**SMM WEB SERVICE**

**SmmWs3 API**

*Date Revised: Mon, 12th February 2026*

## TABLE OF CONTENTS

## TABLES INDEX

## DOCUMENT PURPOSE

This document briefly describes how to interact with SMM SmmWs3 API. Also are described the operations (i.e. methods) that are exposed and response messages.

## API DESCRIPTION

The API exposes the following services:

- `Generate Token*`
- `Create Shipment and generate label`
- `Get Label`
- `Delete Label`
- `Manifest`
- `Upload document [Action: UPLOAD]`
- `Delete document [Action: DELETE]`
- `Tracking`

The `Generate Token` service will provide a valid token, that must be provided to all the other services.

All data passed and received with the web-service is based on headers values and JSON strings and every-thing needs to be passed with **[`POST`]** method. To be able to use all the services (except `Generate To-ken`), it's mandatory to add the following headers in the call:

- `Accept: application/json`
- `username: testusername`
- `token: {token}TEST-TOKEN-a0a2-47b0-b1ab-923d2ec4b8ce`

If any of these mandatory headers is missing, the service will return an error response.

## GENERATE TOKEN

This function allows with valid credentials (http basic authorization) to gain a valid token usable with the other operations. To be able to use the web services is mandatory to add some headers in the call; in the generate Token the headers are:

- `Accept: application/json`
- `Authorization: Basic (USERNAME:PASSWORD base 64 encoded)`
- `forcenew: true (default true) If set to true the token will be regenerated, otherwise the service will return the active one, if exists.`

The obtained token is valid for 24 hours.

Each account may have only 1 token valid at the same time; each time a new one gets generated the old one becomes invalid. To avoid any problem with multiples entities using the same service, you can set the "`forcenew`" field to false in the request to avoid the generation of a new token each time.

### 🔗 URL

```
https://ws3.sanmarinomail.it/SmmWs3/token
```

### ⬆ REQUEST EXAMPLE

```
curl -H "Accept: application/json" -H "Authorization: Basic
YWxhZGRpbjpvcGVuc2VzYW1l" -H "forcenew: false" -X POST
https://ws3.sanmarinomail.it/SmmWs3/token
```

### ⬇ RESPONSE EXAMPLES

#### ❌ ERROR:

```
{
   "errorDescription": "Username or Password Not Valid",
   "errorCode": 3001,
   "status": -1
}
```

#### ✅ SUCCESSFUL (STATUS FIELD = 0):

```
{
   "token": "{token}d87bb9c6-25fa-4035-ae90-c515554f3cf2",
   "status": 0
}
```

## CREATE SHIPMENT

This operation allows you to book a shipment and retrieve the corresponding label.

> PLEASE NOTE TO UPLOAD ANY NEEDED FILE BEFORE
> CALLING THE GENERATE LABEL REQUEST
>
> (see specific upload method information).

The shipment references for the label to be retrieved must be provided as a JSON string.

All fields accepted by the request are listed in Table 1 and Table 2 Table 1 and Table 2.

If the `SMM_PRODUCT_CODE` (optional) field is set, it will be used to identify the SMM product to be used; otherwise, the SMM Algorithm will need to determine the product automatically.

For extra-CEE shipments or shipments containing dangerous goods, all fields in the package object are mandatory.

For multi-parcel shipments, that require parcel-specific information (e.g., UPS, DHL), the `MULTI_PARCEL` field must be set to 1, and a separate *Json Array* must be populated with all parcel details (see example below).

The options available (not mandatory) are:

- `force_reprint_if_existsforce_reprint`: (default value: false): if a label (with same reference) already exists and the field is set to false the system will answer with "Reference already used" error otherwise if set to true the system will send the previously generated label.
- `delete_if_exists`: (default value: false): if the label already exists and the field is set to true the existing label will be deleted, and a new one will be generated.
- `base64`: (default value: false): if set to true, the output of ZPL label will be encoded in Base64, otherwise will be sent in plain text.

The optional json object "mirror" will be added to the responce as received

### MIRROR STRUCTURE USAGE

You can use the `mirror` Json structure by filling it with the fields required for recognition. The structure will be returned unchanged in the response.

### 🔗 URL

https://ws3.sanmarinomail.it/SmmWs3/**createshipment**

## 📄 JSON EXAMPLE

```
{
    "shipment": {
        "CLIENT_SHIP_REF":"TEST_NEW_WS_001",
        "SMM_CUST_CODE":"ITIT0000999",
        "DELIVERY_NAME":"Mario Rossi",
        "DELIVERY_ADDR":"via Farini 81",
        "DELIVERY_CITY":"Milano",
        "DELIVERY_ZIP_CODE":"20159",
        "WEIGHT":12,
        "DEL_ISO_COUNTRY_CODE":"IT",
        "PRE_ALLERT_DATE":"2020-07-20",
        "DELIVERY_DISTRICT":"MI"
    },
    "options": {
        "force_reprint":false,
        "confirmed":false,
        "delete_if_exists":false,
        "base64":true
    },
    "mirror": {
        "cod_attivita_infolog":"temp",
        "field1":"temp2"
    }
}
```

## ↑ CURL EXAMPLE

```
curl -H "Accept: application/json" -H "token: {token}TEST-TOKEN-a0a2-47b0-
b1ab-923d2ec4b8ce" -H "username: testusername" -d '{"CLI-
ENT_SHIP_REF":"testreference","SMM_CUST_CODE":"TEST_ID","base64":false}......
.......{  "shipment":{  "CLIENT_SHIP_REF":"TEST_NEW_WS_002",
"SMM_CUST_CODE":"ITIT0000999",  "DELIVERY_NAME":"Mario Rossi",  "DELIV-
ERY_ADDR":"via farini 81",  "DELIVERY_CITY":"Milano",  "DELIV-
ERY_ZIP_CODE":"20159",  "WEIGHT":12,  "DEL_ISO_COUNTRY_CODE":"IT",
"PRE_ALLERT_DATE":"2020-07-13",  "DELIVERY_DISTRICT":"MI"  },  "op-
tions":{  "force_reprint_if_exists":false,  "confirmed":false,  "de-
lete_if_exists":false,  "base64":true  },  "mirror":{  "field1":"temp"  }  }'
-X POST
http://localhost:8080//retrievelabelhttps://ws3.sanmarinomail.it/SmmWs3/creat
eshipment
```

## ↓ RESPONSE EXAMPLES

### ❌ ERROR:

```
{
  "errorDescription": "Token Not Valid",
  "errorCode": 3002,
  "status": -1
}
```

### ❌ ERROR:

```
{
  "mirror": {
        "cod_attivita_infolog":"temp",
        "cod_stazione":"temp2"
```

```
  },
  "CLIENT_SHIP_REF": "TEST_NEW_WS_003",
  "notes": "",
  "errorDescription": "Error Reference number already used",
  "errorCode": 1063,
  "label":
"^XA~TA000~JSN^LT0^MNW^MTD^PON^PMN^LH0,0^JMA^PR4,4~SD15^JUS^LRN^CI0^XZ ^XA
^MMT ^PW799 ^LL1199 ^LS0 ^FT753,776^A0I,49,36^FH\\^FDSAN MARINO MAIL ITALIA
SRL^FS ^FO269,741^GB519,0,8^FS ^FT728,694^A0I,28,28^FH\\^FDATTENTION ^FS
^FT725,645^A0I,28,28^FH\\^FDIF YOU RECEIVE THIS LABEL^FS
^FT725,597^A0I,28,28^FH\\^FDPLEASE CALL +39900266227799^FS
^FT634,553^A0I,28,28^FH\\^FDOPS DEPT^FS ^FT733,498^A0I,28,28^FH\\^FDE-MAIL
OPS@SANMARINOMAIL.IT^FS ^FWB ^BY3,3,160 ^FO55,50^BC^FD#ICR#^FS ^PQ1,0,1,Y
^FWN ^XZ",
  "status": -1
}
```

## ❌ ERROR:

```
{
  "errorDescription": "Error parsing json input - org.json.JSONException: Ex-
pected a ':' after a key at 103 [character 104 line 1]",
  "errorCode": 999,
  "status": -1
}
```

## ✅ SUCCESSFUL (STATUS FIELD = 0):

```
{
  "MASTER_USN_PARCEL_NR": "465162803",
  "mirror": {
      "field1":"temp"
  },
  "CLIENT_SHIP_REF": "TEST_NEW_WS_004",
  "notes": "",
  "Haulier": "GLS",
  "USN_PARCEL_NUMBER": "600256057",
  "label": "^XA ^LH0,0 ^FO4,2 ^GB804,798,4 ^FS ^FO4,2 ^GB548,52,4 ^FS
^FO19,12 ^ADN,36,20 ^FDTEST WEBSERVICE ^FS ^FO548,2 ^GB260,52,4 ^FS ^FO561,16
^ADN,36,10 ^FDP/V ^FS ^FO611,16 ^ADN,36,10 ^FD1:3 ^FS ^FO693,16 ^ADN,36,10
^FD22/07/20 ^FS ^FO21,48 ^ADN,170,60 ^FDBOLLATE ^FS ^FO593,48 ^ADN,170,60
^FD99 ^FS ^FO15,185 ^AQN,10,10 ^FDPer info sul trattamento dati personali
www.gls-italy.com/privacydest ^FS ^FO4,210 ^GB128,166,4 ^FS ^FO128,210
^GB680,166,4 ^FS ^FO136,220 ^ADN,36,15 ^FDMario Rossi ^FS ^FO136,276
^ADN,36,10 ^FDvia farini 81 ^FS ^FO15,243 ^ADN,108,50 ^FDLD ^FS ^FO136,338
^ADN,36,14 ^FDMilano ^FS ^FO580,338 ^ADN,36,15 ^FDMI ^FS ^FO632,210
^GB176,166,4 ^FS ^FO640,220 ^ADN,18,5 ^FDSEDI CONTIGUE ^FS ^FO695,246
^ADN,18,5 ^FDSCN ^FS ^FO632, 273 ^GB176,103,4 ^FS ^FO640,280 ^ADN,18,5
^FDPeso Kg ^FS ^FO680,318 ^ADN,36,20 ^FD12.0 ^FS ^FO632, 372 ^GB176,48,4 ^FS
^FO640,378 ^ADN,18,5 ^FDPorto: ^FS ^FO658,398 ^ADN,18,10 ^FDFRANCO ^FS
^FO4,372 ^GB804,48,4 ^FS ^FO25,383 ^ADN,36,10 ^FDNote: ^FS ^FO88,383
^ADN,36,10 ^FD ^FS ^FO640,650 ^XGE:GLSMINI.GRF,1,1 ^FS ^FO606,718 ^GB202,82,4
^FS ^FO4,718 ^GB606,82,4 ^FS ^FO50,422 ^BY3 ^BCN,260,N,N,N,A
^FVL2600256057010M451^FS ^FO100,687 ^AAN,26,16 ^FDL2 60025605701 0 M451 01
^FS ^XZ",
"status": 0
}
```

### 📄 MULTIPARCEL JSON REQUEST EXAMPLE:

```
{
  "shipment": {
        "CLIENT_SHIP_REF": "TEST_NEW_WS_001",
        "SMM_CUST_CODE": "ITIT0000999",
        "DELIVERY_NAME": "Mario Rossi",
        "DELIVERY_ADDR": "via Farini 81",
        "DELIVERY_CITY": "Milano",
        "DELIVERY_ZIP_CODE": "20159",
        "WEIGHT": 3,
        "DEL_ISO_COUNTRY_CODE": "IT",
        "PRE_ALLERT_DATE": "2020-07-20",
        "DELIVERY_DISTRICT": "MI",
        "MULTI_PARCEL": 1,
        "packages": [
              {
                    "WEIGHT": 1,
                    "DIMEN_PCS_HEIGHT": 10,
                    "DIMEN_PCS_LENGTH": 11,
                    "DIMEN_PCS_WIDTH": 12
              },
              {
                    "WEIGHT": 2,
                    "DIMEN_PCS_HEIGHT": 20,
                    "DIMEN_PCS_LENGTH": 21,
                    "DIMEN_PCS_WIDTH": 22
              }
        ]
  },
  "options": {
        "force_reprint": false,
        "confirmed": false,
        "delete_if_exists": false,
        "base64": true
  }
}
```

### ↓ MULTIPARCEL RESPONSE EXAMPLES

### ☑ SUCCESSFUL (STATUS FIELD = 0):

```
{
  "MASTER_USN_PARCEL_NR": "1ZXXXXXXXXXXXXXXXX",
  "CLIENT_SHIP_REF": "TESTMC3_05120006",
  "notes": "",
  "SMM_CUST_CODE": "ITIT0000999",
  "SMM_PRODUCT_CODE": "INT_UPS_EXPRESS",
  "packages": [
        {
              "MASTER_USN_PARCEL_NR": "TESTMC3_05120006",
              "CLIENT_SHIP_REF": "TESTMC3_05120006",
              "USN_PARCEL_NUMBER": "1ZXXXXXXXXXXXXXXXX",
              "label":
"Cl5YQQ0KXkxSTg0KXk1OWQ0KXk1GTixODQpeTEgxMCwxMg0KXk1DWQ0KXlBPSQ0KXlBXODEyDQpe
Q0kyNw0KXkZPMTUsN15BME4sMjAsMjReRlZDTElFTlRFIRFU1ReRlMN........peWFoNCg=="
        },
        {
              "MASTER_USN_PARCEL_NR": "TESTMC3_05120006",
              "CLIENT_SHIP_REF": "TESTMC3_05120006",
```

```
                    "USN_PARCEL_NUMBER": "1ZXXXXXXXXXXXXXXXX",
                    "label":
"Cl5YQQ0KXkxSTg0KXk1OWQ0KXk1GTixODQpeTEgxMCwxMg0KXk1DWQ0KXlBPSQ0KXlBXODEyDQpe
Q0kyNw0KXkZPMTUsN15BME4sMjAsMjReRlZDTElFTlRFIFRFU1ReRlMNCl5GTzE1LDI3XkEwTiwyM
CwyNF5GVjAyNjYyMjc3OTleRlMNCl5GTzE1LDQ3XkEwTiwyMCwyNF5GVkNMSUVOVEUgVEVTVF5GUw
0KXkZPMTUsNjdeQTBOLDIwLDI0XkZWVklBIENB.....KXkRODQpeWFoNCg=="
                }
            ],
    "Haulier": "PAITITINT0999",
    "status": 0
}
```

---

## 📄 DROPOFF JSON REQUEST EXAMPLE:

```
{
    "shipment": {
            "DELIVERY_POINTRELAYID": "IT-PDB-RTZ-Z6076",
            "CLIENT_SHIP_REF": "TestDrop0007N",
            "DELIVERY_EMAIL": "ict.smm@sanmarinomail.it",
            "DELIVERY_MOBILE_PH": "9994602892",
            "DELIVERY_NAME": "ICT",
            "DELIVERY_NAME_2": "",
            "DELIVERY_PHONE_NR": "",
            "DEL_ISO_COUNTRY_CODE": "IT",
            "LDV_PRINTING": 1,
            "NR_OF_PCS": 1,
            "OUTPUT_TYPE": 0,
            "PRE_ALLERT_DATE": "2024-10-24",
            "SMM_CUST_CODE": "ITIT0000999",
            "MULTI_PARCEL": 0,
            "packages": [
                    {
                            "WEIGHT": 3.98,
                            "DIMEN_PCS_HEIGHT": 34.0,
                            "DIMEN_PCS_LENGTH": 39.0,
                            "DIMEN_PCS_WIDTH": 28.0,
                            "TOT_VOLUME": 0.0,
                            "CLIENT_SHIP_REF": "TestDrop0007N"
                    }
            ]
    },
    "options": {
            "force_reprint_if_exists": false,
            "confirmed": false,
            "delete_if_exists": false,
            "base64": true
    },
    "mirror": {
            "field1": "temp"
    }
}
```

↓ DROPOFF JSON RESPONSES:

### ❌ ERROR;

```
{
    "mirror": {
        "field1": "temp"
    },
    "CLIENT_SHIP_REF": "TestDrop0008N",
    "notes": "",
    "errorDescription": "Error generating label Cannot get dropoff finfor-
mation",
    "SMM_CUST_CODE": "ITIT0000999",
    "errorCode": 999,
    "label": "XlhBDQoNCl5GW.....",
    "SMM_PRODUCT_CODE": "IT_PDBEXPXL_DROPOFF",
    "Haulier": "PAITITPDB0050",
    "status": -1
}
```

### ✅ SUCCESSFUL (STATUS FIELD = 0):

```
{
    "MASTER_USN_PARCEL_NR": "TestDrop0007N",
    "mirror": {
        "field1": "temp"
    },
    "CLIENT_SHIP_REF": "TestDrop0007N",
    "notes": "",
    "SMM_CUST_CODE": "ITIT0000999",
    "USN_PARCEL_NUMBER": "3UX000X000001",
    "label": "XlhBCgpeRk8zMCw4MjVeR0ZBLDM5MzYsMzkzNiwxNiwsOjo6Ojo6Ojo....",
    "SMM_PRODUCT_CODE": "IT_PDBXXX_DROPOFF",
    "Haulier": "PAITITPDB0050",
    "status": 0
}
```

## GET LABEL

This operation allows you to retrieve a label for a referenced shipment; to be able to gain access is mandatory to have a valid token. The shipment reference for the label to retrieve must be passed as JSON string.

Mandatory fields are:

- `SMM_CUST_CODE`: San Marino Mail client id (string)
- `CLIENT_SHIP_REF`: shipment reference (string)

Optional fields are:

- `base64`: ask for a label string encoded in base64 or not; (Boolean default is true)

### 🔗 URL

```
https://ws3.sanmarinomail.it/SmmWs3/retrievelabel
```

### 📄 JSON REQUEST EXAMPLE:

```
{
   "CLIENT_SHIP_REF": "testreference",
   "SMM_CUST_CODE": "TEST_ID",
   "base64": false,
   "mirror":{
        "cod_attivita_infolog":"temp",
        "cod_stazione":"temp2"
   }
}
```

### ⬆ CURL EXAMPLE

```
curl -H "Accept: application/json" -H "token: {token}TEST-TOKEN-a0a2-47b0-
b1ab-923d2ec4b8ce" -H "username: testusername" -d '{"CLI-
ENT_SHIP_REF":"testreference","SMM_CUST_CODE":"TEST_ID","base64":false…' -X
POST
http://localhost:8080//retrievelabelhttps://ws3.sanmarinomail.it/SmmWs3/retri
evelabel
```

### ⬇ RESPONSE EXAMPLES:

❌ Error:
```
{
   "errorDescription": "No shipments found for reference testreference",
   "errorCode": 1031,
   "status": -1,
   "mirror": {
        "cod_attivita_infolog":"temp",
        "cod_stazione":"temp2"
   }
}
```

## ✅ SUCCESSFUL (STATUS FIELD = 0):

```
{
  "CLIENT_SHIP_REF": "testreference",
  "SMM_CUST_CODE": "TEST_ID",
  "label": "^XA^PON^PQ1^MUd^PW79..............Special Instruction:^FS^XZ",
  "status": 0,
  "usn":"xxx",
  "mirror": {
        "cod_attivita_infolog":"temp",
        "cod_stazione":"temp2"
  }
}
```

## DELETE LABEL

This operation allows to delete a generated label; to be able to gain access is mandatory to have a valid token.

The shipment reference for the label to delete must be passed as JSON string.

Mandatory fields are:

- `SMM_CUST_CODE`: San Marino Mail client id (string)
- `CLIENT_SHIP_REF`: shipment reference (string)

### 🔗 URL

```
https://ws3.sanmarinomail.it/SmmWs3/deletelabel
```

### 📄 JSON REQUEST EXAMPLE:

```
{
  "CLIENT_SHIP_REF": "testreference",
  "SMM_CUST_CODE": "TEST_ID"
}
```

### ⬆ CURL EXAMPLE

```
curl -H "Accept: application/json" -H "token: {token}TEST-TOKEN-a0a2-47b0-
b1ab-923d2ec4b8ce" -H "username: testusername" -d '{"CLI-
ENT_SHIP_REF":"testreference","SMM_CUST_CODE":"TEST_ID…' -X POST
http://localhost:8080//https://ws3.sanmarinomail.it/SmmWs3/deletelabel
```

### ⬇ RESPONSE EXAMPLES:

#### ❌ ERROR:

```
{
  "CLIENT_SHIP_REF": "TEST_NEW_WS_003",
  "errorDescription": "no label to delete with reference TEST_NEW_WS_003",
  "errorCode": 3003,
  "status": -1
}
```

#### ✅ SUCCESSFUL (STATUS FIELD = 0):

```
{
  "CLIENT_SHIP_REF": "TEST_NEW_WS_004",
  "notes": "Label deleted succesfully",
  "SMM_CUST_CODE": "ITIT0000999",
  "status": 0
}
```

## MANIFEST

This operation allows you to confirm a list of shipments and retrieve a PDF report in Base64 format.

To use this web service, it is mandatory to include certain headers in the request. The request must be passed as JSON string.

Mandatory fields are:

- `SMM_CUST_CODE`: San Marino Mail client id (string)
- `CLIENT_SHIP_REF_LIST`: shipment reference list json array

### 🔗 URL

```
https://ws3.sanmarinomail.it/SmmWs3/manifest
```

### 📄 JSON REQUEST EXAMPLE:

```
{
    "manifest”: {
        "SMM_CUST_CODE": "ITIT0000999",
        "CLIENT_SHIP_REF_LIST": [
            "TEST_NEW_WS_003",
            "TEST_NEW_WS_004"
        ]
    }
}
```

### ⬆ CURL EXAMPLE

```
curl -H "Accept: application/json" -H "token: {token}TEST-TOKEN-a0a2-47b0-
b1ab-923d2ec4b8ce" -H "username: testusername" -d '{  "manifest":
{    "SMM_CUST_CODE": "ITIT0000999",    "CLIENT_SHIP_REF_LIST":
["TEST_NEW_WS_003","TEST_NEW_WS_004"]}}' -X POST
https://ws3.sanmarinomail.it/SmmWs3/manifest
```

### ⬇ RESPONSE EXAMPLES:

#### ❌ ERROR:

```
{
  "errorDescription": "Error No shipments to be manifested",
  "errorCode": 1065,
  "status": -1
}
```

#### ✅ SUCCESSFUL (STATUS FIELD = 0):

```
{
  "SMM_CUST_CODE": "ITIT0000999",
  "manifestPdf":
"JVBERi0xLjQKJfbk/N8KMSAwIG9iago8PAovVHlwZSAvQ2F0YWxvZ.........",
  "status": 0
}
```

## DOCUMENT UPLOAD

This operation allows you to upload a file (in base64 format) related to a shipment. When using the "Create Shipment" method populating with value 1 the "`DOCUMENTS_NEEDED`" field, every document related to the shipment (related to the `CLIENT_SHIP_REF` field) will be uploaded to the carrier. The request must be passed as JSON string.

> PLEASE NOTE: UPLOAD ALL THE FILES
> BEFORE CALLING THE GENERATE LABEL REQUEST

Mandatory fields are:

- `action`: "**UPLOAD**"
- `SMM_CUST_CODE`: San Marino Mail client id (string)
- `CLIENT_SHIP_REF`:  shipment reference
- `category`: document category (see list below)
- `documentdata64`: base64 encoding string of file
- `filename`: filename
- `filetype`: filetype (ex. pdf, tiff, png, …)

Optional Fields:

    `SMMProductCode`

Document category list:

|     | SMM_code | DESCRIPTION |
| --- | --- | --- |
| 1 | PDC001 | Commercial invoice |
| 2 | PDC002 | CIN Commercial invoice |
| 3 | PDC003 | Pro forma invoice |
| 4 | PDC004 | Certificate of origin |
| 5 | PDC005 | NAFTA certificate |
| 6 | PDC006 | Custom Declaration |
| 7 | PDC007 | HWB |
| 8 | PDC008 | OTHER |
| 9 | PDC009 | ETD |
| 10 | PDC010 | AUTH FORM |
| 11 | PDC011 | Export Accompanying doc |
| 12 | PDC012 | Export License |
| 13 | PDC013 | Import Permit |
| 14 | PDC014 | Power of Attorney |
| 15 | PDC015 | Packing list |
| 16 | PDC016 | Sed document |
| 17 | PDC017 | shipper letter of instruction |
| 18 | PDC018 | USMCA Commercial Invoice |
| 19 | PDC019 | USMCA Certification of Origin |

## 🔗 URL

```
https://ws3.sanmarinomail.it/SmmWs3/documents
```

## 📄 JSON REQUEST EXAMPLE:

```
{
   "document": {
       "action": "upload",
       "SMM_CUST_CODE": "ITIT0000999",
       "CLIENT_SHIP_REF": "TEST_PPP2",
       "category": "0B",
       "filename" : "test.pdf",
       "filetype" : "pdf",
       "documentdata64":
"JVBERi0xLjYNJeLjz9MNCjM3IDAgb2JqIDw8L0xpbmVhcml6ZWQgMS9MIDIwNTk3L08gNDAvRSAx
NDExNS9OIDEvVCAxOTc5NS9II...."
   }
}
```

## ⬆ CURL EXAMPLE

```
curl -H "Accept: application/json" -H "token: {token}TEST-TOKEN-a0a2-47b0-
b1ab-923d2ec4b8ce" -H "username: testusername" -d '{     "document": {
   "SMM_CUST_CODE": "ITIT0000999",          "CLIENT_SHIP_REF": "TEST_PPP2",
        "category": "0B",            "filename" : "test.pdf",
   "filetype" : "pdf",          "documentdata64":
"JVBERi0xLjYNJeLjz9MNCjM3IDAgb2JqIDw8L0xpbmVhcml6ZWQgMS9MIDIwNTk3L08gNDAvRSAx
NDExS9OIDEvVCAxOTc5NS9II....."   } }' -X POST
https://ws3.sanmarinomail.it/SmmWs3/documentupload
```

## ⬇ RESPONSE EXAMPLES:

### ❌ ERROR:

```
{
   "errorDescription": "missing mandatory fields",
   "errorCode": 3030,
   "status": -1
}
```

### ✅ SUCCESSFUL (STATUS FIELD = 0):

```
{
   "CLIENT_SHIP_REF": "TEST_doc",
   "SMM_CUST_CODE": "ITIT0000999",
   "DOCUMENT_ID": "ITIT0000999_00000000003",
   "category": "0B",
   "status": 0
}
```

## DOCUMENT DELETE

This operation allows you to delete a previously uploaded file. The request must be passed as JSON string

Mandatory fields are:

- `action`: "**DELETE**"
- `SMM_CUST_CODE`: San Marino Mail client id (string)
- `CLIENT_SHIP_REF`: shipment reference
- `DOCUMENT_ID`: identificativo documento ricevuto in risposta alla chiamata di upload

### 🔗 URL

```
https://ws3.sanmarinomail.it/SmmWs3/documents
```

### 📄 JSON REQUEST EXAMPLE:

```
{
   "document": {
        "action": "delete",
        "SMM_CUST_CODE": "ITIT0000999",
        "CLIENT_SHIP_REF": "TEST_PPP2",
        "DOCUMENT_ID": "ITIT0000999_00000000003"
   }
}
```

### ⬆ CURL EXAMPLE

```
curl -H "Accept: application/json" -H "token: {token}TEST-TOKEN-a0a2-47b0-
b1ab-923d2ec4b8ce" -H "username: testusername" -d 'MISSING' -X POST
https://ws3.sanmarinomail.it/SmmWs3/documentupload
```

### ⬇ RESPONSE EXAMPLES:

#### ❌ ERROR:

```
{
   "errorDescription": "missing mandatory fields",
   "errorCode": 3030,
   "status": -1
}
```

#### ✅ SUCCESSFUL (STATUS FIELD = 0):

```
{
   "CLIENT_SHIP_REF": "TEST_doc",
   "SMM_CUST_CODE": "ITIT0000999",
   "DOCUMENT_ID": "ITIT0000999_00000000003",
   "category": "0B",
   "status": 0
}
```

## TRACKING

This operation allows you to retrieve all the events for a referenced shipment; to be able to gain access is mandatory to have a valid token. The shipment reference for tracking service must be passed as JSON string:

Mandatory fields are:

- `CLIENT_SHIP_REF`: Shipment reference (string) or `USN`: shipment (string)
- `SMM_CUST_CODE`: San Marino Mail client id (string)

### 🔗 URL

```
https://ws3.sanmarinomail.it/SmmWs3/tracking
```

### 📄 JSON REQUEST EXAMPLE:

```
{
   "clientshipref": "testreference",
   "SMM_CUST_CODE": "TEST_ID"
}
```

### ⬆ CURL EXAMPLE

```
curl -H "Accept: application/json" -H "token: {token}TEST-TOKEN-a0a2-47b0-
b1ab-923d2ec4b8ce" -H "username: testusername" -d '{"client-
shipref":"testreference","SMM_CUST_CODE": "TEST_ID"}' -X POST
https://ws3.sanmarinomail.it/SmmWs3/tracking
```

### ⬇ RESPONSE EXAMPLES:

#### ❌ ERROR:

```
{
   "errorDescription": "Shipment not found",
   "errorCode": 3003,
   "status": -1
}
```

#### ✅ SUCCESSFUL (STATUS FIELD = 0):

```
{
   {
   "usn": "XXXXXXXXXXXXXXXX",
   "country": "United Kingdom",
   "productCode": "GGGGGGGGG",
   "invoiceecustref": "testreference",
   "countryiso": "GB",
   "clientshipref": "testreference",
   "status": 0,
   "events": [
         {
               "date": "06/12/19",
               "eventcode": "104",
               "eventdescription": "shipment has been delivered",
               "time": "12:14",
               "eventname": "delivered"
```

```
        },
        {
                "date": "06/12/19",
                "eventcode": "141",
                "eventdescription": "shipment has been given to driver for de-
livery",
                "time": "09:20",
                "eventname": "out for delivery"
        },

        ...

        {
                "date": "28/11/19",
                "eventcode": "0",
                "eventdescription": "shipment has been pre alerted by sender",
                "time": "15:14",
                "eventname": "shipment booked"
        }
    ]
  }
}
```

## ANNEXES

Here's a list of all available fields related to the shipment (to be passed on in the json object "shipment"):

| NAME | DESCRIPTION | TYPE | BYTE | MAND. | NOTE |
|---|---|---|---|---|---|
| SMM_CUST_CODE | Company Id | T | 20 | Y | |
| SMM_PRODUCT_CODE | Product code | T | 50 | N | |
| PRE_ALLERT_DATE | Pre allert Date | T | 10 | N | (yyyy-MM-dd) |
| CLIENT_SHIP_REF | Client Shipment Reference | T | 35 | Y | |
| USN_PARCEL_NUMBER | USN parcel number | T | 50 | N | |
| MASTER_USN_PARCEL_NR | Master USN | T | 30 | N | |
| DELIVERY_NAME | Delivery Name | T | 35 | Y | |
| DELIVERY_STR_TYPE | Delivery Street type | T | 30 | N | |
| DELIVERY_ADDR | Delivery address 1 | T | 35 | Y | |
| DELIVERY_ADDR_NO | Delivery address No | T | 5 | N | |
| DELIVERY_ZIP_CODE | Delivery Zip code | T | 12 | Y | |
| DELIVERY_CITY | Delivery City | T | 50 | Y | |
| DELIVERY_DISTRICT | Delivery district (Mandatory for italy) | T | 50 | Y/N | |
| DEL_ISO_COUNTRY_CODE | Delivery iso country | T | 3 | Y | |
| NR_OF_PCS | Number of items | N | 10 | Y | |
| WEIGHT | Weight in KG | N | 10 | Y | |
| PACKAGING_TYPE | Packaging type | T | 40 | N | |
| LDV_PRINTING | Ldv printing (1 , 0) (default 0) | N | 1 | N | |
| ADDRESS_CLEANING | Address cleaning (1,0) | N | 1 | N | |
| CUST_SERVICE_ACTIVITY | Customer service (1,0) | N | 1 | N | |
| WAREHOUSE_TRANSIT | Warehouse Transit (1,0) | N | 1 | N | |
| PARCEL | Parcel Number | N | 10 | Y | |
| SMS_PREADVISE_REQUEST | SMS preadvise | N | 1 | N | |

| EMAIL_PREADVISE_REQUEST | Email preadvise | N | 1 | N | |
|---|---|---|---|---|---|
| CLIENT_SHIP_REF_2 | Client Shipment reference 2 | T | 35 | N | |
| DELIVERY_NAME_2 | Delivery name 2 | T | 35 | N | |
| DELIVERY_ADDR_2 | Delivery address 2 | T | 35 | N | |
| DELIVERY_ADDR_3 | Delivery address 3 | T | 35 | N | |
| DELIVERY_ADDR_4 | Delivery address 4 | T | 35 | N | |
| DELIVERY_PHONE_NR | Delivery phone number | T | 50 | N | |
| DELIVERY_FAX_NR | Delivery fax number | T | 50 | N | |
| DELIVERY_MOBILE_PH | Delivery mobile phone | T | 50 | N | |
| DELIVERY_EMAIL | Delivery email | T | 50 | N | |
| DELIVERY_INSTRUCTION | Delivery Instruction | T | 150 | N | |
| DELIVERY_ADDRESS_ID | Delivery address Id | T | 12 | N | |
| DELIVERY_FISCAL_CODE | Delivery fiscal code | T | 50 | N | |
| DELIVERY_SATURDAY | Delivery saturday (1,0) | N | 1 | N | |
| DIMEN_PCS_LENGTH | Length | N | 10 | N | |
| DIMEN_PCS_WIDTH | Width | N | 10 | N | |
| DIMEN_PCS_HEIGHT | Height | N | 10 | N | |
| TOT_VOLUME | Total volume | N | 10 | N | |
| DECLARED_GOODS_VALUE | Declared goods value | N | 10 | N | |
| CUST_VALUE_CURRENCY | Value currency | T | 10 | N | |
| SENDER_VAT_NUMBER | VAT number | T | 50 | N | |
| SENDER_DOCUMENT_NR | Document Nr | T | 50 | N | |
| INVOICE_NR_FOLLOWING | Invoice Nr | T | 50 | N | |
| COD_AMOUNT | COD amount (otherwise 0) | N | 10 | N | |
| COD_PAYM_METHOD | COD Payment method (CON) | T | 50 | N | |
| COD_CURRENCY | COD currency | T | 10 | N | |
| INSURANCE_VALUE | Insurance value | N | 10 | N | |
| INSURANCE_CURRENCY | Insurance currency | T | 10 | N | |

| | | | | | |
|---|---|---|---|---|---|
| PICKUP_NAME | Pickup name | T | 50 | N | |
| PICKUP_NAME2 | Pickup name 2 | T | 50 | N | |
| PICKUP_ADDR | Pickup address 1 | T | 100 | N | |
| PICKUP_ADDR_NO | Pickup address No | T | 5 | N | |
| PICKUP_ADDR_2 | Pickup address 2 | T | 100 | N | |
| PICKUP_ADDR_3 | Pickup address 3 | T | 100 | N | |
| PICKUP_ADDR_4 | Pickup address 4 | T | 100 | N | |
| PICKUP_ZIP_CODE | Pickup zipcode | T | 10 | N | |
| PICKUP_CITY | Pickup city | T | 50 | N | |
| PICKUP_PHONE_NR | Pickup phone nr | T | 50 | N | |
| PICKUP_FAX_NR | Pickup fax nr | T | 50 | N | |
| PICKUP_MOBILE_PH | Pickup mobile nr | T | 50 | N | |
| PICKUP_EMAIL | Pickup email | T | 50 | N | |
| PICKUP_ISO_COUNTRY_CODE | Pickup ISO country | T | 3 | Y | |
| PICKUP_ADDR_ID | Pickup address Id | T | 12 | N | |
| DELIVERY_AT_FLOOR | Delivery at floor (1,0) | N | 1 | N | |
| DELIVERY_APPOINTMENT | Delivery appointment (1,0) | N | 1 | N | |
| DELIVERY_EVENING | Delivery evening (1,0) | N | 1 | N | |
| PICKUP_USED_RAEE | Used RAEE (1,0) | N | 10 | N | |
| PICKUP_DATE | Pickup date | T | 10 | N | (yyyy-MM-dd) |
| OPENING_HOUR_MORNING | Opening hour morning | N | 1 | N | |
| CLOSING_HOUR_MORNING | Closing hour morning | N | 1 | N | |
| OPENING_HOUR_AFTERNOON | Opening afternoon | N | 1 | N | |
| CLOSING_HOUR_AFTERNOON | Closing afternoon | N | 1 | N | |
| PICKUP_WINDOW | Pickup window | T | 2 | N | |
| PICKUP_DISTRICT | Pickup district | T | 2 | N | |
| CONTENT | Content type | T | 50 | N | |
| GOODS_VALUE | Commercial value | N | 10 | N | |

| | | | | | |
|---|---|---|---|---|---|
| GOODS_CURRENCY | *Commercial currency* | T | 3 | N | |
| CUSTOM_CODE | *Customs code* | T | 12 | N | |
| OUTPUT_TYPE | *OutputType (0=ZPL,1=PDF)* | N | 1 | Y | |
| DELIVERY_GDO | *GDO delivery* | N | 1 | N | |
| DELIVERY_NDS | *Neutral delivery service* | N | 1 | N | |
| MULTI_PARCEL | *set to 1 for multiparcel shipment (default 0)* | N | 1 | N | |
| NR_OF_PARCELS | *Number of parcels* | N | 10 | N | |
| RETURN_LABEL | *return label needed* | N | 1 | N | |
| RETURN_PRODUCT_CODE | *Return label Product code* | T | 50 | N | |
| DOCUMENTS_NEEDED | *set to 1 if previously uploaded documents must be sent to the carrier (default 0)* | N | 1 | N | |
| DELIVERY_POINTRELAYID | *Dropoff id point* | T | 50 | * | N for Home Delivery, Y for Dropoff delivery |

**Table 1 – createShipment available Fields**

Here's a list of all available fields related to the parcel (to be passed in the json onbject "packages"):

| NAME | DESCRIPTION | TYPE | BYTE | MAND. | NOTE |
|---|---|---|---|---|---|
| CLIENT_SHIP_REF | Client Shipment Reference | T | 35 | N | |
| WEIGHT | Weight in KG | N | 10 | Y | |
| DIMEN_PCS_LENGTH | Length | N | 10 | Y | |
| DIMEN_PCS_WIDTH | Width | N | 10 | Y | |
| DIMEN_PCS_HEIGHT | Height | N | 10 | Y | |
| CONTENT | Content type | T | 50 | N | |
| COD_AMOUNT | COD amount | N | 10 | N | (otherwise 0) |
| COD_PAYM_METHOD | COD Payment method (CON) | T | 50 | N | |
| COD_CURRENCY | COD currency | T | 10 | N | |
| INSURANCE_VALUE | Insurance value | N | 10 | N | |
| INSURANCE_CURRENCY | Insurance currency | T | 10 | N | |
| GOODS_VALUE | Commercial value | N | 10 | N | |
| GOODS_CURRENCY | Commercial currency | T | 3 | N | |
| HAZMAT | Hazardous goods | N | 1 | N | |
| HAZMAT_CLASS | Hazardous goods class UN (es. LQ) | T | 20 | N | |
| HAZMAT_SHIPPING_NAME | The Proper Shipping Name assigned by ADR, CFR or IATA. | T | 50 | N | |
| HAZMAT_REGULATION_SET | Hazardous goods The Regulatory set (ex ADR) | T | 5 | N | |
| INCOTERMS | Incoterms | T | 20 | N | |
| HS_CODE | HS Code | T | 20 | N | |
| INVOICE_NR_DATE | Invoice date | T | 10 | N | (yyyy-MM-dd) |
| NR_PIECES_HAZ | Item Quantity of Hazardous goods | N | 10 | N | |
| INVOICE_NR | Invoice Number | T | 30 | N | |
| HAZMAT_UOM | The unit of measure used for the mass capacity of the regulated good. | T | 15 | N | For Example: ml, L, g, mg, kg, cylinder, pound, pint, quart, gallon, ounce |
| HAZ- | This is the hazard class associated to | N | 1 | N | |

| MAT_ClassDivisionNumber | the specified commodity. | | | | |
|---|---|---|---|---|---|
| HAZ-MAT_ChemicalRecordIdentifier | *Should be set to 1* | N | 1 | N | |
| HAZMAT_PackagingType | *The type of package used to contain the regulated good. (Ex: Fiberboard Box). Required if CommodityRegulatedLevelCode = LQ or FR* | T | 255 | N | *Ex. FIBERBOARD BOX, WOOD(EN) BOX, PLASTIC JERRICAN, METAL BOX, STEEL DRUM, OTHER, PLASTIC BOX, PLASTIC DRUM, STYROFOAM BOX, CYLINDERS, ENVIROTAINER, PLYWOOD BOX, ALUMINUM DRUM, ALUMINUM CYLINDERS, PLASTIC PAIL, PLYWOOD DRUM, FIBER DRUM, STEEL JERRICAN, ALUMINUM JERRICAN, STEEL BOX, CARTON, ALUMINUM BOX* |
| HAZMAT_IDNumber | *This is the ID number (UN/NA/ID) for the specified commodity. Required if CommodityRegulatedLevelCode = LR, LQ or FR and if the field applies to the material by regulation.* | T | 6 | N | *UN/NA/ID Identification Number assigned to the specified regulated good. (Include the UN/NA/ID as part of the entry).* |
| HAZMAT_TransportationMode | *Not applicable for ADR regulation set. Required for any other regulation set. Declares that a package was prepared according to ground passenger aircraft or cargo aircraft only.* | T | 30 | N | *Valid values: Highway=Highway Ground=Ground PAX=Passenger Aircraft Passenger Aircraft=Passenger Aircraft CAO=Cargo Aircraft Only Cargo Aircraft Only=Cargo Aircraft Only Valid entries include: Highway, Ground, PAX, Passenger Aircraft, CAO and Cargo Aircraft Only.* |

**Table 2 - createShipments packages Fields**