Spletni servis za sprejem KIR KPR

Področje: DDV

Verzija: 0.03

Kazalo

[1. Uvod 4](#_Toc187322006)

[2. Funkcionalnosti servisa 4](#_Toc187322007)

[3. Dostop do servisa 4](#_Toc187322008)

[4. Format knjige računov 5](#_Toc187322009)

[5. Definicija vmesnika REST servisa 5](#_Toc187322010)

[5.1. Splošni parametri klicev 5](#_Toc187322011)

[Avtentikacija 5](#_Toc187322012)

[Tip kodiranja vhodnih parametrov 5](#_Toc187322013)

[Tip kodiranja izhodnih parametrov 5](#_Toc187322014)

[5.2. Splošne HTTP kode odgovorov 6](#_Toc187322015)

[5.3. Metode za prijavo – /api/v1/Auth 6](#_Toc187322016)

[Metoda /api/v1/Auth/Representing/{taxPayerId}/{taxPayerType} 7](#_Toc187322017)

[Metoda /api/v1/Auth/Certificate/Representing/{taxPayerId}/{taxPayerType} 8](#_Toc187322018)

[5.4. Metode za delo s knjigami računov - /api/v1/InvoiceBook 8](#_Toc187322019)

[Metoda /api/v1/InvoiceBook 8](#_Toc187322020)

[Metoda /api/v1/InvoiceBook/Status/{edpId} 10](#_Toc187322021)

[5.5. Ostale metode v zvezi s knjigami - /api/v1/Documents 11](#_Toc187322022)

[Metoda /api/v1/Documents/OpenPeriods/{formCode} 11](#_Toc187322023)

[Metoda /api/v1/Documents/Permissions 12](#_Toc187322024)

[5.6. Metode v zvezi z delovanjem servisa - /api/v1/Heartbeat 14](#_Toc187322025)

[Metoda /api/v1/Heartbeat 14](#_Toc187322026)

[6. Definicija vmesnika SOAP servisa 15](#_Toc187322027)

[6.1. Obravnavanje napak 15](#_Toc187322028)

[6.2. Servis za prijavo 16](#_Toc187322029)

[Metoda LoginUsingClientCertificate 16](#_Toc187322030)

[6.3. servis za vsebinske funkcionalnosti 16](#_Toc187322031)

[Metoda UploadInvoiceBook 16](#_Toc187322032)

[Metoda GetInvoiceBookStatus 18](#_Toc187322033)

[Metoda GetOpenPeriods 18](#_Toc187322034)

[Metoda GetPermissions 18](#_Toc187322035)

[7. Validacija 19](#_Toc187322036)

[8. Priloge 20](#_Toc187322037)

# Uvod

Spletni servis, opisan v tem dokumentu, omogoča oddajo podatkov iz KIR (knjiga izdanih računov) in KPR (knjiga prejetih računov) za uporabnike sistema eDavki.

Spletni servis ima REST (podpira podatke, kodirane v JSON) in SOAP vmesnik.

# Funkcionalnosti servisa

Spletni servis ima naslednje funkcionalnosti:

* Prijava, ki je omogočena z:
  + digitalnimi potrdili, podprtimi v eDavkih;
  + avtentikacijskimi žetoni, ki jih izda OAuth spletni servis eDavkov.
* Preverjanje pravic in pooblaščencev za oddajo knjig.
* Oddaja knjig (prenos datoteke v sistem eDavki).
* Preverjanje statusa oddane knjige.
* Preverjanje odprtih obdobij za oddajo knjig.

# Dostop do servisa

Spletni servis je na voljo na naslednjih URL:

* REST servis:
  + Produkcijsko okolje: <https://edavki.durs.si/InvoiceBookService/>
  + Testno okolje: <https://beta.edavki.durs.si/InvoiceBookService/>
* SOAP servis za prijavo:
  + Produkcijsko okolje: <https://edavki.durs.si/InvoiceBookService/SoapService/Login/>
  + Testno okolje: <https://beta.edavki.durs.si/InvoiceBookService/SoapService/Login/>
* SOAP servis za vsebinske funkcionalnosti:
  + Produkcijsko okolje: <https://edavki.durs.si/InvoiceBookService/SoapService/>
  + Testno okolje: <https://beta.edavki.durs.si/InvoiceBookService/SoapService/>

Vmesnik REST spletnega servisa si je mogoče ogledati in preizkusiti s pomočjo Swagger spletnega vmesnika na naslednjih naslovih:

* Produkcijsko okolje: <https://edavki.durs.si/InvoiceBookService/swagger/index.html>
* Testno okolje: <https://beta.edavki.durs.si/InvoiceBookService/swagger/index.html>

WSDL SOAP servisa je na voljo na naslednjih naslovih:

* SOAP servis za prijavo:
  + Produkcijsko okolje: <https://edavki.durs.si/InvoiceBookService/SoapService/Login/?singleWsdl>
  + Testno okolje: <https://beta.edavki.durs.si/InvoiceBookService/SoapService/Login/?singleWsdl>
* SOAP servis za vsebinske funkcionalnosti:
  + Produkcijsko okolje: <https://edavki.durs.si/InvoiceBookService/SoapService/?singleWsdl>
  + Testno okolje: <https://beta.edavki.durs.si/InvoiceBookService/SoapService/?singleWsdl>

# Format knjige računov

Servis sprejema knjige, formatirane v enem izmed naslednjih treh formatov: XML, JSON, CSV. Specifikacije formatov so priložene v poglavju »Priloge«.

# Definicija vmesnika REST servisa

V tem poglavju so opisane posamezne metode REST servisa, združene po sklopih funkcionalnosti.

## Splošni parametri klicev

Pri vseh klicih servisa se uporablja nekaj splošnih parametrov, ki se pošiljajo v glavi zahtevka.

### Avtentikacija

Za dostop do vseh funkcionalnosti servisa (razen za prijavo samo) se mora uporabnik avtenticirati. To stori tako, da v klicu metode pošlje avtentikacijski žeton, ki ga pridobi pri prijavi (glej podpoglavje 5.3). Žeton je potrebno poslati v glavi zahtevka v polju Authorization. Vrednost v polju mora imeti naslednjo obliko:

Bearer {žeton}

Primer celotnega polja:

Authorization: Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJzdWIiOiI1NDYxMTMxMCIsImp0aSI6IjA0ZDgwYTYxLTk1MTEtNDEzNy04ODI1LTEwNTc4MzllZTkxYSIsImlhdCI6MTcyMjUwNTExMSwiTG9naW5JZCI6IjI3OSIsIkxvZ2luVHlwZSI6IkNlcnRpZmljYXRlIiwiVXNlclByb2ZpbGVJZCI6IjQzMCIsIkluZGl2aWR1YWxQZXJzb25JZCI6IjU0NjExMzEwIiwiVGF4UGF5ZXJJZCI6IjU0NjExMzEwIiwiVGF4UGF5ZXJUeXBlIjoiRk8iLCJSZXByZXNlbnRpbmdJZCI6IjU0NjExMzEwIiwiUmVwcmVzZW50aW5nVHlwZSI6IkZPIiwiZXhwIjoxNzIyNTkxNTExLCJpc3MiOiJodHRwczovL2R1cnN3ZWIuZW5kYXZhLm5ldDo0NjIvIiwiYXVkIjoiaHR0cHM6Ly9kdXJzd2ViLmVuZGF2YS5uZXQ6NDYyLyJ9.pl4AmJkVRTZkmYKuKCz13xkvS5V2cuIhRr0qvAnb5bM

### Tip kodiranja vhodnih parametrov

Tip kodiranja vhodnih parametrov je definiran v glavi zahtevka v polju Content-Type. Podprti tipi so »application/json« za JSON (izjema je metoda /api/v1/InvoiceBook – glej spodaj).

Primer celotnega polja:

Content-Type: application/json

### Tip kodiranja izhodnih parametrov

Tip kodiranja izhodnih parametrov je definiran v glavi zahtevka v polju accept. Podprti tipi so »application/json« za JSON.

Primer celotnega polja:

accept: application/json

## Splošne HTTP kode odgovorov

V primeru uspešnega klica servisa le-ta v splošnem odgovori s HTTP kodo 200 OK in za metodo specifičnim odgovorom v obliki JSON.

V primeru, ko pri klicu servisa pride do napake, servis odgovori s primerno HTTP kodo. V večini primerov vrne tudi JSON odgovor s podrobnejšimi podatki o vzroku napake tipa *ErrorResponse*:

ErrorResponse{

|  |  |
| --- | --- |
| errorReason | ErrorReason integer($int32)  Tip napake:  0 = BadRequest  1 = Unauthorized  2 = Forbidden  3 = NotFound  4 = ServerError  Enum: [ 0, 1, 2, 3, 4 ] |
| errorMessage | string nullable: true  Opis napake. |
| incidentId | string nullable: true  Opcijski ID incidenta. |

}

Pri vseh klicih servisa so v primeru napak možne naslednje HTTP kode odgovora:

* 400 Bad Request: napačni vhodni podatki
* 401 Unauthorized: uporabnik se ni avtenticiral z veljavnim prijavnim mehanizmom. V večini primerov je telo odgovora prazno, izjemoma lahko vsebuje JSON odgovor.
* 500 InternalServerError: interna napaka na servisu. Pri tej napaki je izpolnjeno opcijsko polje incidentId, ki je namenjeno lažjemu sledenju napaki.

Servis lahko v primeru napak vrne tudi druge HTTP kode, ki so specifične za posamezne metode servisa.

## Metode za prijavo – /api/v1/Auth

S klicem ene izmed metod za prijavo se uporabnik hkrati avtenticira in izbere zastopanega davčnega zavezanca. Servis podpira dva načina avtentikacije:

* Z digitalnimi potrdili, podprtimi v eDavkih. To je priporočen način avtentikacije.
* Z avtentikacijskimi žetoni, ki jih izda OAuth spletni servis eDavkov. Ta način naj bi se uporabljal izjemoma, ko iz tehničnih razlogov ni možna avtentikacija z digitalnimi potrdili.

### Metoda /api/v1/Auth/Representing/{taxPayerId}/{taxPayerType}

* Opis: metoda omogoča izbiro zastopanega davčnega zavezanca, uporabnik se avtenticira z avtentikacijskim žetonom. Obstajata dve možnosti:
  + Uporabnik uporabi žeton, ki ga je izdala eDavki OAuth storitev. Ta izda žeton, v katerem zastopani davčni zavezanec še ni izbran.
  + Uporabnik uporabi predhodno izdan žeton, ki ga je pridobil s klicem te metode ali metode /api/v1/Auth/Certificate/Representing/{taxPayerId}/{taxPayerType} (glej spodaj). Na ta način lahko uporabnik zamenja zastopanega davčnega zavezanca.
* HTTP metoda: GET
* Vhodni parametri:
  + taxPayerId (integer($int32)): davčna številka zastopanega zavezanca
  + taxPayerType (integer($int32):tip zastopanega davčnega zavezanca:
    - *0 = FO*
    - *1 = SP*
    - *2 = PO*
* Specifične HTTP kode odgovora:
  + 403 Forbidden: dostop ni dovoljen (npr. uporabnik nima pravic za izbranega zastopanega zavezanca)
* Izhodni parametri:
  + Status odgovora 200 OK:

TokenResponse{

|  |  |
| --- | --- |
| expires | string($date-time)  Datum in čas poteka žetona. |
| token | string nullable: true  Avtentikacijski žeton. |

}

* Primeri:
  + Vhodni parametri:
  + taxPayerId: 54611310

taxPayerType: 0

* + Odgovor za HTTP status 200 OK:

{

"expires": "2024-01-01T01:12:01",

"token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJzdWIiOiI1NDYxMTMxMCIsImp0aSI6IjA0ZDgwYTYxLTk1MTEtNDEzNy04ODI1LTEwNTc4MzllZTkxYSIsImlhdCI6MTcyMjUwNTExMSwiTG9naW5JZCI6IjI3OSIsIkxvZ2luVHlwZSI6IkNlcnRpZmljYXRlIiwiVXNlclByb2ZpbGVJZCI6IjQzMCIsIkluZGl2aWR1YWxQZXJzb25JZCI6IjU0NjExMzEwIiwiVGF4UGF5ZXJJZCI6IjU0NjExMzEwIiwiVGF4UGF5ZXJUeXBlIjoiRk8iLCJSZXByZXNlbnRpbmdJZCI6IjU0NjExMzEwIiwiUmVwcmVzZW50aW5nVHlwZSI6IkZPIiwiZXhwIjoxNzIyNTkxNTExLCJpc3MiOiJodHRwczovL2R1cnN3ZWIuZW5kYXZhLm5ldDo0NjIvIiwiYXVkIjoiaHR0cHM6Ly9kdXJzd2ViLmVuZGF2YS5uZXQ6NDYyLyJ9.pl4AmJkVRTZkmYKuKCz13xkvS5V2cuIhRr0qvAnb5bM"

}

### Metoda /api/v1/Auth/Certificate/Representing/{taxPayerId}/{taxPayerType}

* Opis: metoda omogoča izbiro zastopanega davčnega zavezanca, uporabnik se avtenticira z digitalnim potrdilom, ki ga uporabi pri vzpostavitvi TLS povezave.
* V vseh ostalih pogledih se metoda obnaša enako kot metoda /api/v1/Auth/Representing/{taxPayerId}/{taxPayerType} (glej zgoraj).

## Metode za delo s knjigami računov - /api/v1/InvoiceBook

### Metoda /api/v1/InvoiceBook

* Opis: metoda omogoča prenos nove knjige računov za zastopanega davčnega zavezanca. Telo klica servisa mora biti formatirano v obliki »multipart/form-data«. Vsebovati mora natanko dve sekciji:
  + Prva sekcija mora vsebovati metapodatke o knjigi računov v JSON obliki. Tip podatkov mora biti eden izmed "application/json", "application/\*+json", "text/json".
  + Druga sekcija mora vsebovati knjigo računov v binarni obliki. Tip podatkov mora biti "application/octet-stream" ali "application/x-zip-compressed".
* HTTP metoda: POST
* Vhodni parametri (metapodatki o knjigi v JSON obliki v prvi sekciji):

UploadInvoiceBookRequest{

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| period | Period{   |  |  | | --- | --- | |  | Obdobje knjige | | begin | string($date-time) nullable: true  Začetek obdobja | | end | string($date-time) nullable: true  Konec obdobja |   } |
| format | InvoiceBookFormat integer($int32)  Format knjige.  0 = XML  1 = JSON  2 = CSV  Enum: [ 0, 1, 2 ] |
| schemaVersion | integer($int32)  Verzija sheme knjige računov. |
| correlationId | string nullable: true  Klientski korelacijski ID. Namenjen lažjemu sledenju procesa obdelave knjig. |

}

* Specifične HTTP kode odgovora:
* Izhodni parametri:
  + Status odgovora 200 OK:

UploadInvoiceBookResponse{

|  |  |
| --- | --- |
| edpId | string nullable: true  Oznaka knjige v sistemu eDavki. |
| status | InvoiceBookStatus integer($int32)  Status knjige (glej spodaj).  Enum: [ 0, 1, 2, 3, 4, 5, 6 ] |

}

* Primeri:
  + Vhodni parametri (prva sekcija):

{

"period": {

"begin": "2024-01-01",

"end": "2024-01-31"

},

"format": 0,

"schemaVersion": 1,

"correlationId": "098sdkljf8907asd8f"

}

* + Odgovor za HTTP status 200 OK:

{

"edpId": "KIR-12345678-123",

"status": 0

}

### Metoda /api/v1/InvoiceBook/Status/{edpId}

* Opis: metoda vrača podatke o statusu predhodno naložene knjige računov.
* HTTP metoda: GET
* Vhodni parametri:
  + *edpId*: oznaka knjige v sistemu eDavki
* Specifične HTTP kode odgovora:
  + 403 Forbidden: knjiga računov s podano oznako ne pripada zastopanemu davčnemu zavezancu.
  + 404 Not Found: knjiga računov s podano oznako ne obstaja v sistemu eDavki
* Izhodni parametri:
  + Status odgovora 200 OK:

InvoiceBookStatusResponse{

|  |  |
| --- | --- |
| status | InvoiceBookStatus integer($int32)  Oznake statusa knjige:  0 = Uploaded – knjiga je naložena  1 = Validating – knjiga je v procesu validacije v sistemu eDavki  2 = ValidatedOk – knjiga je bila uspešno validirana v sistemu eDavki  3 = ValidationError – knjiga je bila validirana v sistemu eDavki, vendar vsebuje napake  4 = InProcess – knjiga je v procesu obdelave v zalednih sistemih  5 = ProcessedOk – knjiga je bila uspešno obdelana v zalednih sistemih  6 = ProcessedError– knjiga ni bila uspešno obdelana v zalednih sistemih  Enum: [ 0, 1, 2, 3, 4, 5, 6 ] |
| validationMessage | string nullable: true  Opcijsko poročilo o najdenih napakah pri validaciji v sistemu eDavki. Poročilo je v JSON formatu (glej poglavje 7). |
| backendMessage | string nullable: true  Opcijsko sporočilo o obdelavi v zalednih sistemih. |

}

* Primeri:
  + Odgovor JSON za HTTP status 200 OK:

{

"status": 3,

"validationMessage": "[{\"Type\":2,\"Code\":5,\"Message\":\"Oznaka države 'AB' ne obstaja v seznamu držav.\",\"BookId\":null,\"BookArrayIndex\":null},{\"Type\":2,\"Code\":7,\"Message\":\"Knjig ne morete oddati, ker v izbranem obdobju '1.6.2023 - 30.6.2023' niste zastopnik tujca v 'string'.\",\"BookId\":null,\"BookArrayIndex\":null}]",

"backendMessage": null

}

## Ostale metode v zvezi s knjigami - /api/v1/Documents

### Metoda /api/v1/Documents/OpenPeriods/{formCode}

* Opis: metoda vrača seznam odprtih obdobij za zastopanega davčnega zavezanca in specificirano kodo obrazca.
* HTTP metoda: GET
* Vhodni parametri:
  + *formCode*: oznaka tipa obrazca. Trenutno sta podprta tipa »DDV\_EVID« (za knjige računov) in informativno tudi »DDV\_O« (za obrazce DDV-O).
* Specifične HTTP kode odgovora: /
* Izhodni parametri:
  + Status odgovora 200 OK:

OpenPeriodsResponse{

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| openPeriods | [ nullable: true  Lista odprtih obdobij.  OpenPeriod{   |  |  | | --- | --- | |  | Odprto obdobje | | period | Period{   |  |  | | --- | --- | |  | Obdobje | | begin | string($date-time) nullable: true  Začetek obdobja | | end | string($date-time) nullable: true  Konec obdobja |   } | | representedForeignerId | string nullable: true  Opcijska identifikacijska oznaka zastopanega tujca. | | dueDate | string($date-time) nullable: true  Rok za oddajo |   }] |

}

* Primeri:
  + Odgovor za HTTP status 200 OK:

{

"openPeriods": [

{

"period": {

"begin": "2024-01-01T00:00:00",

"end": "2024-01-31T00:00:00"

},

"representedForeignerId": null,

"dueDate": "2024-01-31T00:00:00"

},

{

"period": {

"begin": "2024-02-01",

"end": "2024-02-29"

},

"representedForeignerId": "XY1234567890",

"dueDate": null

}

]

}

### Metoda /api/v1/Documents/Permissions

* Opis: metoda vrača seznam pravic, ki jih ima uporabnik v sistemu eDavki.
* HTTP metoda: GET
* Vhodni parametri:
  + *formCode*: opcijska oznaka tipa obrazca. Trenutno je podprt samo tip »DDV\_O«.
* Specifične HTTP kode odgovora: /
* Izhodni parametri:
  + Status odgovora 200 OK:

DocumentPermissionsResponse{

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| permissions | [ nullable: true  Lista pravic.  DocumentPermission{   |  |  | | --- | --- | |  | Specifikacija pravic za tip obrazca. | | mandator | TaxPayer{   |  |  | | --- | --- | |  | Podatki o pooblastitelju. | | id | integer($int32)  Davčna številka | | type | TaxPayerType integer($int32)  Tip davčnega zavezanca.  0 = FO  1 = SP  2 = PO  Enum: [ 0, 1, 2 ] |   } | | formCode | string nullable: true  Oznaka tipa obrazca. | | fill | boolean  Pravica za vnos obrazca. | | sign | boolean  Pravica za vložitev obrazca. | | viewSent | boolean  Pravica za pregled vloženega obrazca. |   }] |

}

* Primeri:
  + Odgovor za HTTP status 200 OK:

{

"permissions": [

{

"mandator": {

"id": 12345678,

"type": 0

},

"formCode": "DDV\_O",

"fill": true,

"sign": false,

"viewSent": true

},

{

"mandator": {

"id": 23456789,

"type": 2

},

"formCode": "DDV\_O",

"fill": true,

"sign": true,

"viewSent": true

}

]

}

## Metode v zvezi z delovanjem servisa - /api/v1/Heartbeat

### Metoda /api/v1/Heartbeat

* Opis: metoda je namenjena preverjanju delovanja servisa.
* HTTP metoda: GET
* Vhodni parametri: *-*
* Specifične HTTP kode odgovora: /
* Izhodni parametri:
  + Status odgovora 200 OK:

*HeartbeatResponse{*

|  |  |
| --- | --- |
|  | *Odgovor na poizvedbo za delovanje servisa* |
| *hostingEnvironment* | *string nullable: true*  *Oznaka izvajalnega okolja.* |
| *apiVersion* | *string nullable: true*  *Verzija vmesnika servisa.* |
| *productVersion* | *string nullable: true*  *Verzija servisa.* |
| *serverTime* | *string($date-time)*  *Trenutni čas na servisu.* |

*}*

# Definicija vmesnika SOAP servisa

Iz tehničnih razlogov je SOAP servis razdeljen na dva ločena servisa, servis za prijavo in servis za vsebinske funkcionalnosti.

Funkcionalno SOAP servis podvaja funkcionalnosti REST servisa, zato se pri tehničnih podrobnostih metod (vhodni in izhodni parametri, …) večinoma sklicujemo na ustrezne metode REST servisa.[[1]](#footnote-1)

## Obravnavanje napak

V primeru, ko klic SOAP servisa ne uspe, ta vrne objekt tipa Fault. Podrobnosti o napaki se nahajajo v elementu *detail*, vsebinsko ustrezajo HTTP kodam oz. odgovorom, ki jih v primeru napak vrača REST servis.

Primer:

<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">

<s:Body>

<s:Fault>

<faultcode>s:Client</faultcode>

<faultstring xml:lang="en-US">Unauthorized</faultstring>

<detail>

<UnauthorizedFault xmlns="http://schemas.datacontract.org/2004/07/Endava.Edp.InvoiceBook.Web.Soap" xmlns:i="http://www.w3.org/2001/XMLSchema-instance">

<Message>Niste prijavljeni oziroma ni izbrana zastopana oseba.</Message>

</UnauthorizedFault>

</detail>

</s:Fault>

</s:Body>

</s:Envelope>

## Servis za prijavo

### Metoda LoginUsingClientCertificate

* Opis: Metoda omogoča izbiro zastopanega davčnega zavezanca, uporabnik se avtenticira z digitalnim potrdilom, ki ga uporabi pri vzpostavitvi TLS povezave. Rezultat uspešnega klica metode vsebuje avtentikacijski žeton, s katerim se uporabnik avtenticira pri klicu metod servisa za vsebinske funkcionalnosti.
* Vhodni parametri:
  + taxPayerId (int): davčna številka zastopanega zavezanca
  + taxPayerType (TaxPayerType): tip zastopanega davčnega zavezanca
* Izhodni parametri pri uspešnem klicu: objekt tipa TokenResponse
* Tehnične podrobnosti: glej »Metoda /api/v1/Auth/Certificate/Representing/{taxPayerId}/{taxPayerType}«

## servis za vsebinske funkcionalnosti

### Metoda UploadInvoiceBook

* Opis: Metoda omogoča prenos nove knjige računov za zastopanega davčnega zavezanca. Metoda podpira MTOM (Message Transmission Optimization Mechanism) z XOP (XML-binary Optimized Packaging) in tako omogoča prenos knjig računov v binarni obliki (brez kodiranja v base64). Uporabo MTOM pri klicu metode močno priporočamo, še posebej pri velikih knjigah.
* Vhodni parametri: iz tehničnih razlogov se struktura zahtevka razlikuje glede na ekvivalentno metodo REST servisa: del vhodnih parametrov se nahaja v glavi zahtevka (element soapenv:Header), del pa v telesu zahtevka (element soapenv:Body).
  + Parametri v glavi SOAP zahtevka: podatki o naloženi knjigi računov, ustrezajo parametrom schemaVersion, period, format in correlationId v tipu UploadInvoiceBookRequest.
  + Parametri v telesu SOAP zahtevka (element soapenv:Body): data (byte[]): knjiga računov v binarni obliki (referenca na binarni del sporočila, če je uporabljen MTOM oz. base64 kodirana vsebina knjige, če ni uporabljen MTOM)
* Izhodni parametri pri uspešnem klicu: objekt tipa UploadInvoiceBookResponse
* Tehnične podrobnosti: glej »Metoda /api/v1/InvoiceBook«
* Primer klica z uporabo MTOM:

*POST http://localhost:5288/SoapService/ HTTP/1.1*

*Accept-Encoding: gzip,deflate*

*Content-Type: multipart/related; type="application/xop+xml"; start="<rootpart@soapui.org>"; start-info="text/xml"; boundary="----=\_Part\_41\_364276937.1727935652947"*

*SOAPAction: "http://tempuri.org/ISoapService/UploadInvoiceBook"*

*MIME-Version: 1.0*

*Authorization: Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJzdWIiOiI1NDYxMTMxMCIsImp0aSI6IjcwZmU0YzQ0LWRkZGQtNDk4MS05MzcwLTE2MGMwNzg0ZjNhNyIsImlhdCI6MTcyNzg3NTE0MCwiTG9naW5JZCI6IjM0MCIsIkxvZ2luVHlwZSI6IkNlcnRpZmljYXRlIiwiVXNlclByb2ZpbGVJZCI6Ijk3IiwiSW5kaXZpZHVhbFBlcnNvbklkIjoiNTQ2MTEzMTAiLCJUYXhQYXllcklkIjoiNTQ2MTEzMTAiLCJUYXhQYXllclR5cGUiOiJGTyIsIlJlcHJlc2VudGluZ0lkIjoiNTQ2MTEzMTAiLCJSZXByZXNlbnRpbmdUeXBlIjoiRk8iLCJleHAiOjE3Mjc5NjE1NDAsImlzcyI6Imh0dHA6Ly9sb2NhbGhvc3Q6NjM5MzkvIiwiYXVkIjoiaHR0cDovL2xvY2FsaG9zdDo2MzkzOS8ifQ.bvhMdskZDRgPKFhEaanIRrTb-J50XZ99tuO9toCNpgY*

*Content-Length: 1538*

*Host: localhost:5288*

*Connection: Keep-Alive*

*User-Agent: Apache-HttpClient/4.5.5 (Java/12.0.1)*

*------=\_Part\_41\_364276937.1727935652947*

*Content-Type: application/xop+xml; charset=UTF-8; type="text/xml"*

*Content-Transfer-Encoding: 8bit*

*Content-ID: <rootpart@soapui.org>*

*<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:tem="http://tempuri.org/" xmlns:end="http://schemas.datacontract.org/2004/07/Endava.Edp.InvoiceBook.Web.Models">*

*<soapenv:Header>*

*<tem:SchemaVersion>0</tem:SchemaVersion>*

*<tem:Period>*

*<!--Optional:-->*

*<end:Begin>2024-01-01</end:Begin>*

*<!--Optional:-->*

*<end:End>2024-01-31</end:End>*

*</tem:Period>*

*<tem:Format>0</tem:Format>*

*<tem:CorrelationId>lfkadjsfklj</tem:CorrelationId>*

*</soapenv:Header>*

*<soapenv:Body>*

*<tem:UploadInvoiceBookRequest>*

*<tem:data><inc:Include href="cid:1405716680238" xmlns:inc="http://www.w3.org/2004/08/xop/include"/></tem:data>*

*</tem:UploadInvoiceBookRequest>*

*</soapenv:Body>*

*</soapenv:Envelope>*

*------=\_Part\_41\_364276937.1727935652947*

*Content-Type: application/zip; name=test\_KIR\_CSV.zip*

*Content-Transfer-Encoding: binary*

*Content-ID: <1405716680238>*

*Content-Disposition: attachment; name="test\_KIR\_CSV.zip"; filename="test\_KIR\_CSV.zip"*

*….(binarna vsebina izpuščena)…*

*------=\_Part\_41\_364276937.1727935652947--*

### Metoda GetInvoiceBookStatus

* Opis: Metoda vrača podatke o statusu predhodno naložene knjige računov.
* Vhodni parametri:
  + edpId (string): oznaka knjige v sistemu eDavki
* Izhodni parametri pri uspešnem klicu: objekt tipa InvoiceBookStatusResponse
* Tehnične podrobnosti: glej »Metoda /api/v1/InvoiceBook/Status/{edpId}«

### Metoda GetOpenPeriods

* Opis: Metoda vrača seznam odprtih obdobij za zastopanega davčnega zavezanca in specificirano kodo obrazca.
* Vhodni parametri:
  + formCode (string): oznaka tipa obrazca. Trenutno je podprt samo tip »DDV\_O«
* Izhodni parametri pri uspešnem klicu: objekt tipa OpenPeriodsResponse
* Tehnične podrobnosti: glej »Metoda /api/v1/Documents/OpenPeriods/{formCode}«

### Metoda GetPermissions

* Opis: metoda vrača seznam pravic, ki jih ima uporabnik v sistemu eDavki.
* Vhodni parametri:
  + formCode (string): opcijska oznaka tipa obrazca. Trenutno je podprt samo tip »DDV\_O«
* Izhodni parametri pri uspešnem klicu: objekt tipa DocumentPermissionsResponse
* Tehnične podrobnosti: glej »Metoda /api/v1/Documents/Permissions«

# Validacija

Naložene knjige računov servis validira na sintaktične in vsebinske napake. Knjige, za katere validacija ne najde kritičnih napak, gredo v status *ValidatedOk* in čakajo na prenos v zaledje. Knjige, ki vsebujejo kritične napake, gredo v status *ValidationError* in se posledično ne prenesejo v zaledje; uporabnik mora naložiti novo knjigo z odpravljenimi napakami.

Po opravljeni validaciji naložene knjige lahko uporabnik pridobi rezultat validacije s klicem metode /api/v1/InvoiceBook/Status/{edpId}) na REST vmesniku (glej Metoda /api/v1/InvoiceBook/Status/{edpId}) oz. metode GetInvoiceBookStatus na SOAP vmesniku (glej Metoda GetInvoiceBookStatus).

Sintaksa seznama napak v formatu JSON:

[ValidationError{

|  |  |
| --- | --- |
| type | ValidationErrorType string  Info  Warning  Error  Enum: Array [ 3 ]  ***Tip napake. Napake tipa Error se štejejo kot kritične.*** |
| code | ValidationErrorMessaage string  Enum: Array [ 48 ]  ***Koda napake v sistemu eDavki.*** |
| message | string nullable: true  ***Opis napake.*** |
| bookId | integer($int32) nullable: true  ***Zaporedna številka vnosa v listi računov, ki vsebuje napako. Ni definirano za napake v glavi.*** |
| bookArrayIndex | integer($int32) nullable: true  ***Zaporedna številka polja v vnosu, ki vsebuje napako, če jo je moč določiti.*** |

}]

# 

# Priloge

XML shema knjig računov:



JSON shema knjig računov:



Specifikacija CSV formata knjig:



OpenApi (OAS 3.0) datoteka REST spletnega servisa:



Enkratna prijava s protokolom oauth



WSDL SOAP spletnega servisa za prijavo



WSDL SOAP spletnega servisa za funkcionalnosti



Popis polj:



Pravopis pravil:



Vse priloge so na voljo tudi v zip datoteki, objavljeni na naslovu: <https://edavki.durs.si/OpenPortal/Dokumenti/DDV_KIR_KPR.zip>

1. Pomembna razlika je v serializaciji enum vrednosti. V REST servisu so serializirane kot tip integer, v SOAP servisu pa kot tip string. [↑](#footnote-ref-1)