

7/3/2023

SOFTVFD RECEIPT API

RECEIPT POSTING APPLICATION
PROGRAMMING INTERFACE.
(APIv2.1.2)



Prepared by:

SOFTNET TECHNOLOGIES LIMITED

SOFTVFD API DOCUMENTATION

Welcome to the SOFTVFD API documentation for our SOFTVSD Middleware service. This document provides a comprehensive guide on how to utilize two crucial APIs that are integral to accessing and interacting with our service.

The first API is Token Generation, allows users to acquire a secure token necessary for authentication and authorization purposes. The token serves as a unique identifier and is required for making requests to other APIs. It's important to note that tokens have a limited lifespan and must be renewed when they expire.

The second API is Receipt Posting, enables users to submit receipts to our system. However, to successfully post a receipt, users must include a valid token in their request. If the token provided is expired or invalid, users will need to obtain a new token before proceeding with the receipt posting process.

By incorporating these APIs into your application, you can ensure secure access to our system's resources and maintain control over the receipt submission workflow. The token-based authentication mechanism enhances the overall security of the system by preventing unauthorized access.

This API documentation will guide you through the necessary steps to generate a token, handle token expiration, and post receipts. By following the provided examples and guidelines, you'll be able to seamlessly integrate these functionalities into your application and provide a secure and efficient experience for your users.

Let's dive into the details of each API and explore how they can be effectively utilized to enhance your application's functionality and security.

TOKEN REQUEST API

Endpoint: <https://softapitest.softnet.co.tz/api/GetToken>

Description:

This API is used to generate a token for authentication and authorization purposes. The token serves as a secure identifier for accessing the system's resources and is required for making requests to other APIs. The token has an expiration time, after which it becomes invalid, and a new token must be requested.

Request Payload:

The request payload should be sent as a JSON object with the following parameters:

Field	Data Type		Description
username	String	Mandatory	The username of the user.
password	String	Mandatory	The password associated with the user account
tin	String	Mandatory	The Taxpayer Identification Number (TIN) of the user

Request Payload Sample:

JSON FORMAT

```
{
  "userName": " xxxx",
  "password": " xxxx",
  "tinNumber": "xxx-xxx-xxx"
}
```

Response:

The API will respond with a JSON object containing the following fields:

Field	Data Type		Description
code	Integer		The status code of the response (0 indicates success).
tokenValidityInMinutes	Integer		The validity duration of the token in minutes.
token	String		The generated authentication token
message	String		A message indicating the status of the token request

RESPONSE CODE DESCRIPTION

RESPONSE CODE	MESSAGE
0	Success
1	Invalid login credentials
2	Technical error please contact system administrator

Response Sample:

JSON FORMAT

```
{
  "code": 0,
  "tokenValidityInMinutes": 30,
  "token": "0000051-40i40i4ca-000049",
  "message": "Success"
}
```

Notes:

- The token received in the response should be stored securely and used for subsequent API calls that require authentication.
- The token is valid for the duration specified by the `tokenValidityInMinutes` field.

- If the request fails, the response will contain an appropriate error message in the `message` field and a non-zero value in the `code` field

RECEIPT POSTING API

Endpoint: <https://softapitest.softnet.co.tz/api/PostTransaction>

Description:

This API allows users to post receipts to the system. However, in order to make a successful request, a valid token must be included in the request. If the token has expired, the user needs to obtain a new token by making a request to the Token Generation API.

Request Payload:

The request payload should be sent as a JSON object with the following parameters:

Field	Data Type		Description
token	String	Mandatory	The authentication token associated with the transaction.
paymentType	Integer	Mandatory	The payment type code for the transaction. Available Payment type According to TRA <ul style="list-style-type: none"> - 1. CASH - 2. CHEQUE - 3. CCARD - 4. EMONEY - 5. INVOICE
customerIdType	Integer	Mandatory	The customer identification type code. Available Customer id type According to TRA are: <ul style="list-style-type: none"> - 1. TIN - 2. Driving License - 3. Voter's Number - 4. Passport - 5. NID (National Identity) - 6. NIL (If there is no ID)
customerName	String		The name of the customer

mobileNumber	String		The mobile number associated with the customer.
customerIdNumber	String		The identification number of the customer (Depends on customer id type)
transactionID	String	Mandatory	The unique transaction ID or invoice number.
vrn	String		The Vendor Registration Number associated with the transaction (or "N/A" if not applicable).
receiptItems	Array	Mandatory	An array of objects representing individual items in the transaction receipt.
itemNumber	Integer	Mandatory	The item number or index
taxCode	Integer	Mandatory	The tax code associated with the item. Available Tax code type According to TRA are: <ul style="list-style-type: none"> - 1. Standard rate (18%) - 2. Special rate (0%) - 3. Zero rate (0%) - 4. Special Relief (0%) - 5. Exempt (0%)
quantity	Float	Mandatory	The quantity of the item
paidAmount	Integer	Mandatory	The amount paid for the item
itemDescription	String	Mandatory	The description or name of the item
discount	Float		The discount applied to the item (if any).
unitPrice	Integer	Mandatory	The unit price of the item

Within the Receipt items array, each object represents a specific item in the transaction. The fields within each Receipt items object are as follows:

itemNumber (integer), taxCode (integer), quantity (float), paidAmount (integer), itemDescription (string), discount (float), unitPrice (integer)

Request Payload Sample:

JSON FORMAT

```
{
  "token": "0000051-40i40i4ca-000049",
  "paymentType": 5,
  "customerIdType": 6,
  "customerName": "Mushi, Nelson Jared",
  "mobileNumber": "",
  "customerIdNumber": "",
  "transactionID": "INV000128317",
  "vrn": "N/A",
  "transactionItems": [
    {
      "itemNumber": 1,
      "taxCode": 1,
      "quantity": 1.0,
      "paidAmount": 11800,
      "itemDescription": "Fabric (NPS6 XP) 1m x 1m",
      "discount": 0.0,
      "unitPrice": 10000
    },
    {
      "itemNumber": 2,
      "taxCode": 2,
      "quantity": 1.0,
      "paidAmount": 10000,
      "itemDescription": "Fabric (NPS6 XP) 2m x 2m",
      "discount": 0.0,
      "unitPrice": 10000
    }
  ]
}
```


Response:

The API will respond with a JSON object containing the following fields:

Field	Data Type		Description
code	Integer		The status code of the response (0 indicates success).
message	String		A message indicating the status of the token request
receiptID	Integer		This is receipt identification number of the transaction you posted. This is also identified by TRA as receipt number of the transaction posted
receiptVerificationCode	String		This is receipt verification code
verificationUrl	String		This is verification link that can be used to verify posted transaction on TRA website
totalTransactionAmount	Double		Total amount of the posted Transaction
totalTaxAmount	Double		Total amount of Tax posted

RESPONSE CODE DESCRIPTION

Response Code	Message
0	Success
1	Invalid operator ID
2	Invalid token
3	Failed to insert Transaction
4	Failed to generate receipt number
5	Technical error please contact system administrator
6	Receipt has no items
7	Transaction ID exist

Response Sample:

JSON FORMAT

```
{
  "code": 0,
  "message": "Success",
  "receiptID": "3",
  "transactionID": "INV000128317",
  "receiptVerificationCode": "680F933",
  "verificationUrl": "https://verify.tra.go.tz/680F933_182910",
  "totalTaxAmount": 1800.0
  "totalTransactionAmount": 21800.0
}
```

Notes:

- If the request fails, the response will contain an appropriate error message in the `message` field and a non-zero value in the `code` field.

This API endpoints are available in testing environment only. The live endPoints will be shared after successful UAT.