

SPORTident Timing SOAP API Developer Documentation

SPORTident GmbH
Arnstadt, Germany
May 5, 2017

Contents

Contents.....	2
Introduction.....	3
Available functions.....	5
Error codes.....	7
Additional error codes for InsertStamp and InsertStampList.....	7
SelectStampListFromEvent(eventId, offset, count).....	8
SelectStampListFromEventBtwTime(eventId, offset, count, startDate, endDate).....	8
SelectStampListByControlBtwTime (eventId, controlNo, offset, count, startDate, endDate).....	10
SelectStampListByCardId (eventId, cardId, offset, count).....	12
SelectStampListByCardIdBtwTime (eventId, cardId, offset, count, startDate, endDate).....	12
SelectStampsCount (eventId).....	14
SelectStampsCountBtwTime (eventId, startDate, endDate).....	14
SelectStampsCountControl (eventId, controlNo).....	16
SelectStampsCountControlBtwTime (eventId, controlNo, startDate, endDate).....	16
SelectStampsCountCardId (eventId, cardId).....	18
SelectStampsCountCardIdBtwTime (eventId, cardId, startDate, endDate).....	18
Class stampObj.....	20
Class result.....	22
InsertStamp (stampObj).....	24
InsertStampList (stampObj).....	26
Revision History.....	29

Introduction

The SPORTident Timing SOAP service provides direct access to raw result data that has been processed by SPORTident Timing. You can use the given result data to evaluate your event results and create reports.

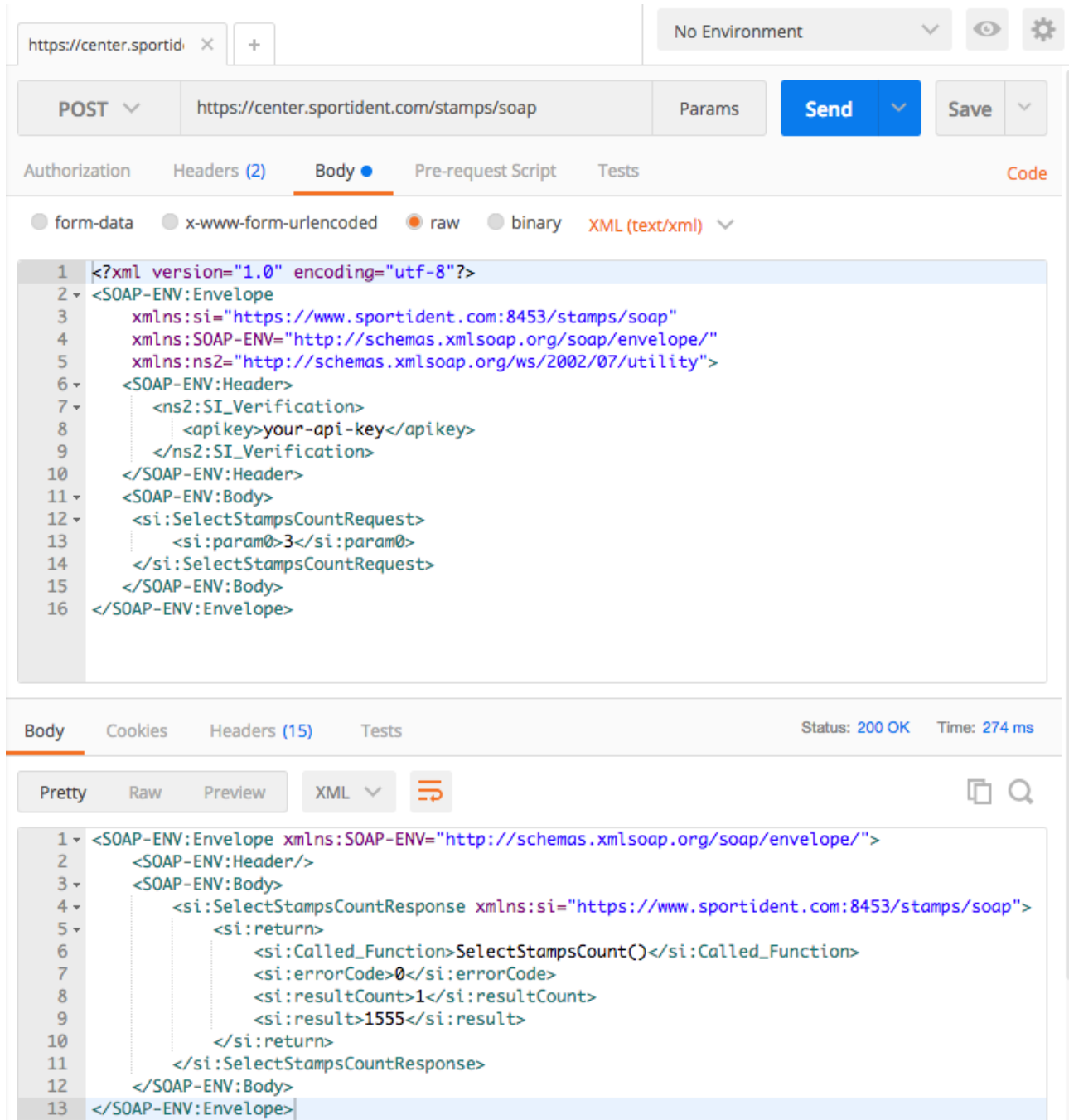
The API is composed of a set of functions which are summarized in section [Available functions](#) and then described in more detail in the rest of the document. As explained in more detail below, the client program sends a XML HTTP request to initiate a function call and receives a XML document in response. The client can also use a WSDL (Web Services Description Language) document which is a XML-based language that provides a model for describing Web services. The WSDL enables clients to automatically discover which functions are available.

After sending a request, the client program should check the response's HTTP status code. Any preliminary problems with the request, such as invalid XML markup, for example, are indicated by the status code. It is therefore recommended to always check for a '200 OK' status code in the response. All functions of the API send an `errorCode` field as part of their response which should be checked by the client as well. The possible SOAP error codes are listed in section [Error codes](#).

Before you can use this service you need an account for SPORTident Center. Please contact SPORTident at timing@sportident.com to request a SPORTident Center account. With SPORTident Center you can manage events, monitor stamps and access and reset your API key. The API key is used for authentication with the SOAP API.

The SOAP server is available at the following location (a SSL secured connection is required): <https://center.sportident.com/stamps/soap>. The location of the WSDL document is: <https://center.sportident.com/stamps/soap/stamps.wsdl>. Please note that the SOAP namespace "si" is "https://www.sportident.com:8453/stamps/soap" and is distinct and should not be confused with the URL at which the SOAP service is available.

One of the easiest ways to get started with the SOAP API is to use a tool such as the [Postman Chrome App](#). With Postman you can edit and send SOAP requests *by hand* and then inspect the response, as the screenshot on the following page shows.



The screenshot displays a REST client interface with a tab for `https://center.sportid`. The environment is set to `No Environment`. The request method is `POST` to `https://center.sportident.com/stamps/soap`. The request body is in `XML (text/xml)` format and contains the following SOAP message:

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <SOAP-ENV:Envelope
3   xmlns:si="https://www.sportident.com:8453/stamps/soap"
4   xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
5   xmlns:ns2="http://schemas.xmlsoap.org/ws/2002/07/utility">
6   <SOAP-ENV:Header>
7     <ns2:SI_Verification>
8       <apikey>your-api-key</apikey>
9     </ns2:SI_Verification>
10  </SOAP-ENV:Header>
11  <SOAP-ENV:Body>
12    <si:SelectStampsCountRequest>
13      <si:param0>3</si:param0>
14    </si:SelectStampsCountRequest>
15  </SOAP-ENV:Body>
16 </SOAP-ENV:Envelope>
```

The response status is `200 OK` with a time of `274 ms`. The response body is shown in `XML` format and contains the following SOAP message:

```
1 <SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
2   <SOAP-ENV:Header/>
3   <SOAP-ENV:Body>
4     <si:SelectStampsCountResponse xmlns:si="https://www.sportident.com:8453/stamps/soap">
5       <si:return>
6         <si:Called_Function>SelectStampsCount()</si:Called_Function>
7         <si:errorCode>0</si:errorCode>
8         <si:resultCount>1</si:resultCount>
9         <si:result>1555</si:result>
10      </si:return>
11    </si:SelectStampsCountResponse>
12  </SOAP-ENV:Body>
13 </SOAP-ENV:Envelope>
```

Available functions

Name	Description
SelectStampListFromEvent (eventId, offset, count)	Returns a list of stamps, filtered by the given event id, returning count items starting at offset.
SelectStampListByControl (eventId, controlNo, offset, count)	Returns a list of stamps, filtered by the given event and control number, returning count items starting at offset.
SelectStampListByCardId (eventId, cardId, offset, count)	Returns a list of stamps, filtered by the given event and cardId, returning count items starting at offset.
SelectStampListFromEventBtwTime (eventId, offset, count, startDate, endDate)	Returns a list of stamps, filtered by the given event id, returning count items starting at offset between startDate and endDate.
SelectStampListByControlBtwTime (eventId, controlNo, offset, count, startDate, endDate)	Returns a list of stamps, filtered by the given event and control number, returning count items starting at offset between startDate and endDate.
SelectStampListByCardIdBtwTime (eventId, cardId, offset, count, startDate, endDate)	Returns a list of stamps, filtered by the given event and cardId, returning count items starting at offset between startDate and endDate.
SelectStampsCount (eventId)	Returns the count of records being available for the given event id.
SelectStampsCountControl (eventId, controlNo)	Returns the count of records being available for the given event id and control number.
SelectStampsCountCardId (eventId, controlNo)	Returns the count of records being available for the given event id and cardId
SelectStampsCountBtwTime (eventId, startDate, endDate)	Returns the count of records being available for the given event id.
SelectStampsCountControlBtwTime (eventId, controlNo, startDate, endDate)	Returns the count of records being available for the given event id and control number.
SelectStampsCountCardIdBtwTime (eventId, cardId, startDate, endDate)	Returns the count of records being available for the given event id and cardId.
SelectEventIdByName (eventName)	Returns the event id for the given event name.

SelectEventNameById
(eventId)

Return the event name for the given event id.

InsertStamp
(stampObj)

Inserts a SPORTident stamp object into the database.

InsertStampList
(array(stampObj))

Inserts multiple SPORTident stamp objects into the database.

The startDate and endDate of all BtwTime functions refer to the stamp punch time, not to the stamp readout time (which is the date and time when the stamps arrived on the server).

Error codes

Value	Name	Description
0	ERROR_SUCCESS	success, no error
2	ERROR_INVALID_PARAMETER	One of the given parameters is invalid.
4	ERROR_INVALID_EVENTID	The given event id is not valid or existing.
6	ERROR_OFFSET_OUT_OF_RANGE	Offset is larger than the count of items available.
8	ERROR_COUNT_OUT_OF_RANGE	Count is larger than the maximum possible value of 1000.
10	ERROR_NOT_ENOUGH_RIGHTS	No read/write rights for this event
12	ERROR_UNKNOWN	Unknown error occurred

Additional error codes for InsertStamp and InsertStampList

Value	Name	Description
14	ERROR_INVALID_CARDID	The given card id has a invalid format.
15	ERROR_INVALID_CONTROL_CODE	The given control code has a invalid format.
16	ERROR_INVALID_STAMP_TYPE	The given stamp type has a invalid format.
17	ERROR_INVALID_READOUT_DATETIME	The given readout date time has a invalid format.
18	ERROR_INVALID_PUNCH_DATETIME	The given punch date time has a invalid format.
19	ERROR_DUPLICATE_PUNCH	The given punch is present in Database.

SelectStampListFromEvent(eventId, offset, count)

SelectStampListFromEventBtwTime(eventId, offset, count, startDate, endDate)

Parameters

Description	Type	Notes
Param 0 (eventId)	int	The id of the SPORTident event
Param 1 (offsetId)	int	Defines at what offset the data is to be read
Param 2 (count)	int	The number of stamps to be read, maximum value is 1000
Param3 (startDate)	string	The date and time of first Stamp to be read
Param4 (endDate)	string	The date and time of last Stamp to be read

Returns

The call returns an object instance of SelectStampListFromEventResponse.

Description	Type	Result
Called_Function	string	SelectStampListFromEvent()
resultCount	int	The number of items returned. This can be less than the number of items requested. - or - A negative number indicating an error (see errorCode)
errorCode	int	ERROR_SUCCESS ERROR_INVALID_PARAMETER ERROR_INVALID_EVENTID ERROR_OFFSET_OUT_OF_RANGE ERROR_COUT_OUT_OF_RANGE ERROR_NOT_ENOUGH_RIGHTS
result	stampObj[]	A node list containing the result items

Request to Server

```
<?xml version="1.0" encoding="utf-8"?>
<SOAP-ENV:Envelope
  xmlns:si="https://www.sportident.com:8453/stamps/soap"
  xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:ns2="http://schemas.xmlsoap.org/ws/2002/07/utility">
  <SOAP-ENV:Header>
    <ns2:SI_Verification>
      <apikey>your_key</apikey>
    </ns2:SI_Verification>
  </SOAP-ENV:Header>
  <SOAP-ENV:Body>
    <si>SelectStampListFromEvent[BtwTime]Request>
      <si:param0>20150301246116</si:param0>
      <si:param1>0</si:param1>
      <si:param2>10</si:param2>
      <si:param3>2015-07-02 00:00:00</si:param3>
      <si:param4>2015-07-02 23:59:59</si:param4>
    </si>SelectStampListFromEvent[BtwTime]Request>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Response from Server

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header/>
  <SOAP-ENV:Body>
    <si>SelectStampListFromEventResponse
      xmlns:si="https://www.sportident.com:8453/stamps/soap">
      <si:return>
        <si:Called_Function>SelectStampListFromEvent()</si:Called_Function>
        <si:errorCode>0</si:errorCode>
        <si:resultCount>1</si:resultCount>
        <si:result>
          <si:item>
            <si:id_stamp>58722</si:id_stamp>
            <si:id_event>20150301246116</si:id_event>
            <si:stamp_card_id>253092</si:stamp_card_id>
            <si:stamp_control_code>3</si:stamp_control_code>
            <si:stamp_control_mode>0</si:stamp_control_mode>
            <si:stamp_station_serial>0</si:stamp_station_serial>
            <si:stamp_pointer>256</si:stamp_pointer>
            <si:stamp_type>1</si:stamp_type>
            <si:stamp_readout_datetime>2015-07-02 15:53:28</si:stamp_readout_datetime>
            <si:stamp_punch_timesi>104038911</si:stamp_punch_timesi>
            <si:stamp_punch_datetime>2015-07-02 16:53:20</si:stamp_punch_datetime>
            <si:stamp_punch_ms>511</si:stamp_punch_ms>
            <si:stamp_punch_wday>4</si:stamp_punch_wday>
            <si:last_modification>2015-07-02 15:53:28</si:last_modification>
            <si:id_pk>63449</si:id_pk>
          </si:item>
        </si:result>
      </si:return>
    </si>SelectStampListFromEventResponse>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

SelectStampListByControl (eventId, controlId, offset, count)

SelectStampListByControlBtwTime (eventId, controlId, offset, count, startDate, endDate)

Parameters

Description	Type	Notes
Param 0 (eventId)	int	The id of the SPORTident event
Param 1 (cardId)	int	The code number of the control point
Param 2 (offsetId)	int	Defines at what offset the data is to be read
Param 3 (count)	int	The number of stamps to be read, maximum value is 1000
Param4 (startDate)	string	The date and time of first Stamp to be read
Param5 (endDate)	string	The date and time of last Stamp to be read

Returns

The call returns an object instance of SelectStampListByControlResponse.

Description	Type	Result
Called_Function	string	SelectStampListByControl()
resultCount	int	The number of items returned. This can be less than the number of items requested. - or - A negative number indicating an error (see errorCode)
errorCode	int	ERROR_SUCCESS ERROR_INVALID_PARAMETER ERROR_INVALID_EVENTID ERROR_OFFSET_OUT_OF_RANGE ERROR_COUT_OUT_OF_RANGE ERROR_NOT_ENOUGH_RIGHTS
result	stampObj[]	A node list containing the result items.

Request to Server

```
<?xml version="1.0" encoding="utf-8"?>
<SOAP-ENV:Envelope
  xmlns:si="https://www.sportident.com:8453/stamps/soap"
  xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:ns2="http://schemas.xmlsoap.org/ws/2002/07/utility">
  <SOAP-ENV:Header>
    <ns2:SI_Verification>
      <apikey>your_key</apikey>
    </ns2:SI_Verification>
  </SOAP-ENV:Header>
  <SOAP-ENV:Body>
    <si:SelectStampListByControl[BtwTime]Request>
      <si:param0>20150301246116</si:param0>
      <si:param1>4</si:param1>
      <si:param2>0</si:param2>
      <si:param3>1</si:param3>
      <si:param3>2015-07-02 00:00:00</si:param3>
      <si:param4>2015-07-02 23:59:59</si:param4>
    </si:SelectStampListByControl[BtwTime]Request>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Response from Server

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header/>
  <SOAP-ENV:Body>
    <si:SelectStampListByControlResponse
      xmlns:si="https://www.sportident.com:8453/stamps/soap">
      <si:return>
        <si:Called_Function>SelectStampListByControl()</si:Called_Function>
        <si:errorCode>0</si:errorCode>
        <si:resultCount>1</si:resultCount>
        <si:result>
          <si:item>
            <si:id_stamp>58741</si:id_stamp>
            <si:id_event>20150301246116</si:id_event>
            <si:stamp_card_id>1394200</si:stamp_card_id>
            <si:stamp_control_code>4</si:stamp_control_code>
            <si:stamp_control_mode>0</si:stamp_control_mode>
            <si:stamp_station_serial>0</si:stamp_station_serial>
            <si:stamp_pointer>264</si:stamp_pointer>
            <si:stamp_type>1</si:stamp_type>
            <si:stamp_readout_datetime>2015-07-02 16:13:47</si:stamp_readout_datetime>
            <si:stamp_punch_timesi>104352114</si:stamp_punch_timesi>
            <si:stamp_punch_datetime>2015-07-02 17:13:42</si:stamp_punch_datetime>
            <si:stamp_punch_ms>882</si:stamp_punch_ms>
            <si:stamp_punch_wday>4</si:stamp_punch_wday>
            <si:last_modification>2015-07-02 16:13:47</si:last_modification>
            <si:id_pk>63470</si:id_pk>
          </si:item>
        </si:result>
      </si:return>
    </si:SelectStampListByControlResponse>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

SelectStampListByCardId (eventId, cardId, offset, count)

SelectStampListByCardIdBtwTime (eventId, cardId, offset, count, startDate, endDate)

Parameters

Description	Type	Notes
Param 0 (eventId)	int	The id of the SPORTident event
Param 1 (cardId)	int	The SiCard number
Param 2 (offsetId)	int	Defines at what offset the data is to be read
Param 3 (count)	int	The number of stamps to be read, maximum value is 1000
Param4 (startDate)	string	The date and time of first Stamp to be read
Param5 (endDate)	string	The date and time of last Stamp to be read

Returns

The call returns an object instance of SelectStampListByCardIdResponse.

Description	Type	Result
Called_Function	string	SelectStampListByCardId()
resultCount	int	The number of items returned. This can be less than the number of items requested. - or - A negative number indicating an error (see errorCode)
errorCode	int	ERROR_SUCCESS ERROR_INVALID_PARAMETER ERROR_INVALID_EVENTID ERROR_OFFSET_OUT_OF_RANGE ERROR_COUT_OUT_OF_RANGE ERROR_NOT_ENOUGH_RIGHTS
result	stampObj[]	A node list containing the result items

Request to Server

```
<?xml version="1.0" encoding="utf-8"?>
<SOAP-ENV:Envelope
  xmlns:si="https://www.sportident.com:8453/stamps/soap"
  xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:ns2="http://schemas.xmlsoap.org/ws/2002/07/utility">
  <SOAP-ENV:Header>
    <ns2:SI_Verification>
      <apikey>your_key</apikey>
    </ns2:SI_Verification>
  </SOAP-ENV:Header>
  <SOAP-ENV:Body>
    <si:SelectStampListByCardId[BtwTime]Request>
      <si:param0>20150301246116</si:param0>
      <si:param1>1470316</si:param1>
      <si:param2>0</si:param2>
      <si:param3>1</si:param3>
      [ <si:param4>2015-07-02 00:00:00</si:param4> ]
      [ <si:param5>2015-07-02 23:59:59</si:param5> ]
    </si:SelectStampListByCardId[BtwTime]Request>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Response from Server

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header/>
  <SOAP-ENV:Body>
    <si:SelectStampListByCardIdResponse
      xmlns:si="https://www.sportident.com:8453/stamps/soap">
      <si:return>
        <si:Called_Function>SelectStampListByCardId()</si:Called_Function>
        <si:errorCode>0</si:errorCode>
        <si:resultCount>1</si:resultCount>
        <si:result>
          <si:item>
            <si:id_stamp>58729</si:id_stamp>
            <si:id_event>20150301246116</si:id_event>
            <si:stamp_card_id>1470316</si:stamp_card_id>
            <si:stamp_control_code>3</si:stamp_control_code>
            <si:stamp_control_mode>0</si:stamp_control_mode>
            <si:stamp_station_serial>0</si:stamp_station_serial>
            <si:stamp_pointer>312</si:stamp_pointer>
            <si:stamp_type>1</si:stamp_type>
            <si:stamp_readout_datetime>2015-07-02 16:06:03</si:stamp_readout_datetime>
            <si:stamp_punch_timesi>104232715</si:stamp_punch_timesi>
            <si:stamp_punch_datetime>2015-07-02 17:05:59</si:stamp_punch_datetime>
            <si:stamp_punch_ms>11</si:stamp_punch_ms>
            <si:stamp_punch_wday>4</si:stamp_punch_wday>
            <si:last_modification>2015-07-02 16:06:03</si:last_modification>
            <si:id_pk>63458</si:id_pk>
          </si:item>
        </si:result>
      </si:return>
    </si:SelectStampListByCardIdResponse>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

SelectStampsCount (eventId)

SelectStampsCountBtwTime (eventId, startDate, endDate)

Parameters

Description	Type	Notes
Param 0 (eventId)	int	The id of the event
Param1 (startDate)	string	The date and time of first Stamp to be read
Param2 (endDate)	string	The date and time of last Stamp to be read

Returns

The call returns an object instance of SelectStampsCountResponse.

Description	Type	Result
Called_Function	string	SelectStampsCount()
resultCount	int	The number of results available. In this case this value is always 1
errorCode	int	ERROR_SUCCESS - or - an error number like described above.
result	int	The number of records being available.

Request to Server

```
<?xml version="1.0" encoding="utf-8"?>
<SOAP-ENV:Envelope
  xmlns:si="https://www.sportident.com:8453/stamps/soap"
  xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:ns2="http://schemas.xmlsoap.org/ws/2002/07/utility">
  <SOAP-ENV:Header>
    <ns2:SI_Verification>
      <apikey>your_key</apikey>
    </ns2:SI_Verification>
  </SOAP-ENV:Header>
  <SOAP-ENV:Body>
    <si:SelectStampsCount [BtwTime]Request>
      <si:param0>20150301246116</si:param0>
      [<si:param1>2015-07-02 00:00:00</si:param1>]
      [<si:param2>2015-07-02 23:59:59</si:param2>]
    </si:SelectStampsCount [BtwTime]Request>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Response from Server

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header/>
  <SOAP-ENV:Body>
    <si:SelectStampsCountResponse
  xmlns:si="https://www.sportident.com:8453/stamps/soap">
      <si:return>
        <si:Called_Function>SelectStampsCount(</si:Called_Function>
        <si:errorCode>0</si:errorCode>
        <si:resultCount>1</si:resultCount>
        <si:result>586</si:result>
      </si:return>
    </si:SelectStampsCountResponse>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

SelectStampsCountControl (eventId, controlNo)

SelectStampsCountControlBtwTime (eventId, controlNo, startDate, endDate)

Parameters

Description	Type	Notes
Param 0 (eventId)	int	The id of the SPORTident Event
Param 1 (cardId)	int	The code number of the control point
Param2 (startDate)	string	The date and time of first Stamp to be read
Param3 (endDate)	string	The date and time of last Stamp to be read

Returns

The call returns an object instance of SelectStampsCountControlResponse.

Description	Type	Result
Called_Function	string	SelectStampsCardId()
resultCount	int	The number of results available. In this case this value is always 1.
errorCode	int	ERROR_SUCCESS ERROR_INVALID_PARAMETER
result	int	The number of records being available.

Request to Server

```
<?xml version="1.0" encoding="utf-8"?>
<SOAP-ENV:Envelope
  xmlns:si="https://www.sportident.com:8453/stamps/soap"
  xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:ns2="http://schemas.xmlsoap.org/ws/2002/07/utility">
  <SOAP-ENV:Header>
    <ns2:SI_Verification>
      <apikey>your_key</apikey>
    </ns2:SI_Verification>
  </SOAP-ENV:Header>
  <SOAP-ENV:Body>
    <si:SelectStampsCountControl[BtwTime]Request>
      <si:param0>20150301246116</si:param0>
      <si:param1>3</si:param1>
      [<si:param2>2015-07-02 00:00:00</si:param2>]
      [<si:param3>2015-07-02 23:59:59</si:param3>]
    </si:SelectStampsCountControl[BtwTime]Request>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Response from Server

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header/>
  <SOAP-ENV:Body>
    <si:SelectStampsCountControlResponse
      xmlns:si="https://www.sportident.com:8453/stamps/soap">
      <si:return>
        <si:Called_Function>SelectStampsCountControl()</si:Called_Function>
        <si:errorCode>0</si:errorCode>
        <si:resultCount>1</si:resultCount>
        <si:result>197</si:result>
      </si:return>
    </si:SelectStampsCountControlResponse>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

SelectStampsCountCardId (eventId, cardId)

SelectStampsCountCardIdBtwTime (eventId, cardId, startDate, endDate)

Parameters

Description	Type	Notes
Param 0 (eventId)	int	The id of the SPORTident Event
Param 1 (cardId)	int	The code number of the control point
Param2 (startDate)	string	The date and time of first Stamp to be read
Param3 (endDate)	string	The date and time of last Stamp to be read

Returns

The call returns an object instance of SelectStampsCardIdResponse.

Description	Type	Result
Called_Function	string	SelectStampsCardId()
resultCount	int	The number of results available. In this case this value is always 1
errorCode	int	ERROR_SUCCESS ERROR_INVALID_PARAMETER
result	int	The number of records being available

Request to Server

```
<?xml version="1.0" encoding="utf-8"?>
<SOAP-ENV:Envelope
  xmlns:si="https://www.sportident.com:8453/stamps/soap"
  xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:ns2="http://schemas.xmlsoap.org/ws/2002/07/utility">
  <SOAP-ENV:Header>
    <ns2:SI_Verification>
      <apikey>your_key</apikey>
    </ns2:SI_Verification>
  </SOAP-ENV:Header>
  <SOAP-ENV:Body>
    <si:SelectStampsCountCardId[BtwTime]Request>
      <si:param0>20150301246116</si:param0>
      <si:param1>1470316</si:param1>
      [<si:param1>2015-07-02 00:00:00</si:param1>]
      [<si:param2>2015-07-02 23:59:59</si:param2>]
    </si:SelectStampsCountCardId[BtwTime]Request>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Response from Server

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header/>
  <SOAP-ENV:Body>
    <si:SelectStampsCountCardIdResponse
      xmlns:si="https://www.sportident.com:8453/stamps/soap">
      <si:return>
        <si:Called_Function>SelectStampsCountCardId()</si:Called_Function>
        <si:errorCode>0</si:errorCode>
        <si:resultCount>1</si:resultCount>
        <si:result>3</si:result>
      </si:return>
    </si:SelectStampsCountCardIdResponse>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Class stampObj

The stampObj class is used to transmit data to the SOAP server. It contains raw punch record data as well as interpreted information depending on the event definition.

Parameters

To leave a parameter out, you can supply an empty tag but need to add the attribute `xsi:nil="true"`. See sample below.

Description	Type	Notes
Param0 (id_stamp)	int	The PK from database. Leave this out or set to 0.
Param1 (id_event)	int	The event id
Param2 (stamp_card_id)	int	SPORTident card number (SIID)
Param3 (stamp_control_code)	int	The control code of the station.
Param4 (stamp_control_mode)	int	The operating mode of the control station. (See programmer's guide for values.)
Param5 (stamp_station_serial)	int	The serial number of the station. You can leave this value out or set to 0.
Param6 (stamp_pointer)	int	The pointer of the stamp in the station memory. You can leave this value out or set to 0.
Param7 (stamp_type)	int	The source of the record. Can be one of the following values: READOUT = 0 ONLINE = 1 BACKUP = 2 RADIO_READOUT = 3
Param8 (stamp_readout_datetime)	datetime	The date and time when the record has been read from the station or card. You should use the following standard format: YYYY-MM-DD hh:mm:ss e. g. 2013-11-29 23:57:33
Param9 (stamp_punch_timesi)	int	The punch time in SPORTident time format. You can leave this value out or set to 0.
Param10 (stamp_punch_datetime)	datetime	The date and time of the records creation. Same format like stamp_readout_datetime.
Param11 (stamp_punch_ms)	int	The fractions of a second of the records creation date time.

Param12 (stamp_punch_wday)	int	The day of week of the records creation. Value range: (0-7) 0 - Sunday, 1 - Monday ... 6 - Saturday, 7 - invalid day of week
Param13 (last modification)	datetime	The date and time of the last modification to this instance. You can leave this value out.
Param14 (id_pk)	int	Reserved. Leave this out or set to 0.

Sample

```
<si:item>
  <si:id_stamp />
  <si:id_event>2008122301</si:id_event>
  <si:stamp_card_id>1006005</si:stamp_card_id>
  <si:stamp_control_code>31</si:stamp_control_code>
  <si:stamp_control_mode>2</si:stamp_control_mode>
  <si:stamp_station_serial />
  <si:stamp_pointer />
  <si:stamp_type>2</si:stamp_type>
  <si:stamp_readout_datetime>2013-03-22 15:14:26</si:stamp_readout_datetime>
  <si:stamp_punch_timesi />
  <si:stamp_punch_datetime>2013-03-22 15:14:26</si:stamp_punch_datetime>
  <si:stamp_punch_ms>0</si:stamp_punch_ms>
  <si:stamp_punch_wday>0</si:stamp_punch_wday>
  <si:last_modification />
  <si:id_pk />
</si:item>
```

Class result

The result class is used to respond to insertion requests. It contains information for every single item insertion operation.

Parameters

If a insertion operation has failed, you will find the `errorCode` parameter of the response set to a value unequal to `ERROR_SUCCESS`.

The result instance returned then will be `<si:result />`.

In success cases, `errorCode` will be `ERROR_SUCCESS` and result will contain:

Description	Type	Notes
Param0	array()	An array containing detailed results for every single insertion operation
	string	A hash of the first item that has been inserted (see below for calculation)
	array(int)	An array containing the error codes for every item that has been inserted

Hash code algorithm

The hash code is calculated and returned on insert operation to give you a possibility to identify your insertion package when using asynchronous calls. You can easily compare the hash against the hash you have calculated before sending the request and verify the package identity.

The calculation method works as following:

1. Concat `id_event`, `stamp_card_id`, `stamp_control_code`, `stamp_punch_datetime`
2. Calculate an MD5 hash over the resulting string
3. Take the hex representation of the resulting byte array, leave out any separators and convert it lower case.
4. The hash code should be a string with a length of 32 characters like this:
a849414587a54cddc3bfca55486c2831

Hash code sample

```
var stamps = new stampObj()
{
    id_event = 2008122301,
    stamp_card_id = 1002000,
    stamp_control_code = 31,
    stamp_punch_datetime = "2013-11-28 08:10:14",
};

string stringToHash = id_event +
    stamp_card_id +
    stamp_control_code +
    stamp_punch_datetime;

//stringToHash = "20081223011002000312013-11-28 08:10:14"

string hashCode = MD5.Hash(stringToHash);

//stringToHash = " a849414587a54cddc3bfca55486c2831"
```

InsertStamp (stampObj)

Parameters

Description	Type	Notes
Param 0 (stampObj)	stampObj	An instance of stampObj that contains the data to be inserted into the database

Returns

The call returns an object instance of InsertStampResponse.

Description	Type	Notes
Called_Function	string	InsertStamp()
resultCount	int	The overall number of items inserted into the database. In this case, this can be 0 (in case of an error) or 1 (successful)
errorCode	int	One of the possible error codes described above - or - ERROR_SUCCESS if successful
result	array()	An empty value if any errors occurred - or - An array containing detailed results:
	string	A hash of the inserted item
	array(int)	The error code for every inserted item In this case, the array contains only one item

Request to Server

```
<?xml version="1.0" encoding="utf-8"?>
<SOAP-ENV:Envelope
  xmlns:si="https://www.sportident.com:8453/stamps/soap"
  xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:ns2="http://schemas.xmlsoap.org/ws/2002/07/utility">
  <SOAP-ENV:Header>
    <ns2:SI_Verification>
      <apikey>your_key</apikey>
    </ns2:SI_Verification>
  </SOAP-ENV:Header>
  <SOAP-ENV:Body>
    <si:InsertStampRequest>
      <si:param0>
        <si:id_stamp />
        <si:id_event>2008122301</si:id_event>
        <si:stamp_card_id>1002000</si:stamp_card_id>
        <si:stamp_control_code>31</si:stamp_control_code>
        <si:stamp_control_mode>2</si:stamp_control_mode>
        <si:stamp_station_serial>0</si:stamp_station_serial>
        <si:stamp_pointer>0</si:stamp_pointer>
        <si:stamp_type>0</si:stamp_type>
        <si:stamp_readout_datetime>2013-11-28 11:46:08
        </si:stamp_readout_datetime>
        <si:stamp_punch_timesi>0</si:stamp_punch_timesi>
        <si:stamp_punch_datetime>2013-11-28 08:10:14
        </si:stamp_punch_datetime>
        <si:stamp_punch_ms>142</si:stamp_punch_ms>
        <si:stamp_punch_wday>4</si:stamp_punch_wday>
        <si:last_modification />
        <si:id_pk />
      </si:param0>
    </si:InsertStampRequest>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Response from Server

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header/>
  <SOAP-ENV:Body>
    <si:InsertStampResponse xmlns:si="https://www.sportident.com:8453/stamps/soap">
      <si:return>
        <si:Called_Function>InsertStamp()</si:Called_Function>
        <si:errorCode>0</si:errorCode>
        <si:resultCount>1</si:resultCount>
        <si:result>
          <si:item>a849414587a54cddc3bfca55486c2831</si:item>
          <si:item>
            <si:item>0</si:item>
          </si:item>
        </si:result>
      </si:return>
    </si:InsertStampResponse>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

InsertStampList (stampObj)

Parameters

Description	Type	Notes
Param 0 (stampObj)	stamp[]	An array of stamp objects that contain the data to be inserted into the database

Returns

The call returns an object instance of InsertStampListResponse.

Description	Type	Notes
Called_Function	string	InsertStampList()
resultCount	int	The overall number of items inserted into the database.
errorCode	int	One of the possible error codes described above - or - ERROR_SUCCESS if successful
result	array()	An empty value if any errors occurred - or - An array containing detailed results:
	string	A hash of the first inserted item
	array(int)	The error code for every inserted item In this case, the array contains only one item

Request to Server

```

<?xml version="1.0" encoding="utf-8"?>
<SOAP-ENV:Envelope xmlns:si="https://www.sportident.com:8453/stamps/soap"
  xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:ns2="http://schemas.xmlsoap.org/ws/2002/07/utility">
  <SOAP-ENV:Header>
    <ns2:SI_Verification><apikey>your_key</apikey></ns2:SI_Verification>
  </SOAP-ENV:Header>
  <SOAP-ENV:Body>
    <si:InsertStampListRequest>
      <si:param0>
        <si:item>
          <si:id_stamp />
          <si:id_event>2008122301</si:id_event>
          <si:stamp_card_id>1002000</si:stamp_card_id>
          <si:stamp_control_code>31</si:stamp_control_code>
          <si:stamp_control_mode>2</si:stamp_control_mode>
          <si:stamp_station_serial>0</si:stamp_station_serial>
          <si:stamp_pointer>0</si:stamp_pointer>
          <si:stamp_type>0</si:stamp_type>
          <si:stamp_readout_datetime>2013-11-28 13:53:45</si:stamp_readout_datetime>
          <si:stamp_punch_timesi>0</si:stamp_punch_timesi>
          <si:stamp_punch_datetime>2013-11-28 08:10:14</si:stamp_punch_datetime>
          <si:stamp_punch_ms>142</si:stamp_punch_ms>
          <si:stamp_punch_wday>4</si:stamp_punch_wday>
          <si:last_modification />
        </si:item>
        <si:item>
          <si:id_stamp />
          <si:id_event>2008122301</si:id_event>
          <si:stamp_card_id>1002000</si:stamp_card_id>
          <si:stamp_control_code>32</si:stamp_control_code>
          <si:stamp_control_mode>2</si:stamp_control_mode>
          <si:stamp_station_serial>0</si:stamp_station_serial>
          <si:stamp_pointer>0</si:stamp_pointer>
          <si:stamp_type>0</si:stamp_type>
          <si:stamp_readout_datetime>2013-11-28 13:53:45</si:stamp_readout_datetime>
          <si:stamp_punch_timesi>0</si:stamp_punch_timesi>
          <si:stamp_punch_datetime>2013-11-28 08:34:31</si:stamp_punch_datetime>
          <si:stamp_punch_ms>780</si:stamp_punch_ms>
          <si:stamp_punch_wday>4</si:stamp_punch_wday>
          <si:last_modification />
        </si:item>
        <si:item>
          <si:id_stamp />
          <si:id_event>2008122301</si:id_event>
          <si:stamp_card_id>1002000</si:stamp_card_id>
          <si:stamp_control_code>33</si:stamp_control_code>
          <si:stamp_control_mode>2</si:stamp_control_mode>
          <si:stamp_station_serial>0</si:stamp_station_serial>
          <si:stamp_pointer>0</si:stamp_pointer>
          <si:stamp_type>0</si:stamp_type>
          <si:stamp_readout_datetime>2013-11-28 13:53:45</si:stamp_readout_datetime>
          <si:stamp_punch_timesi>0</si:stamp_punch_timesi>
          <si:stamp_punch_datetime>2013-11-28 09:01:12</si:stamp_punch_datetime>
          <si:stamp_punch_ms>57</si:stamp_punch_ms>
          <si:stamp_punch_wday>4</si:stamp_punch_wday>
          <si:last_modification />
        </si:item>
      </si:param0>
    </si:InsertStampListRequest>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

Response from Server

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header/>
  <SOAP-ENV:Body>
    <si:InsertStampListResponse xmlns:si="https://www.sportident.com:8453/stamps/soap">
      <si:return>
        <si:Called_Function>InsertStampList()
        </si:Called_Function>
        <si:errorCode>0</si:errorCode>
        <si:resultCount>3</si:resultCount>
        <si:result>
          <si:item>a849414587a54cddc3bfca55486c2831</si:item>
          <si:item>
            <si:item>0</si:item>
            <si:item>0</si:item>
            <si:item>0</si:item>
          </si:item>
        </si:result>
      </si:return>
    </si:InsertStampListResponse>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Revision History

August 12, 2015	Improved code examples, formatting, and smaller fixes
July 13, 2015	Updated for SOAP API version 2
November 9, 2015	Updated for SOAP API version 3
May 5, 2017	Updated URL for SOAP API version 3, clarified semantics of startDate and endDate