

## SERVIÇO *SPFSERVICE*



### Specification

18-10-2018

#### Controlo de versões

<i>Versão</i>	<i>Data</i>	<i>Autor</i>	<i>Descrição</i>
1.0	13/09/2018	EF	Initial release1
1.1	18/10/2018	EF	Remove redundant APIKEY on authentication; Entering entity identifier in query request

## ÍNDICE

<b>1</b>	<b>OBJECTIVO</b>	<b>3</b>
<b>2</b>	<b>DESCRIÇÃO</b>	<b>4</b>
2.1	METHODS	4
2.1.1	<i>importPayment – Upload a new payment</i>	
2.1.2	<i>getPayment – Consult existing payment</i>	5
2.2	ENTITIES	6
2.2.1	<i>Entity Payment</i>	6
2.3	CODES	7
2.3.1	<i>Response Codes</i>	7
<b>3</b>	<b>WSDL</b>	<b>8</b>
<b>4</b>	<b>examples</b>	<b>12</b>
4.1	REQUEST TO LOAD A NEW PAYMENT	12
4.2	REQUEST TO CONSULT EXISTING PAYMENT	13
4.3	SUCCESS RESPONSE	14
4.4	ERROR RESPONSE	15



## 1 OBJECTIVE

*This document aims to present the definition of the service that allows the loading and consultation of invoices in the SPF - Invoice Payment Service on the EconómicoNet platform..*



## 2 DESCRIPTION

### 2.1 METHODS

The service consists of the following methods::

#### 2.1.1 IMPORTPAYMENT – LOAD A NEW PAYMENT

Create a new payment in SPF

INPUT - *importPaymentRequest*

Argument	Type	Required	description
authToken	Text	Yes	Entity Authentication Token
entityId	Text	Yes	Entity Identifier
productNumber	Numeric	Yes	Product code
sourceId	Text	Yes	Source identifier
reference	Text	No	Reference for ATM Automatic teller machine (EMIS)
amount	Numeric	No	Amount
startDate	Date	No	Start Date (only used on invoice products)
endDate	Date	No	End Date
taxRate	Numeric	No	Tax Rate
clientName	Text	No	Client name
clientAddress	Text	No	Customer Address
clientVatNumber	Text	No	Customer Tax Number
clientEmail	Text	No	Client Email
clientPhoneNumber	Text	No	Customer Phone

OUTPUT - *importPaymentResponse*

Argumento	Type	Required	description
resultCode	Text	Yes	Response code (see table 2.3.1)
reason	Text	No	Reason for error (informational and debug support only)
payment	<i>Payment</i>	No	Payment Information

## 2.1.2 GETPAYMENT – CONSULT EXISTING PAYMENT

Queries an existing payment in SPF INPUT - *getPaymentRequest*

Argument	Type	Required	description
authToken	Text	Sim	Entity Authentication Token
entityId	Text	Sim	Entity Identifier
paymentIds	Lista	Não	Payment Identifiers
sourceIds	Lista	Não	Identifiers at Source

OUTPUT - *getPaymentResponse*

Argument	Type	Required	Description
resultCode	Text	Yes	Response code (see table 2.3.1)
reason	Text	No	Reason for error (informational and debug support only)
payments	Payment List	No	Payment Information

## 2.2 ENTIDADES

### 2.2.1 ENTIDADE PAYMENT

Field	Type	Description
paymentId	Text	Payment identifier created
sourceId	Text	Source identifier
entityId	Text	Entity Identifier
reference	Text	Reference for ATM Automatic teller machine (EMIS)
amount	Numeric	Amount
startDate	Date	Start Date
endDate	Date	End date
status	'ACTIVE', 'INACTIVE', 'CANCELED', 'PAID', 'ERROR'	Payment Status

## 2.3 CODES

### 2.3.1 ANSWER CODES

Code	Description
OK	Success
AUTH_FAILED	Authentication failed (invalid API key or token)
DUPLICATED	Attempting to create a duplicate business object (from: payment)
NOT_FOUND	Attempt to access a non-existent business object (e.g., payment)
SERVICE_UNAVAILABLE	Service temporarily <b>unavailable</b>
GENERIC_ERROR	unexpected error

### 3 WSDL

```
<?xml version="1.0"?>
<wsdl:definitions targetNamespace="http://bancoeconomico.ao/economiconet/spfService"
  xmlns:sch="http://bancoeconomico.ao/economiconet/spfService"
  xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/"
  xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
  xmlns:tns="http://bancoeconomico.ao/economiconet/spfService">
  <wsdl:types>
    <schema attributeFormDefault="unqualified" elementFormDefault="unqualified" targetNamespace="http://bancoeconomico.ao/economiconet/spfService"
      xmlns:msg="http://bancoeconomico.ao/economiconet/message"
      xmlns="http://www.w3.org/2001/XMLSchema">
      <element name="importPaymentRequest">
        <complexType>
          <sequence>
            <element name="authToken" type="string"/>
            <element name="entityId" type="string">
              <annotation>
                <documentation>Identificador da entidade</documentation>
              </annotation>
            </element>
            <element name="productNumber" type="int">
              <annotation>
                <documentation>Código do produto</documentation>
              </annotation>
            </element>
            <element name="sourceId" type="string">
              <annotation>
                <documentation>Identificador na origem</documentation>
              </annotation>
            </element>
            <element minOccurs="0" name="reference" type="string">
              <annotation>
                <documentation>Referência para Multicaixa / EMIS</documentation>
              </annotation>
            </element>
            <element minOccurs="0" name="amount" type="decimal">
              <annotation>
                <documentation>Montante</documentation>
              </annotation>
            </element>
            <element minOccurs="0" name="startDate" type="date">
              <annotation>
                <documentation>Data de início (apenas usado em produtos de tipo factura)</documentation>
              </annotation>
            </element>
            <element minOccurs="0" name="endDate" type="date">
              <annotation>
                <documentation>Data de fim</documentation>
              </annotation>
            </element>
            <element minOccurs="0" name="taxRate" type="decimal">
              <annotation>
                <documentation>Taxa Imposto</documentation>
              </annotation>
            </element>
            <element minOccurs="0" name="clientName" type="string">
              <annotation>
                <documentation>Nome do cliente</documentation>
              </annotation>
            </element>
            <element minOccurs="0" name="clientAddress" type="string">
              <annotation>
                <documentation>Morada do cliente</documentation>
              </annotation>
            </element>
            <element minOccurs="0" name="clientVatNumber" type="string">

```



```
<annotation>
  <documentation>Contribuinte do cliente</documentation>
</annotation>
</element>
<element minOccurs="0" name="clientEmail" type="string">
  <annotation>
    <documentation>Email do cliente</documentation>
  </annotation>
</element>
<element minOccurs="0" name="clientPhoneNumber" type="string">
  <annotation>
    <documentation>Telefone do cliente</documentation>
  </annotation>
</element>
</sequence>
</complexType>
</element>
<element name="importPaymentResponse">
  <complexType>
    <sequence>
      <element name="resultCode" type="string"/>
      <element minOccurs="0" name="reason" type="string"/>
      <element minOccurs="0" name="payment" type="tns:Payment">
        <annotation>
          <documentation>Informação do pagamento</documentation>
        </annotation>
      </element>
    </sequence>
  </complexType>
</element>
<element name="getPaymentRequest">
  <complexType>
    <sequence>
      <element name="authToken" type="string"/>
      <element name="entityId" type="string">
        <annotation>
          <documentation>Identificador da entidade</documentation>
        </annotation>
      </element>
      <element maxOccurs="unbounded" minOccurs="0" name="paymentIds" type="string">
        <annotation>
          <documentation>Identificador(es) de pagamento</documentation>
        </annotation>
      </element>
      <element maxOccurs="unbounded" minOccurs="0" name="sourceIds" type="string">
        <annotation>
          <documentation>Identificador(es) na origem</documentation>
        </annotation>
      </element>
    </sequence>
  </complexType>
</element>
<element name="getPaymentResponse">
  <complexType>
    <sequence>
      <element name="resultCode" type="string"/>
      <element minOccurs="0" name="reason" type="string"/>
      <element maxOccurs="unbounded" minOccurs="0" name="payments" type="tns:Payment">
        <annotation>
          <documentation>Informação de pagamentos</documentation>
        </annotation>
      </element>
    </sequence>
  </complexType>
</element>
<complexType name="Payment">
```

```
<annotation>
  <documentation>Informação de pagamento</documentation>
</annotation>
<sequence>
  <element name="paymentId" type="string">
    <annotation>
      <documentation>Identificador do pagamento criado</documentation>
    </annotation>
  </element>
  <element name="sourceId" type="string">
    <annotation>
      <documentation>Identificador na origem</documentation>
    </annotation>
  </element>
  <element name="entityId" type="string">
    <annotation>
      <documentation>Identificador da entidade</documentation>
    </annotation>
  </element>
  <element minOccurs="0" name="reference" type="string">
    <annotation>
      <documentation>Referência para Multicaixa / EMIS</documentation>
    </annotation>
  </element>
  <element minOccurs="0" name="amount" type="decimal">
    <annotation>
      <documentation>Montante</documentation>
    </annotation>
  </element>
  <element minOccurs="0" name="startDate" type="date">
    <annotation>
      <documentation>Data de início</documentation>
    </annotation>
  </element>
  <element minOccurs="0" name="endDate" type="date">
    <annotation>
      <documentation>Data de fim</documentation>
    </annotation>
  </element>
  <element name="status" type="tns:PaymentStatus"/>
</sequence>
</complexType>
<simpleType name="PaymentStatus">
  <restriction base="string">
    <enumeration value="ACTIVE"/>
    <enumeration value="INACTIVE"/>
    <enumeration value="CANCELED"/>
    <enumeration value="PAID"/>
    <enumeration value="ERROR"/>
  </restriction>
</simpleType>
</schema>
</wsdl:types>
<wsdl:message name="getPaymentRequest">
  <wsdl:part element="tns:getPaymentRequest" name="getPaymentRequest"/>
</wsdl:message>
<wsdl:message name="getPaymentResponse">
  <wsdl:part element="tns:getPaymentResponse" name="getPaymentResponse"/>
</wsdl:message>
<wsdl:message name="importPaymentResponse">
  <wsdl:part element="tns:importPaymentResponse" name="importPaymentResponse"/>
</wsdl:message>
<wsdl:message name="importPaymentRequest">
  <wsdl:part element="tns:importPaymentRequest" name="importPaymentRequest"/>
</wsdl:message>
<wsdl:portType name="SPFService">
```

```
<wsdl:operation name="getPayment">
  <wsdl:input_message="ins:getPaymentRequest" name="getPaymentRequest"/>
  <wsdl:output_message="ins:getPaymentResponse" name="getPaymentResponse"/>
</wsdl:operation>
<wsdl:operation name="importPayment">
  <wsdl:input_message="ins:importPaymentRequest" name="importPaymentRequest"/>
  <wsdl:output_message="ins:importPaymentResponse" name="importPaymentResponse"/>
</wsdl:operation>
</wsdl:portType>
<wsdl:binding name="SPFServiceSoap11" type="ins:SPFService">
  <soap:binding style="document" transport="http://schemas.xmlsoap.org/soap/http"/>
  <wsdl:operation name="getPayment">
    <soap:operation soapAction=""/>
    <wsdl:input name="getPaymentRequest">
      <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output name="getPaymentResponse">
      <soap:body use="literal"/>
    </wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="importPayment">
    <soap:operation soapAction=""/>
    <wsdl:input name="importPaymentRequest">
      <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output name="importPaymentResponse">
      <soap:body use="literal"/>
    </wsdl:output>
  </wsdl:operation>
</wsdl:binding>
<wsdl:service name="SPFServiceService">
  <wsdl:port binding="ins:SPFServiceSoap11" name="SPFServiceSoap11">
    <soap:address location="http://localhost:7080/besa-pe-services-external-webapp/ws/SPFService"/>
  </wsdl:port>
</wsdl:service>
</wsdl:definitions>
```

## 4 EXAMPLES

### 4.1 REQUEST TO LOAD A NEW PAYMENT

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:spf="http://bancoeconomico.ao/economiconet/spfService">
  <soapenv:Header/>
  <soapenv:Body>
    <spf:importPaymentRequest>
      <authToken>YYYYYYY</authToken>
      <entityId>12345</entityId>
      <productNumber>1</productNumber>
      <sourceId>A12345</sourceId>
      <amount>1234.56</amount>
    </spf:importPaymentRequest>
  </soapenv:Body>
</soapenv:Envelope>
```



## 4.2 REQUEST TO CONSULT EXISTING PAYMENT

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:spf="http://bancoeconomico.ao/economiconet/spfService">
  <soapenv:Header/>
  <soapenv:Body>
    <spf:getPaymentRequest>
      <authToken>YYYYYYY</authToken>
      <entityId>12345</entityId>
      <paymentIds>12345</paymentIds>
    </spf:getPaymentRequest>
  </soapenv:Body>
</soapenv:Envelope>
```



## 4.3 SUCCESS RESPONSE

```
<SOAP-ENV:Envelope xmlns:SOAP-
ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header/>
  <SOAP-ENV:Body>
    <ns4:importPaymentResponse
xmlns:ns4="http://bancoeconomico.ao/economiconet/spfService">
      <resultCode>OK</resultCode>
      <payment>
        <paymentId>12345</paymentId>
        <sourceId>A12345</sourceId>
        <entityId>12345</entityId>
        <amount>1234.56</amount>
        <status>ACTIVE</status>
      </payment>
    </ns4:importPaymentResponse>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```



## 4.4 ERROR RESPONSE

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:ret="http://bancoeconomico.ao/economiconet/spfService">
  <soapenv:Header/>
  <soapenv:Body>
    <ret:getPaymentResponse>
      <resultCode>NOT_FOUND</resultCode>
      <reason>Payment not found</reason>
    </ret:getPaymentResponse>
  </soapenv:Body>
</soapenv:Envelope>
```

