

Technical Specification for Data Sharing through Software Fiscal Component from TIER 1 Retailer with FBR

 [Print This Post](#)

Posted on December 5, 2019 | No comments

Introduction

The purpose of the document is to facilitate **TIER 1 RETAILER** to understand methods for invoice data sharing with FBR. The software fiscal component will be installed on the same computer on which POS is installed. The Fiscal component will be integrated with POS system to fiscalize the invoice and returns Fiscal Invoice Number. For further guidance on integration of retail outlets with board's computerized system, please follow the [Sales Tax Rules, 2006 Chapter XIV-AA Online Integration of Tier-1 Retailers](#).

Table of Contents

1. Introduction
2. Scope
3. Defination and Acronyms
4. POS Registration
5. Installation of the FBR Software Fiscal Component
6. Integration of Software Fiscalization Component
 - 6.1. Invoice Model Details
 - 6.2. .Net Code Example: Fill the invoice object
 - 6.2.1. WebClient to post data to provided URL and get the response
 - 6.3. Web Client to post data from Cloud and get the response
 - 6.4. Sample JSON Format to POST to IMS Component
 - 6.5. Posting Model in Postman
 - 6.5.1. GET Endpoint
 - 6.5.2. POST Endpoint
7. Uninstall
8. Technical Support

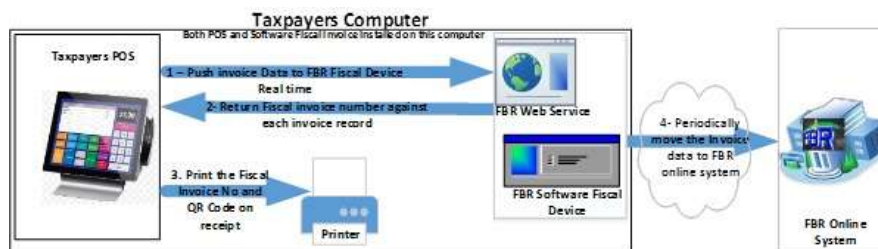


Figure 1 – Invoice Fiscalization Process

POS will be integrated with FBR as per following steps.

- Register each POS in FBR system from which you want to upload the data. Details are given in Section **POS Registration**.
- Download the software component from the FBR website
- Install the software on the local computer (**internet connection is required for this step*).
- Taxpayer will integrate their system with FBR software component as per provided details in the Section **Integration of Taxpayer System with FBR System**.

Scope

The scope of the document is limited to technical description of FBR software fiscal component. FBR Software Fiscal component will reside on the same system where the POS is installed by the taxpayer. It will fiscalize each invoice generated through the POS on real time and generate unique fiscal invoice number for each invoice to print on receipt with QR code. Software fiscal component will periodically move recorded invoices data to FBR online system through an independent process.

Defination and Acronyms

Acronym	Defination
FBR	Federal Board of Revenue
NTN	National Tax Number

POS Registration

Tier 1 Retail sector will register their POS for each branch using eFBR portal by providing the required data.

For POS registration, log in on web site <https://e.fbr.gov.pk> with your user credential and get registered by accessing the page Registration à POS Client Registration

On successful submission of data, FBR will generate POS registration number which will be referenced in compiling each invoice into required format.

Following is the detail of required data for POS registration:

Point of Sale Registration

Business Information	Contact Information	Branch Information	Pos Details	Generate Test POS
NTN 8000023-1	Business Name FBR Mobile Test User 3			
Brand Name(s) test	Product(s) test	Manufacturer Yes		
Estimated annual turnover 1000	Estimated number of transactions per day 100	WebSite URL http://www.webs.com		
POS Software Name POS j	POS Software front/backend technologies Windows	POS Type Client server		
POS Software Vendor Vendor jmn	In-house software development Yes	Number of Technical Staff 1		
<input type="button" value="Save"/> <input type="button" value="Cancel"/>				

Figure 2 – POS Registration – Business Information

Business Information

Field	Mandatory	Details
NTN	System Generated	
Business Name	System Generated	
STRN	No	Open field (Numeric)
Brand Name	No	E.g. Outfitters
Products	No	Open field
Register for	Yes	Options for integration of income tax or sales tax under SRO 779(1)/2020 dated 26th August 2020
Manufacturer	Yes	Yes/No
Estimated annual turnover	No	Open field (Numeric)
Estimated number of transactions per day	No	Open field (Numeric)
Website URL	No	Open field
POS Software name	No	Open field
POS software front/back end technologies	No	Open field
POS type	Yes	Cloud Based/Client Server
POS software vendor	No	Open field
In-house software development	Yes	Yes/No
Number of technical staff	No	Open field (Numeric)

8000023-1 - IND (RTO ISLAMABAD)

Point of Sale Registration

Business Information | **Contact Information** | Branch Information | Pos Details | Generate Test POS

Contact Person: [Text Field] Mobile: [Text Field] Landline number: [Text Field]

Email: [Text Field] Address: [Text Field]

Contact Type: [Select]

[Add] [Cancel]

Seq No	Person	Mobile	LandLine	Email	Address	ContactType	
1	Person	01234567890	01234567890	email@email.com	address	Business	Edit

Figure 3 – POS Registration – Contact Information

Contact Information

Field	Mandatory	Field Information
Contact person	Yes	Contact Person Name
Mobile	Yes	Open field
Landline number	No	Open field
Email	Yes	Open field
Address	Yes	Open field
Contact Type	Yes	General/Technical

8000023-1 - IND (RTO ISLAMABAD)

Point of Sale Registration

Business Information | Contact Information | **Branch Information** | Pos Details | Generate Test POS

Business Branch: [Text Field] City: [Select] Weekly off days: [Mon] [Tues] [Wed] [Thur] [Fri] [Sat] [Sun]

Sector: [Select] Branch Address: [Text Field]

Contact Person: [Select] Latitude: [Text Field] Longitude: [Text Field]

Version: [Text Field] Franchise: [Select] Store Timings: [Select] [Select]

[Add Detail] [Cancel]

Seq No	Branch Name	BranchAddress	City	Contact Person	Sector	
1	branch	address	ABBASPUR	Person	Leather and Textile	Edit

Figure 4 – POS Registration – Branch Information Details

Branch Information

Field	Mandatory	Field Information
Business Branch	Yes	Branch Name
City	Yes	Branch City
Weekly Off Days	Yes	Select off days i.e. when the POS shall not be operational.
Sector	Yes	Select from list of sectors
Branch Address	Yes	Open field
Contact Person	Yes	Select from provided General contacts
Latitude	Yes	Open field

Longitude	Yes	Open field
Version	Yes	Open field
Franchise	Yes	Yes/No
Store Timings (From and To)	Yes	Open fields

8000023-1 - IND (RTO ISLAMABAD)

Point of Sale Registration

Business Information	Contact Information	Branch Information	Pos Details	Generate Test POS					
POS Branch Select	POS Identification Number [Text Field]	POS Type Select	Mac Address [Text Field]	IP Address [Text Field]					
<input type="button" value="Add Detail"/> <input type="button" value="Cancel"/>									
Sr	POS ID	Identification No	Branch Name	POS Status	Connected Date	Code	Edit	Reset POSID	Deactivate POSID
1	110014	PRAL-001	STP I-9/3	Connected	8/12/2020 2:38:18 PM	12345678	Edit	Reset POSID	Deactivate POSID
2	110019	34	fsdf	Disconnected	9/18/2019 11:29:01 AM	12345678	Edit	Reset POSID	Deactivate POSID
3	110547	Haseeb-01	I-9 BRANCH	Disconnected	9/20/2019 4:11:29 PM	12345678	Edit	Reset POSID	Deactivate POSID

Figure 5 – POS Registration – POS Details

Pos Details

Field	Mandatory	Field Information
POS Branch	Yes	Select from list of Branches
POS Identification Number	Yes	Open field (Unique)
POS Type	Yes	Select from list of Types(You can select any option. Mostly Primary is selected for daily used POS PC)
Mac Address	Yes	Open field
IP address	Yes	Open field

Generate Test POS

Business Information	Contact Information	Branch Information	Pos Details	Generate Test POS
Generate Test POS IDs				
<input type="button" value="Generate Test POS IDs"/>				
Seq. No	Pos Registration Number	BusinessName	Code	Reset POSID
1	910415	Shabbir Tiles and Ceramics Ltd	12345678	Reset POSID
2	910416	Delectane mgt pvt limited	12345678	Reset POSID
3	910417	Delectante Management Pvt Ltd	12345678	Reset POSID

Figure 6 – POS Registration – Generate Test POS

TaxPayer can generate Test POS ID from this screen.

Integration of POS with FBR Software Fiscal Component

FBR has provided the local software fiscal Component to Tier 1 Retail sector to share the invoice data with FBR. This component will be accessible only to registered POS Clients. Transaction handling and relevant data dissemination to FBR PRAL will be the responsibility of registered POS Clients.

To facilitate taxpayers, 2 separate Software components shall be made available i.e. a software component for testing and another for live transactions.

For the integration of taxpayer POS with FBR software fiscal Component require following steps:

1. Download the Software Fiscal Component from FBR Web Site
 - https://download.fbr.gov.pk/IMS_Setup/FBRIMS.zip

2. Install the Setup as instructed in section **Installation of the FBR Software Fiscal Component**.
3. After the installation, a Self Hosted Http service will be configured on following address bindings to be integrated with POS:
 - <http://localhost:8524/api/IMSFiscal/get>
 - It will show "Service is responding"
 - **Note:** To verify the successful installation of fiscal service. Please confirm fiscalization service is up and running in the system services list (service.msc) as shown in Figure -7.
4. Consume the local fiscal service (<http://localhost:8524/api/IMSFiscal/get>) in the POS according to given instruction in Section **Steps of Integration for Http Service Consumption**.
5. FBR service will return the invoice number against each record.
6. The FBR Invoice number should be printed on the Customer invoice receipt along with QR Code. As POS Invoice # length ranges between 13-30 Digits. For Best Readability, please ensure the following specifications for QR Code Printing:
 - QR Code Version : 2.0 (25x25)
 - QR Code Dimensions: 0.70 x .70 - > 1.0 x 1.0 Inch.
7. Please use following image to print in invoice



8. An integrated retailer to prominently display on each outlet a sign board bearing FBR's official logo along with the text "Integrated with FBR". The registration number of each POS has also to be prominently displayed.



9. The invoice data will be uploaded on FBR Servers by the Software fiscal Component automatically.

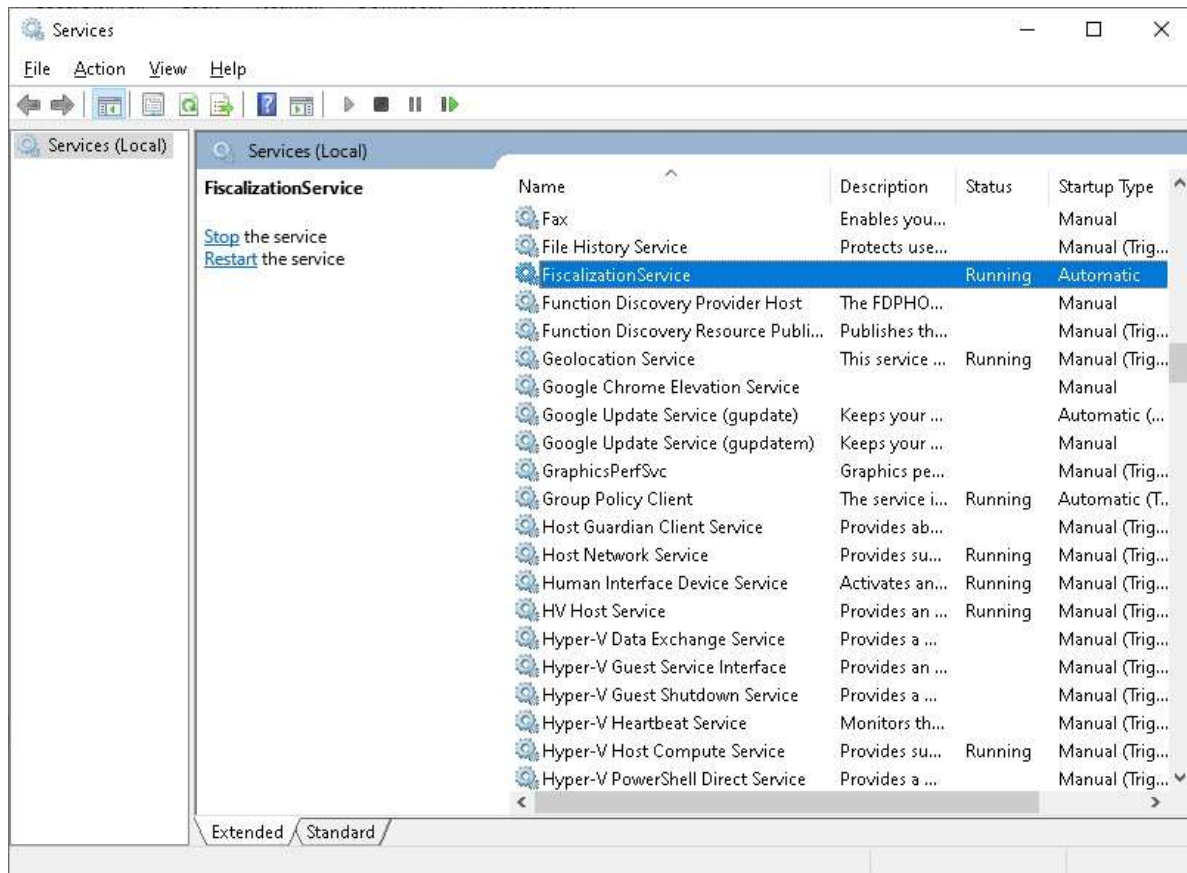


Figure 7 – Fiscalization Service

Installation of the FBR Software Fiscal Component

Following are the pre-requisites for the installation of Software Component.

- Windows Based Operating System (Win 7 and above).
- Internet information services (IIS) must be installed.
- .Net Frame Work 4.5 or above required.
- Installation User must have the administrator rights.

Following are the Steps for the installation of the FBR Software Component.

1. Run the Setup.exe as administrator as shown in **Figure – 8**.
2. Select the complete radio option and click Next as shown in **Figure – 10**.
3. During the installation,
4. setup will prompt for POS Registration Number, Access Code, Test/Productions selection and Target Folder which is issued at the time of POS Registration as shown in **Figure – 11**.
5. Press **“Browse”** button and select the folder in which software component will generate local files.
 - **Note: It is recommended to select the Drive other than C Drive for security purpose.**
6. Provide the required POS Registration No and press **“Install”** button.
7. Click ‘Finish’ to complete the set up.

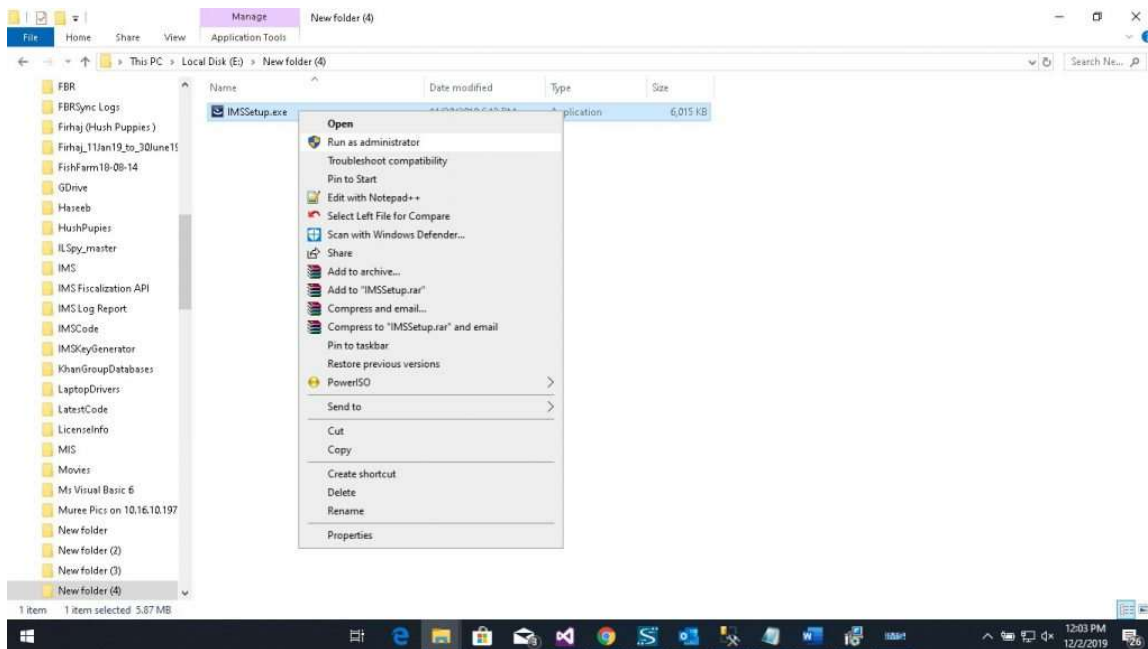


Figure 8 – Set Up

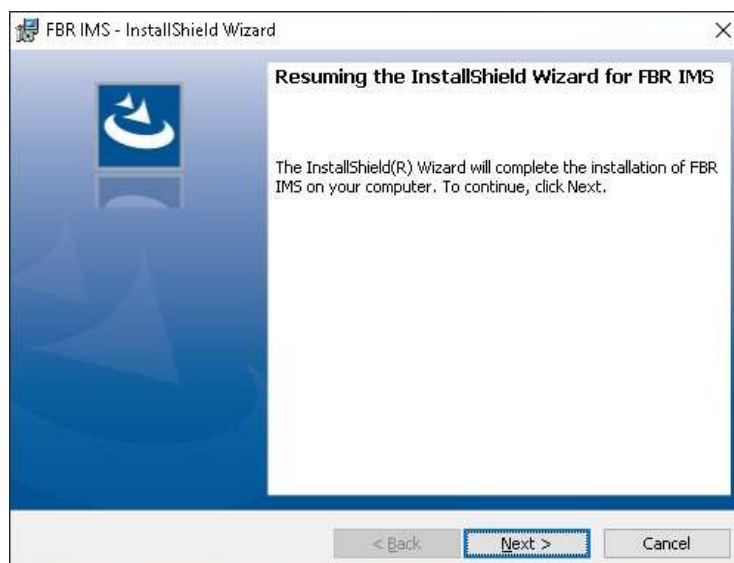


Figure 9 – Start Installation

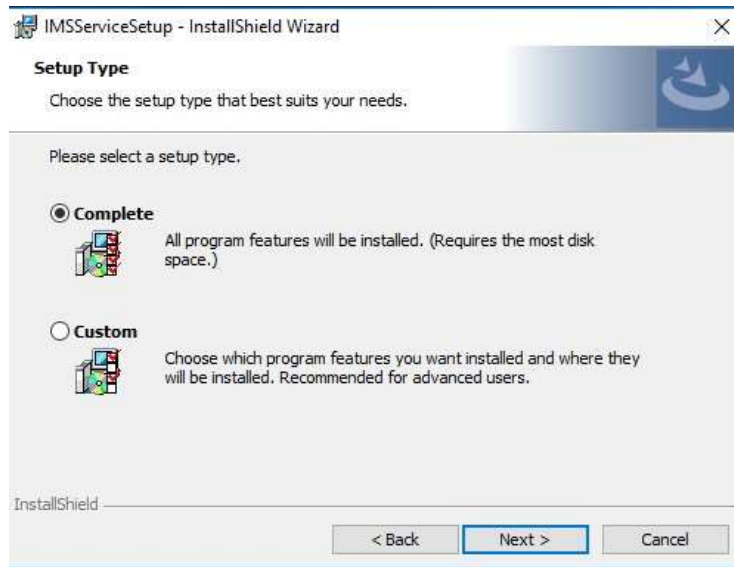


Figure 10 – Installation

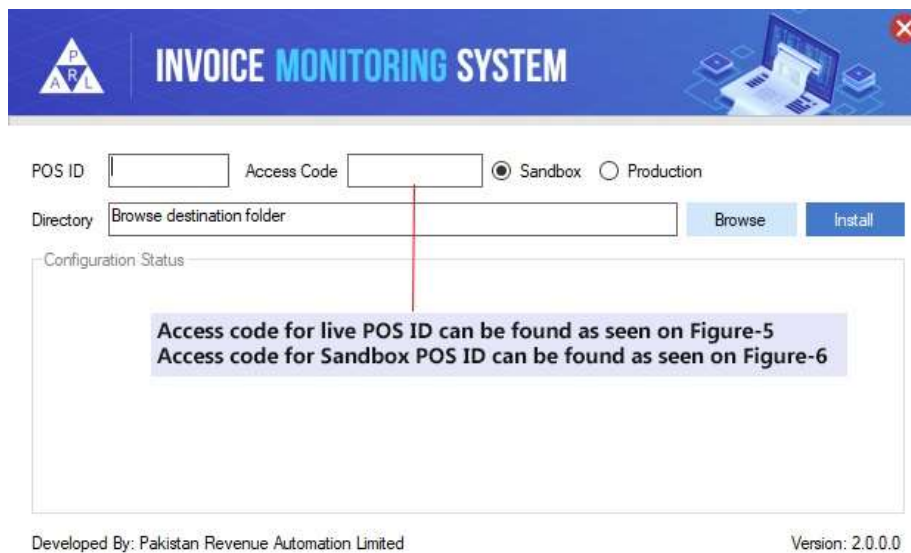


Figure 11 – Registration Number, Access Code, Test/Production selection and Directory

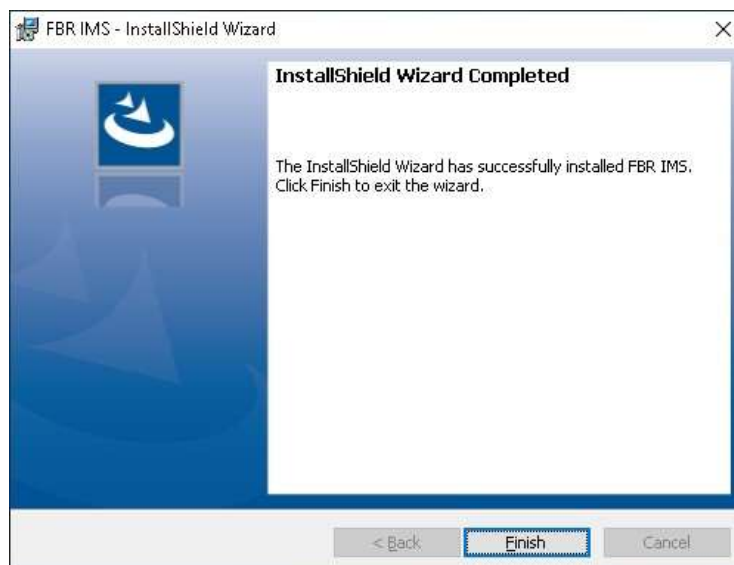


Figure 12 – Installation Completed

Integration of Software Fiscalization Component

Invoice Model Details

The fields of the invoice model are given below.

Name	Data Type	Status	Sample Value
InvoiceNumber	varchar(30)	Blank	
POSID (POS Registration Number given by FBR on registration)	bigint	Compulsory	100000
USIN	varchar(50)	Compulsory	001897
RefUSIN	varchar(50)	Optional	001896
DateTime	datetime	Compulsory	2018-11-01 00:00:00.000
BuyerName	varchar(150)	Optional	Shahzad Ahmed
BuyerNTN	Varchar(9)	Optional	1234567-8
BuyerCNIC	Varchar(13)	Optional	1111122222223
BuyerPhoneNumber	varchar(20)	Optional	923001234567
TotalSaleValue	double	Compulsory	1,298 (Sum of Item Sales Values) Exclusive of Tax and Discount
TotalTaxCharged	double	Compulsory	221 (Sum of Item Tax)
TotalQuantity	double	Compulsory	15 (Count of Total Items)
Discount	double	Optional	380 (Sum of Item Discount)
FurtherTax	double	Optional	0.00 (Sum of Further Tax)
TotalBillAmount	double	Compulsory	1,519
PaymentMode	Int	Compulsory	1. Cash 2. Card 3. Gift Voucher 4. Loyalty Card 5. Mixed 6. Cheque
InvoiceType	Int	Compulsory	1. New 2. Debit 3. Credit
List<Invoiceltems>	list	Compulsory	Table No 2

Invoice Data 1

Table No. 2 (recursive, will contain multiple items record in an invoice)

Name	Data Type	Status	Sample Value
ItemCode	varchar(50)	Compulsory	0001
ItemName	varchar(150)	Compulsory	Blanket
PCTCode	varchar(8)	Compulsory	01011000 (Should be less than or equal to 8)(Pakistan Custom Tariff Code) https://www.fbr.gov.pk/categ/customs-tariff/51149/70853/131188
Quantity	double	Compulsory	1.00
TaxRate	float	Compulsory	17

SaleValue	double	Compulsory	1,298 (Actual Sale Price on which Tax Calculated) Exclusive of Tax and Discount
Discount	double	Optional	380.00
FurtherTax	double	Optional	0.00
TaxCharged	double	Compulsory	221 (Actual Tax)
TotalAmount	double	Compulsory	1,519
InvoiceType	Int	Compulsory	1. New 3. Credit 11. 3rd Schedule New 12. 3rd Schedule Credit (If an item falls under 3rd schedule category then use 11 for new item and use 12 for 3rd schedule item return invoice.)
RefUSIN	Varchar	Optional (In case of Credit or Debit Note)	001897

Invoice Item 1

Return Fiscal Invoice Number

Json Response string from FBR Web API

```
{\"FBRInvoiceNumber\":11000120181112000369,\"Response\": \"Invoice received successfully\", \"Code\": \"100\"}
```

.Net Code Example: Fill the invoice object

```
Invoice objInv = new Invoice();
objInv.InvoiceNumber = string.Empty;
objInv.POSID = 123123;
objInv.USIN = \"123457\";
objInv.DateTime = DateTime.Now;
objInv.BuyerNTN = \"1234567-9\";
objInv.BuyerCNIC = \"12345-1234567-8\";
objInv.BuyerName = \"Buyer Name
objInv.BuyerPhoneNumber = \"0345-1234567\";
objInv.PaymentMode = 1;
objInv.TotalSaleValue = 0;
objInv.TotalQuantity = 0;
objInv.TotalBillAmount = 0;
objInv.TotalTaxCharged = 0;
objInv.Discount = 1000;
objInv.FurtherTax = 100;
objInv.InvoiceType = 1;
```

```

objInv.Items = Items();

private List<InvoiceItems> Items()
{
List<InvoiceItems> lst = new List<InvoiceItems>();
InvoiceItems objItem = new InvoiceItems();
objItem.ItemCode = "0000";
objItem.ItemName = "Item Name";
objItem.Quantity = 3;
objItem.TotalAmount = Convert.ToDouble(3000.00);
objItem.SaleValue = Convert.ToDouble(3180);
objItem.TaxCharged = Convert.ToDouble(180);
objItem.TaxRate = 6;
objItem.PCTCode = "11001010";
objItem.FurtherTax = 20;
objItem.InvoiceType = 1;
objItem.Discount = 500;
lst.Add(objItem);
return lst;
}

```

WebClient to post data to provided URL and get the response

Below code is used to call the WebAPI. JSON string is passed to the function along with API URL to which data is being posted.

```

using System.Net.Http;

using Newtonsoft.Json;

HttpClient Client = new HttpClient();

var content = new StringContent(JsonConvert.SerializeObject(objinvoice), Encoding.UTF8, "application/json");

HttpResponseMessage response = Client.PostAsync("http://localhost:8524/api/IMSFiscal/GetInvoiceNumberByModel",
content).Result;

if (response.IsSuccessStatusCode)
{
Console.WriteLine("Response from API");

Console.WriteLine("-----");

Console.WriteLine(response.Content.ReadAsStringAsync().Result);

}

```

Web Client to post data from Cloud and get the response

Below code is used to call the WebAPI. JSON string is passed to the function along with API URL to which data is being posted.

Sandbox URL:

<https://esp.fbr.gov.pk:8244/FBR/v1/api/Live/PostData>

Production URL:

<https://gw.fbr.gov.pk/imsp/v1/api/Live/PostData>

```
using System.Net.Http;
```

```
using Newtonsoft.Json;
```

```
HttpClient client = new HttpClient();
```

```
client.DefaultRequestHeaders.Authorization = new AuthenticationHeaderValue("Bearer", "1298b5eb-b252-3d97-8622-a4a69d5bf818");
```

```
StringContent content = new StringContent(JsonConvert.SerializeObject(objinvoice), Encoding.UTF8, "application/json");
```

```
System.Net.ServicePointManager.ServerCertificateValidationCallback = delegate { return true; };
```

```
HttpResponseMessage response = client.PostAsync("https://esp.fbr.gov.pk:8244/FBR/v1/api/Live/PostData", content).Result;
```

```
if (response.IsSuccessStatusCode)
```

```
{
```

```
Console.WriteLine("Response from API");
```

```
Console.WriteLine("-----");
```

```
Console.WriteLine(response.Content.ReadAsStringAsync().Result);
```

```
}
```

Sample JSON Format to POST to IMS Component

```
{
  "InvoiceNumber":"","
  "POSID":110014,
  "USIN":"USIN0",
  "DateTime":"2020-01-01 12:00:00",
  "BuyerNTN":"1234567-8",
  "BuyerCNIC":"12345-1234567-8",
  "BuyerName":"Buyer Name",
  "BuyerPhoneNumber":"0000-0000000",
  "TotalBillAmount":0.0,
  "TotalQuantity":0.0,
  "TotalSaleValue":0.0,
  "TotalTaxCharged":0.0,
  "Discount":0.0,
  "FurtherTax":0.0,
  "PaymentMode":1,

  "RefUSIN":null,
  "InvoiceType":1,
  "Items":[
  {
    "ItemCode":"IT_1011",
    "ItemName":"Test Item",
    "Quantity":1.0,
    "PCTCode":"11001010",
    "TaxRate":0.0,
    "SaleValue":0.0,
    "TotalAmount":0.0,
    "TaxCharged":0.0,
    "Discount":0.0,
    "FurtherTax":0.0,
    "InvoiceType":1,
```

```

"RefUSIN":null
},
{
  "ItemCode":"IT_1012",
  "ItemName":"Test Item",
  "Quantity":1.0,
  "PCTCode":"11001010",
  "TaxRate":0.0,
  "SaleValue":0.0,
  "TotalAmount":0.0,
  "TaxCharged":0.0,
  "Discount":0.0,
  "FurtherTax":0.0,
  "InvoiceType":1,
  "RefUSIN":null
}
]}

```

Posting Model in Postman

GET Endpoint

Request:

Open new tab in Postman and write URL: <http://localhost:8524/api/IMSFiscal/Get> . Now click send button to check the response from the Fiscalization Service.

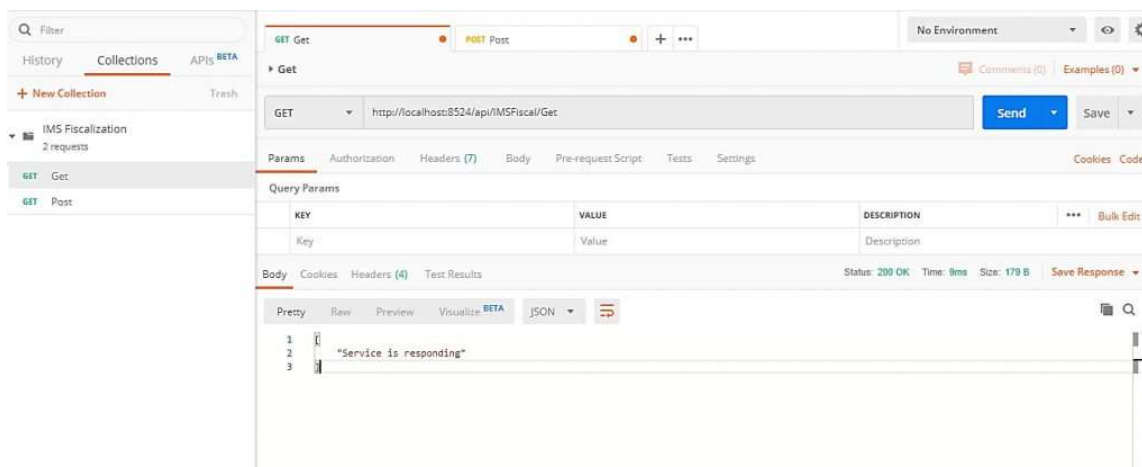


Figure 13 – Check IMS Service Response

Response:

```

[
  "Service is responding"
]

```

POST Endpoint

Request:

Open new tab in Postman and write URL: <http://localhost:8524/api/IMSFiscal/GetInvoiceNumberByModel>

Change type from **Get** to **POST** and then Select **raw** radio button on Postman and change **Text** to **JSON**. Add the sample json mention in section Sample JSON Format to POST to IMS Component.

Now click send button to check the response from the Fiscalization Service.

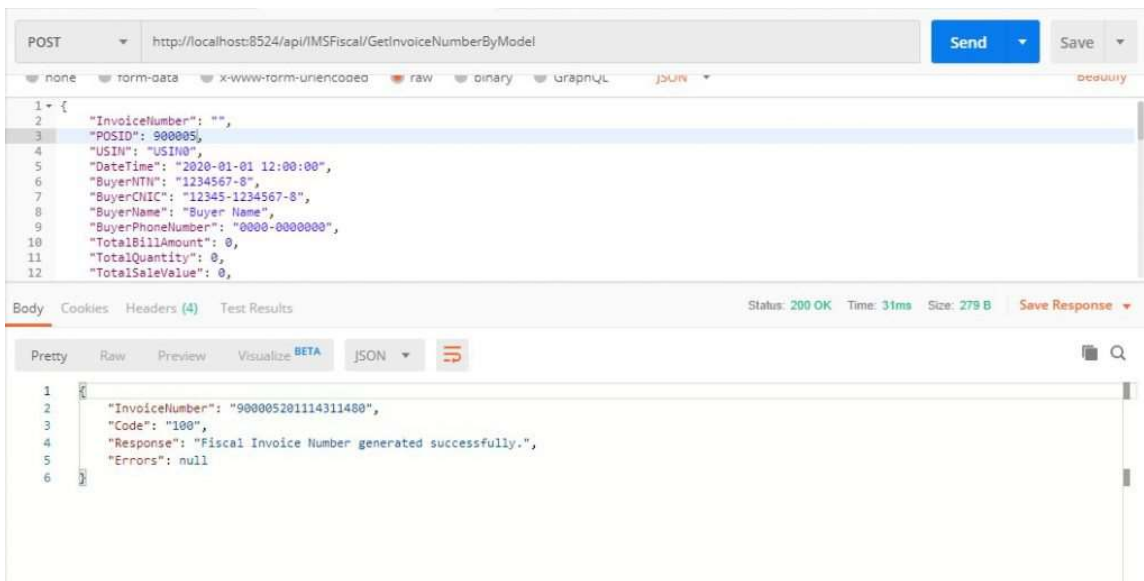


Figure 14 – Post Data to Fiscalization Service

Response:

```

{
  "InvoiceNumber": "9000052011142444901",
  "Code": "100",
  "Response": "Fiscal Invoice Number generated successfully.",
  "Errors": null
}

```

Uninstall

You have to uninstall the sandbox component properly before trying to reinstall the live component. Please do not uninstall IMS component from control panel. To properly uninstall the component, follow these steps:

1. Run the Setup.exe as administrator as shown in **Figure – 8**.
2. The prompt will ask that do you want to perform an upgrade, press "Yes".
3. Click "Next" on wizard screen.
4. In program maintenance screen, select "Remove" radio button. Press Remove button on next wizard screen. It will remove the component properly.
5. Close the wizard by click on "Finish" button. Your component has been uninstalled successfully.

Technical Support

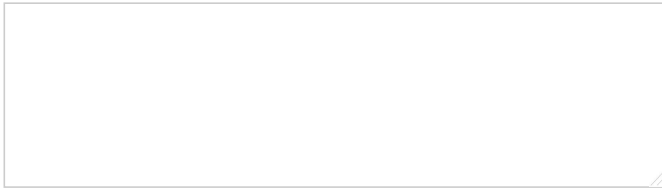
- Email: helpline@fbr.gov.pk
- Phone: (051) 111-772-772

Category: eFBR Portal, How to, Sales Tax, Sales Tax Registration

COMMENTS / SUGGESTIONS

Your email address will not be published. Required fields are marked *

Comment



*Name

Your Name (No Keywords)

*Email

your-real-email@example.com

Post Comment

Copyright @ PRAL 2022 Knowledge Base