19 - Item

Article

Objekt je namenjen za posredovanje informacij o dodatnih storitvah CN23 pošiljke. Kolekcija podatkov je obvezna tudi takrat, če dodatne storitve niso navedene.

Element	Tip	Opis	Obvezen
code	String(10)	Opis postavke	O
amount	Amount	Vrednost	O

Tabela 31 – Article

C# implementacija

```
public class Article
{
    public string code { get; set; }
    public Amount amount { get; set; }
}
```

Implementacija 20 - Article

Amount

Objekt je namenjen za posredovanje informacij o vrednostih. V primeru uporabe objekta so podatki obvezni.

Element	Tip	Opis	Obvezen
amountValue	decimal	Vrednost v valuti	O
currencyCode	string(3)	ISO-kratka denarne valute	O

Tabela 32 – Amount

C# implementacija

```
public class Amount
{
    public decimal amountValue { get; set; }
    public string currencyCode { get; set; }
}
```

Implementacija 21 - Amount

Implementacija C#

Pomožne metode

Decompress

```
public string Decompress(string aValue)
{
    byte[] gzBuffer = Convert.FromBase64String(aValue);
    using (MemoryStream ms = new MemoryStream())
    {
        int msgLength = BitConverter.ToInt32(gzBuffer, 0);
        ms.Write(gzBuffer, 4, gzBuffer.Length - 4);
        byte[] buffer = new byte[msgLength];
        ms.Position = 0;
        using (System.IO.Compression.GZipStream zip = new System.IO.Compression.GZipStream(ms,
            System.IO.Compression.CompressionMode.Decompress))
        {
            zip.Read(buffer, 0, buffer.Length);
        }
        return System.Text.Encoding.Unicode.GetString(buffer, 0, buffer.Length);
    }
}
```

Metoda 1 – C# - Pomožna metoda - Decompress

Post

```
private string Post(string aUsername, string aPassword, string aUrl, string aData)
{
    string result = string.Empty;
    using (var client = new WebClient())
    {
        client.Headers[HttpRequestHeader.ContentType] = "application/json";
        client.Encoding = Encoding.UTF8;
        if (!String.IsNullOrEmpty(aUsername) &&
            !String.IsNullOrEmpty(aPassword))
        {
            string credentials = Convert.ToBase64String(Encoding.ASCII.GetBytes(aUsername + ":" +
                aPassword));
            client.Headers[HttpRequestHeader.Authorization] = string.Format("Basic {0}",
                credentials);
        }
        result = client.UploadString(aUrl, "POST", aData);
        if (client.ResponseHeaders["Content-Encoding"] == "gzip")
            result = Decompress(result);
    }
    return result;
}
```

Metoda 2 – C# - Pomožna metoda - Post

Post_Document()

Metoda je namenjena za pošiljanje vseh podatkov, ki so potrebni za kreiranje dokumenta CN23.

```
private void Post_Document()
{
    string route = "CN23/C_Document";

    Authentication authentication = new Authentication()
    {
        KomitentId = Convert.ToInt32(txtKomitentId.Text),
        PogodbaId = Convert.ToInt32(txtPogodbaId.Text),
        PodruznicaId = Convert.ToInt32(txtPodruznicaId.Text),
        PostaId = txtPostaId.Text
    };

    CN23Document document = new CN23Document();
    Consignee consignee = new Consignee();
    consignee.name = "Naslovník-Naziv";
    consignee.surname = "Naslovni-DodatenNaziv";
    consignee.street = "Naslovik-Naslov";
    consignee.post = "2000";
    consignee.city = "Maribor";
    consignee.houseNumber = string.Empty;
    consignee.countryCode = "SI";
    consignee.phone = "Telefon";
    consignee.email = "nekdo@nekje.si";

    Shipment shipment = new Shipment();
    List<Item> items = new List<Item>();

    Item item = new Item();
    item.goodsDescription = "Postavka-Vsebins";
    item.quantity = 1;
    item.weight = 10500;
    item.originCountryCode = "SI";
    Amount amount = new Amount();
    amount.amountValue = 10;
    amount.currencyCode = "EUR";
    item.amount = amount;
    items.Add(item);

    Category category = new Category();
    category.categoryType = "Other";
    category.explanation = "Opomba";

    shipment.licence = "Licenca";
    shipment.certificate = "Certifikat";
    shipment.invoice = "Račun";
    shipment.comment = "Ref2";
    shipment.category = category;
    shipment.nonDeliveredStatusCode = "ReturnToSender";
    shipment.items = items.ToArray();

    var articles = new List<Article>();
    shipment.articles = articles.ToArray();
}
```

```

document.shipment = shipment;
document.consignee = consignee;
document.docNumber = txtCrtnaKoda.Text;

CN23Request request = new CN23Request
{
    Authentication = authentication,
    Data = document,
    Route = route
};

var JSONRequest = JsonConvert.SerializeObject(request);
string url = _url + "/" + route + "/";
string JSONResponse = Post(txtBasicAuthUporabnik.Text, txtBasicAuthGeslo.Text, url,
JSONRequest);
CN23Response response = JsonConvert.DeserializeObject<CN23Response>(JSONResponse);
}

```

Metoda 3 – C# - Post_Document

Get_Report()

Metoda je namenjena za pridobivanje obrazca CN23v PDF-obliki. Obrazec se vrne v obliki zapisa BASE64.

```

private void Post_Report()
{
    string route = "CN23/R_Report";
    Authentication authentication = new Authentication()
    {
        KomitentId = Convert.ToInt32(txtKomitentId.Text),
        PogodbaId = Convert.ToInt32(txtPogodbaId.Text),
        PodruznicaId = Convert.ToInt32(txtPodruznicaId.Text),
        PostaId = txtPostaId.Text
    };

    CN23Document document = new CN23Document();
    document.docNumber = txtCrtnaKoda.Text;

    CN23Request request = new CN23Request
    {
        Authentication = authentication,
        Data = document,
        Route = route
    };

    var JSONRequest = JsonConvert.SerializeObject(request);
    string url = _url + "/" + route + "/";
    string JSONResponse = Post(txtBasicAuthUporabnik.Text, txtBasicAuthGeslo.Text, url,
JSONRequest);
    CN23Response response = JsonConvert.DeserializeObject<CN23Response>(JSONResponse);

    string dokumentIme = root.docNumber + ".pdf";
    using (FileStream stream = System.IO.File.Create(dokumentIme))
    {
        byte[] byteArray = Convert.FromBase64String((string)response.Value);
        stream.Write(byteArray, 0, byteArray.Length);
        stream.Flush();
    }
}

```

Metoda 4 – C# - Get_Report

Implementacija v PHP

Pomožne metode

Decompress

```
<?php
function Decompress(string $arg1)
{   return gzdecode(substr(base64_decode($arg1), 4)); }
?>
```

Metoda 5 – PHP - Decompress

Post

```
<?php

$authentication = array("KomitentId"=>0, "Pogodbald"=>0, "Podruznicald"=>0, "Postald"=>"0");
$request = json_encode(array("Authentication" => $authentication ));

function Post($aUsername, $aPassword, $aUrl, $aData)
{
    $curl = curl_init($aUrl);
    curl_setopt($curl, CURLOPT_HEADER, false);
    curl_setopt($curl, CURLOPT_RETURNTRANSFER, true);
    curl_setopt($curl, CURLOPT_HTTPAUTH, CURLAUTH_BASIC);
    curl_setopt($curl, CURLOPT_USERPWD, "$aUsername:$aPassword");
    curl_setopt($curl, CURLOPT_HTTPHEADER,
                array("Content-type: application/json"));
    curl_setopt($curl, CURLOPT_POST, true);
    curl_setopt($curl, CURLOPT_POSTFIELDS, $aData);
    $json_response = curl_exec($curl);
    $status = curl_getinfo($curl, CURLINFO_HTTP_CODE);
    if ( $status != 200 ) {die("Error: call to URL $url failed with status $status, response
$json_response, curl_error " . curl_error($curl) . ", curl_errno " . curl_errno($curl));}
    curl_close($curl);
    return Decompress($json_response);
}
?>
```

Metoda 6 – PHP - Post

Psevdokoda za druge programske jezike

Dekodiranje odgovora iz strežnika

1. Niz iz strežnika v Base64 obliki

```
"LgAAAB+LCAAAAAAABACrZlBicGbIZ0hhSAWyrBhMGQyAUAfIDgKK1jKUQMWVgLCWAQC5lo7KLgAAAA
=="
```

2. Niz pretvorimo iz Base64 oblike v polje zlogov (šestnajstiški zapis)

```
2E-00-00-00-1F-8B-08-00-00-00-00-00-04-00-AB-66-50-62-70-66-C8-67-48-61-48-
05-B2-AC-18-4C-19-0C-80-50-07-C8-0E-02-8A-96-32-94-40-C5-95-80-B0-96-01-00-
B9-96-8E-CA-2E-00-00-00
```

V desetiškem zapisu po zlogih:

0	1	2	3	4	5	6	7	8	9
46	0	0	0	31	139	8	0	0	0
10	11	12	13	14	15	16	17	18	19
0	0	4	0	171	102	80	98	112	102
20	21	22	23	24	25	26	27	28	29
200	103	72	97	72	5	178	172	24	76
30	31	32	33	34	35	36	37	38	39
25	12	128	80	7	200	14	2	138	150
40	41	42	43	44	45	46	47	48	49
50	148	64	197	149	128	176	150	1	0
50	51	52	53	54	55	56	57		
185	150	142	202	46	0	0	0		

3. Niz po odstranitvi prvih štirih zlogov:

```
1F-8B-08-00-00-00-00-00-04-00-AB-66-50-62-70-66-C8-67-48-61-48-05-B2-AC-18-
4C-19-0C-80-50-07-C8-0E-02-8A-96-32-94-40-C5-95-80-B0-96-01-00-B9-96-8E-CA-
2E-00-00-00
```

V desetiškem zapisu:

0	1	2	3	4	5	6	7	8	9
31	139	8	0	0	0	0	0	4	0
10	11	12	13	14	15	16	17	18	19
171	102	80	98	112	102	200	103	72	97
20	21	22	23	24	25	26	27	28	29
72	5	178	172	24	76	25	12	128	80
30	31	32	33	34	35	36	37	38	39
7	200	14	2	138	150	50	148	64	197
40	41	42	43	44	45	46	47	48	49
149	128	176	150	1	0	185	150	142	202
50	51	52	53						
46	0	0	0						

4. Nato niz dekompresiramo po metodi gzip:

7B-00-22-00-43-00-6F-00-64-00-65-00-22-00-3A-00-35-00-30-00-30-00-2C-00-22-00-
52-00-6F-00-75-00-74-00-65-00-22-00-3A-00-22-00-22-00-7D-00

Zapis v unicode kodni tabeli: {"Code":500,"Route":""}

Implementacija REST client

Post_Document

Request

Metoda: **POST**

URL: https://tstePortal.posta.si/Services/eSpremnica.Provider/WebApi/api/CN23/C_Document

Authorization: **BasicAuth**

Content-Type: **application/json**

RAW:

```
{
  "Data": {
    "docNumber": "__BARCODE__",
    "consignee": {
      "name": "Naslovník-
Naziv",
      "surname": "Naslovni-DodatenNaziv",
      "organizationName": null,
      "street": "Naslovik-
Naslov",
      "houseNumber": "",
      "post": "2000",
      "city": "Maribor",
      "countryCode": "SI",
      "phone": "Telefon",
      "email": "nekdo@nekje.si",
      "shipment": {
        "items": [
          {
            "goodsDescription": "Postavka-
Vsebina",
            "quantity": 1,
            "weight": 1.5,
            "amount": {
              "amountValue": 10.0,
              "currencyCode": "EUR"
            },
            "originCountryCode": "SI",
            "tariffNumber": null
          }
        ],
        "category": {
          "categoryType": "Other",
          "explanation": "Opomba"
        },
        "licence": "Licenca",
        "certificate": "Certifikat",
        "invoice": "Račun",
        "comment": "Ref2",
        "grossWeightTotal": 0.0,
        "amountTotal": null,
        "amountCOD": null,
        "nonDeliveredStatusCode": "ReturnToSender",
        "articles": []
      }
    },
    "Authentication": {
      "PostaId": "__POSTAID__",
      "PodružnicaId": "__PODRUZNICAID__",
      "PogodbaId": "__POGODBAID__",
      "KomitentId": "__KOMITENTID__",
      "Route": "CN23/C_Document"
    }
  }
}
```

Response

Content-Encoding: **gzip**

Content-Type: **application/json**

```
{
  "Code": 200,
  "Value": "db3b8f9f-de48-4f3e-a6e4-ac271720a74e",
  "Route": "CN23/C_Document"
}
```

Implementacija 22 – REST- Post_Document

Get_Report()

Request

Metoda: **POST**

URL: https://tstePortal.posta.si/Services/eSpremnica.Provider/WebApi/api/CN23/R_Report

Authorization: **BasicAuth**

Content-Type: **application/json**

RAW:

```
{
  "Data": {"docNumber": "__BARCODE__"},
  "Authentication": {"PostaId": "__POSTAID__", "PodruznicaId": "__PODRUZNICAID__", "PogodbaId": "__POGODBAID__", "KomitentId": "__KOMITENTID__"},
  "Route": "CN23/R_Report"
}
```

Response

Content-Encoding: **gzip**

Content-Type: **application/json**

```
{
  "Code": 200,
  "Value": "db3b8f9fde484f3ea6e4ac271720a74e....",
  "Route": "CN23/R_Report"
}
```

Implementacija 23 – REST – Get_Report

Kazalo tabel

Tabela 1 – Spremembe dokumenta	4
Tabela 2 - ProviderRequest.....	6
Tabela 3 – Authentication	6
Tabela 4 – Data.....	7
Tabela 5 – EnumeOddajaRequestDataType	7
Tabela 6 – ProviderResponse	8
Tabela 7 – EnumeResponseCode	8
Tabela 8 – Value	9
Tabela 9 – ProcessStatus.....	9
Tabela 10 - EnumTransactionCode.....	9
Tabela 11 - Error	10
Tabela 12 – CN23DocumentValue	10
Tabela 13 – eOddajaData	12
Tabela 13 – eOddajaParcel.....	13
Tabela 14 – EnumParcelContent.....	13
Tabela 15 – eOddajaService.....	14
Tabela 16 – eOddajaAddressee.....	14
Tabela 17 – eOddajaCN23.....	15
Tabela 18 - eOddajaInternational.....	16
Tabela 19 - EnumNonDeliveryInstruction	17
Tabela 20 - EnumInternationalCategoryType.....	17
Tabela 21 - eOddajaUPN.....	18
Tabela 22 - EnumUPNType.....	18
Tabela 23 – CN23Document	20
Tabela 24 - Consignee.....	21
Tabela 25 - Shipment.....	22
Tabela 26 - Category.....	23
Tabela 27 – AmountTotal	23
Tabela 28 - AmountCOD	24
Tabela 29 - Item	25
Tabela 30 – Article.....	26
Tabela 31 – Amount	26

Kazalo metod

Metoda 1 – C# - Pomožna metoda - Decompress.....	27
Metoda 2 – C# - Pomožna metoda - Post.....	27
Metoda 3 – C# - Post_Document.....	29
Metoda 4 – C# - Get_Report.....	29
Metoda 5 – PHP - Decompress.....	30
Metoda 6 – PHP - Post.....	30

Kazalo implementacij

Implementacija 1 - ProviderRequest.....	6
Implementacija 2 - Authentication.....	7
Implementacija 3 – ProviderResponse.....	8
Implementacija 4 - Error.....	10
Implementacija 5 - CN23DocumentValue.....	10
Implementacija 6 - eOddajaData.....	12
Implementacija 7 – eOddajaParcel.....	13
Implementacija 8 – eOddajaService.....	14
Implementacija 9 – eOddajaAddressee.....	14
Implementacija 10 – eOddajaCN23.....	15
Implementacija 11 – eOddajaInternational.....	16
Implementacija 12 - eOddajaUPN.....	18
Implementacija 13 – CN23Document.....	20
Implementacija 14- Consignee.....	21
Implementacija 15 – Shipment.....	22
Implementacija 16 – Category.....	23
Implementacija 17 - AmountTotal.....	23
Implementacija 18 - AmountCOD.....	24
Implementacija 19 - Item.....	25
Implementacija 20 - Article.....	26
Implementacija 21 - Amount.....	26
Implementacija 22 – REST- Post_Document.....	33
Implementacija 23 – REST – Get_Report.....	34