INSTALLATION AND USAGE MANUAL FOR ZATCA QRCODE ON SALES INVOICES

pibiCo

Content

1. PibiCut Invoice ERPNext Custom App

1.1. Introduction

This guide explains how to install and use the custom app for ERPNext (pibicut_invoice) to produce QRCode for Sales Invoices as per ZATCA Tax Invoices QRCode Creation Specifications (https://zatca.gov.sa/en/E-Invoicing/SystemsDevelopers/Documents/QRCodeCreation.pdf)

1.2. Installation Procedure

1.2.1. Requirements

It is imperative to have access to the server's command line with enough rights and permissions to install the custom app.

Apart from the above, a Frappe/ERPNext server instance is required (refer to https://github.com/frappe/frappe and https://github.com/frappe/erpnext), and also QR Code (refer to https://github.com/lincolnloop/python-qrcode).

1.2.2. Compatibility

PibiCut_Invoice has been tested and programmed for Frappe/ERPNext version-13 only.

1.2.3. Detailed Installation

From the frappe-bench folder execute the following commands from the command line.

```
$ bench get-app pibicut_invoice https://github.com/pibico/pibicut_invoice.git
$ bench install-app pibicut_invoice
```

If you are using a multi-tenant environment, use the following command

```
$ bench --site site_name install-app pibicut
```

1.2.4. Update

Run updates with:

```
$ bench update
```

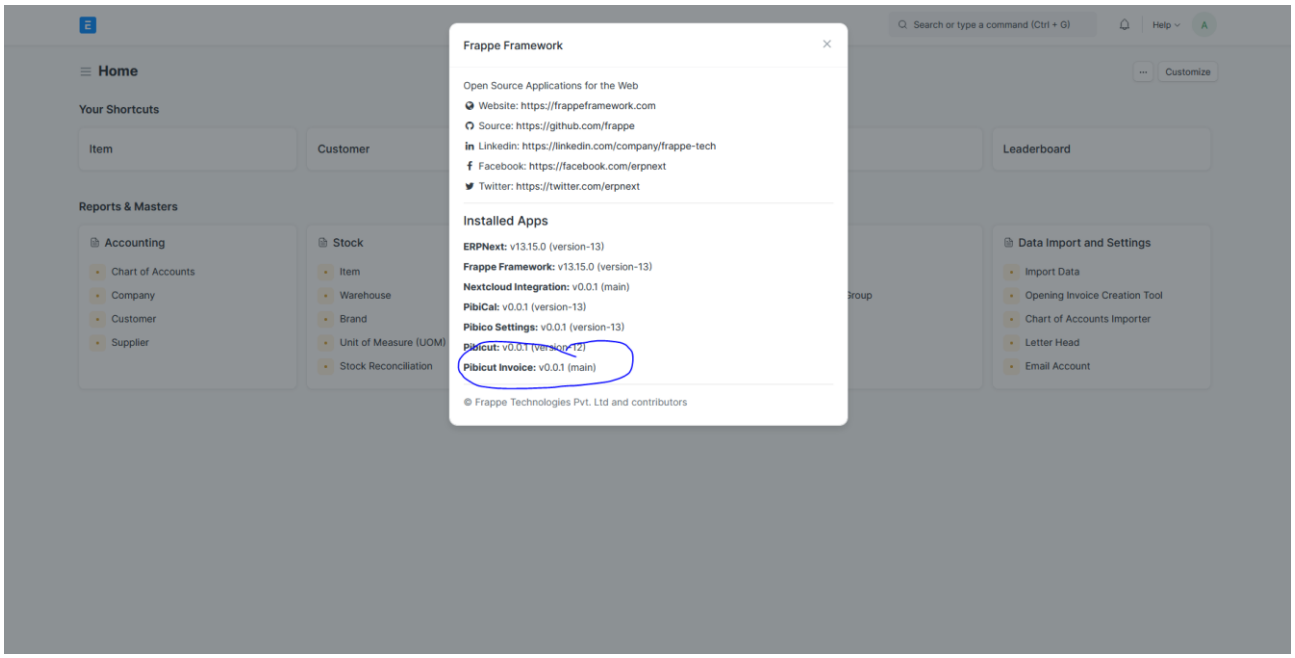
In case you update from the sources and observe errors, make sure to update dependencies with

```
$ bench update --requirements
```

1.2.5. Features

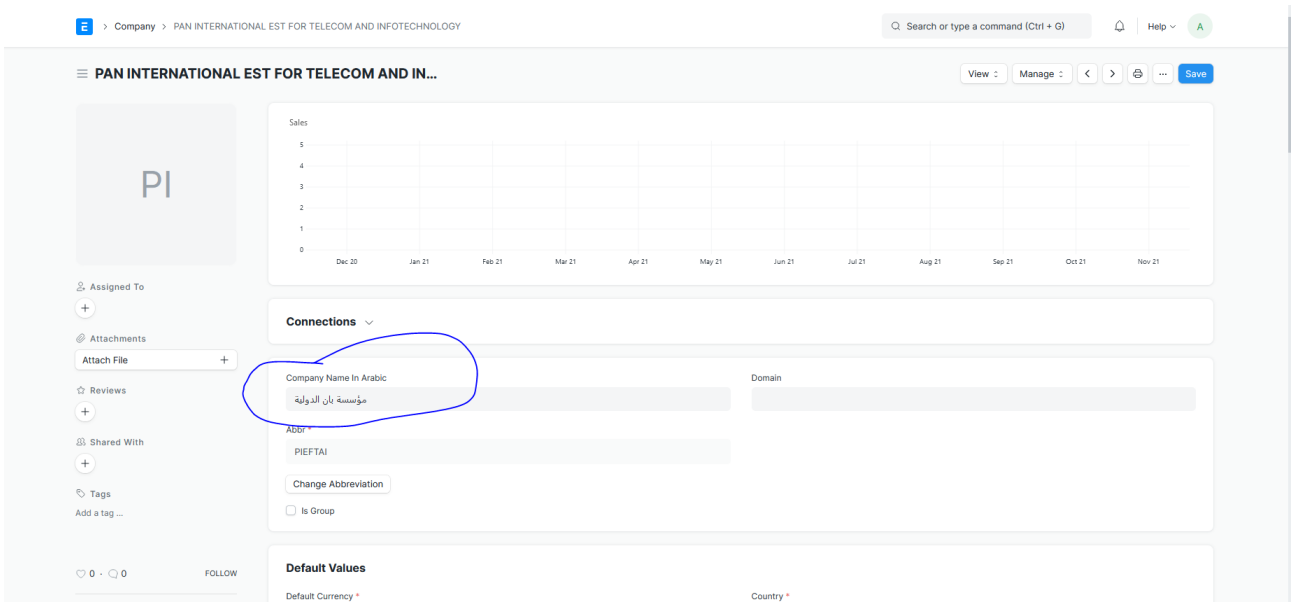
Once installed, new custom fields are created into Sales Invoice Doctype, POS Invoice Doctype and Company Doctype. Those custom fields are related to QR Code encoded in Base64 as per ZATCA Requirements and QR Code Image and QR Preview, to use it on Printed or PDF Sales Invoices or Receipts. (https://zatca.gov.sa/en/RulesRegulations/UnderConsultations/Pages/implementing-E-invoicing.aspx)

In order to check if the custom app is available, you can go to Help > About to see all the Custom Apps installed in your ERPNext Instance



1.3. Usage

First custom field is related to Company Name in Arabic. To fill in this field, you go to **Accounting > Accounting Masters > Company** and Select the Company to name in Arabic. You can fill this custom field. In case that the Company Name in Arabic is not filled in, the QR Code will encode the Standard Company Name.



The following custom fields are related to Sales Invoices. We can go to **Accounting > Accounts Receivable > Sales Invoice** to create a new Sales Invoice.

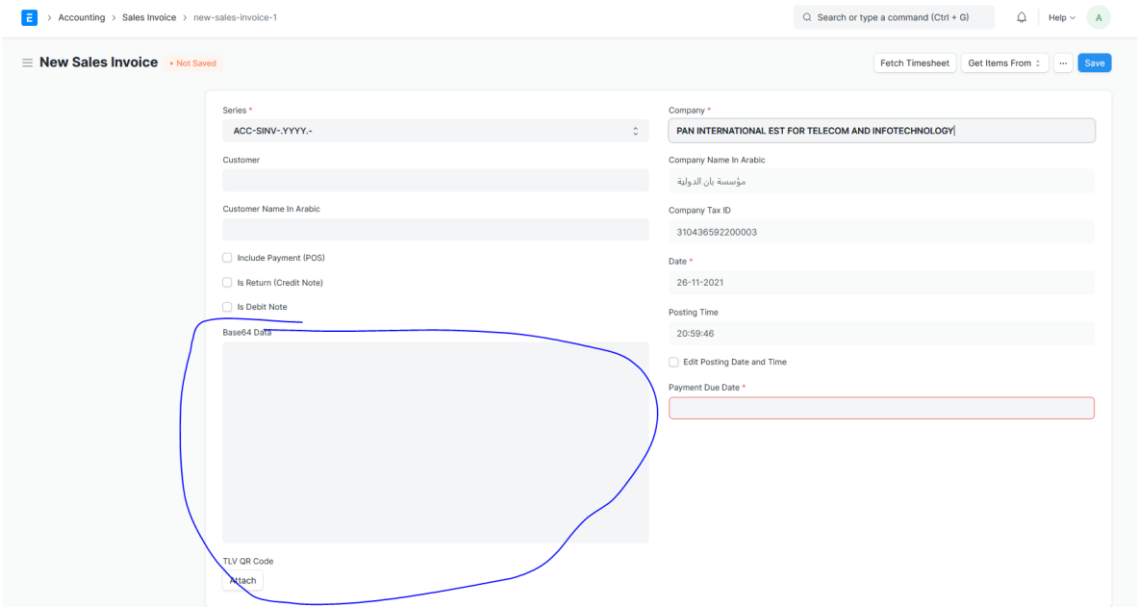
The Company Name in Arabic is shown if it has been filled in. Also the custom fields Base64 Data and TLV QR Code Image are shown. Those custom fields are automatically filled whenever the

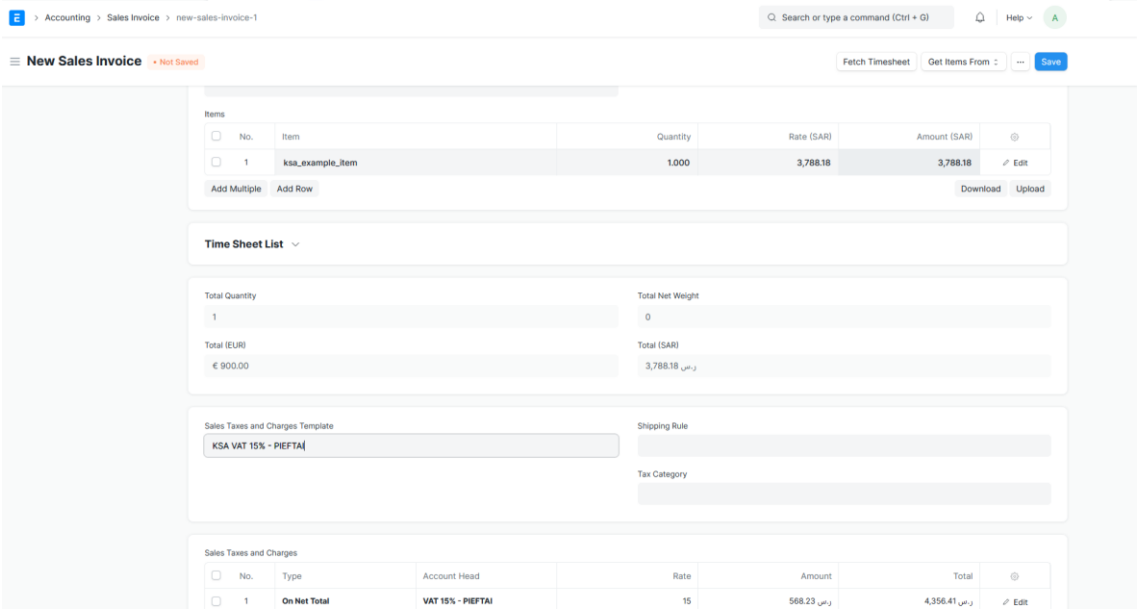
Sales Invoice is saved. The QR Code produced takes into consideration the ZATCA Specifica
filling the following tags with the Sales Invoices Data already saved.

ZATCA QR Code Specifications

Field Definition for the QR Code

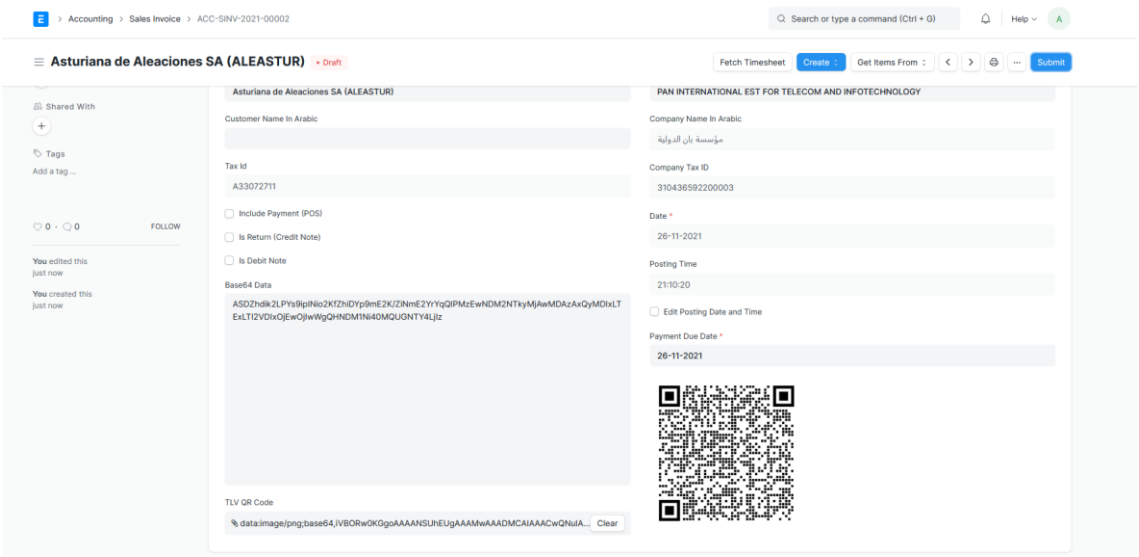
Description	Tag	Due Date
Seller's name	1	4th Dec 2021
VAT registration number of the seller	2	4th Dec 2021
Time stamp of the invoice (date and time)	3	4th Dec 2021
Invoice total (with VAT)	4	4th Dec 2021
VAT total	5	4th Dec 2021
Hash of XML invoice	6	1st Jan 2023
ECDSA signature	7	1st Jan 2023
ECDSA public key	8	1st Jan 2023
For Simplified Tax Invoices and their associated notes, the ECDSA signature of the cryptographic stamp's public key by ZATCA's technical CA	9	1st Jan 2023



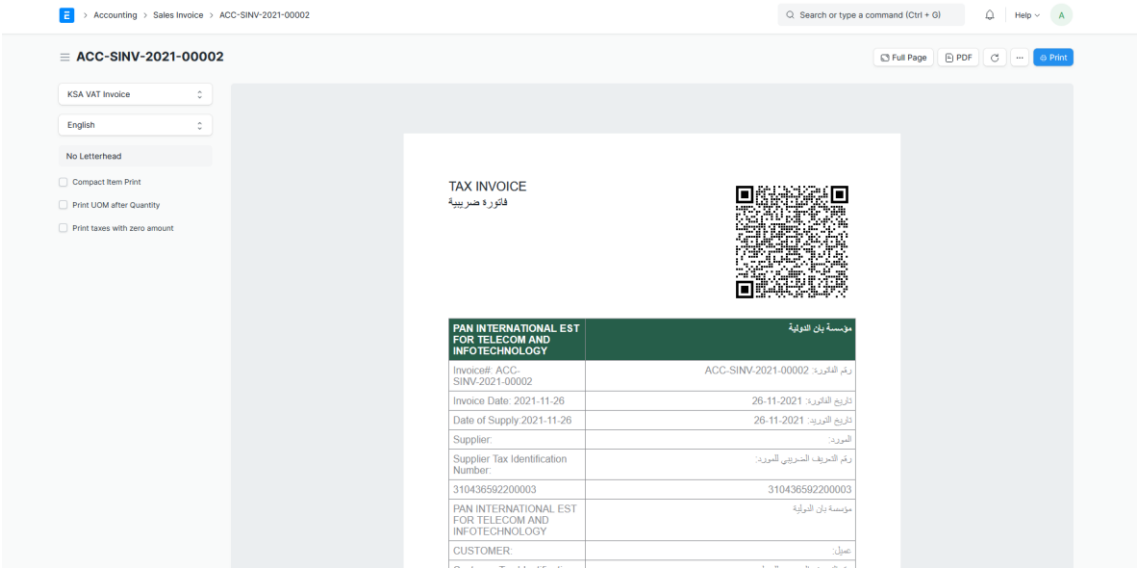


If you want a logo in the middle of the QR Code, you can upload an image file (extension .png, jpg) squared with no more width than 512 pixels and white background (not transparent). In case you upload your logo, the QR Code will show it, in case not logo is provided, the QR Code will not show the centered image.

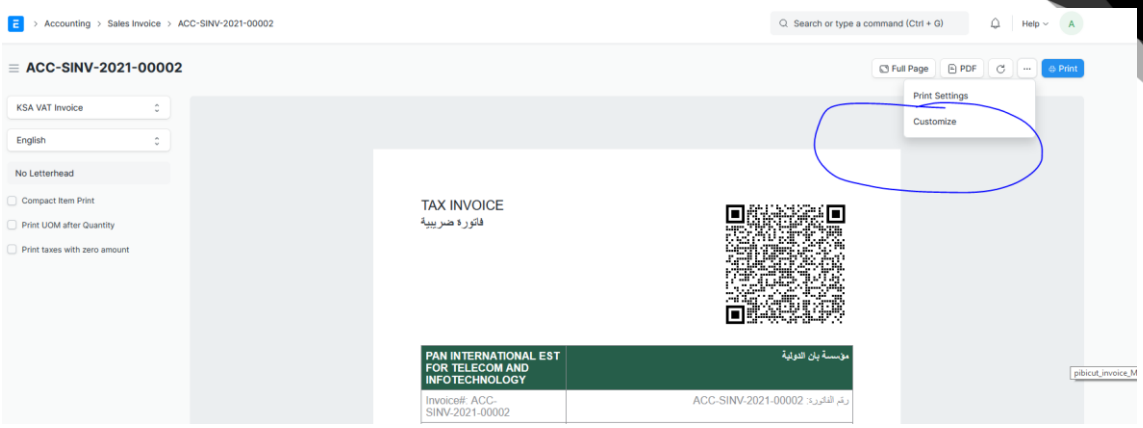
Whenever the Sales Invoice is saved, the QR Code is produced and can be seen attached to the Sales Invoice.



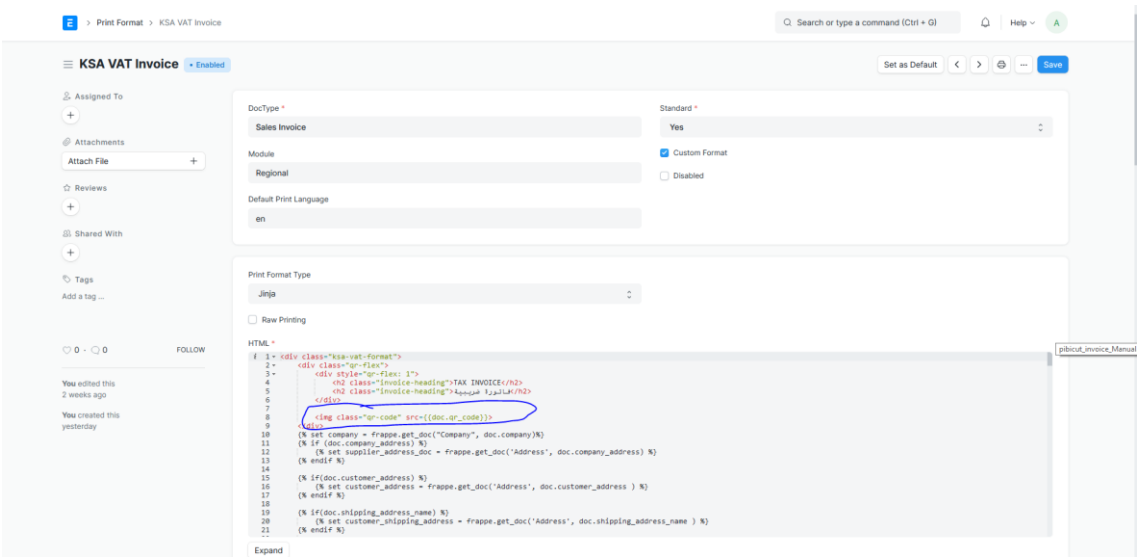
Now, we can print produce the pdf file just printing the Invoice. On the Custom Print Templates we will use ours or just the Standard KSA VAT Invoice.



If we need to attach the generated QR Code to our Custom Print Format for Sales Invoice. For that purpose, we select the Customize Menu to modify the template.



And, in the HTML we can add the piece of code `<img class="qr-code" src={{doc.qr_code}}>` wherever we need to show the QR Code.



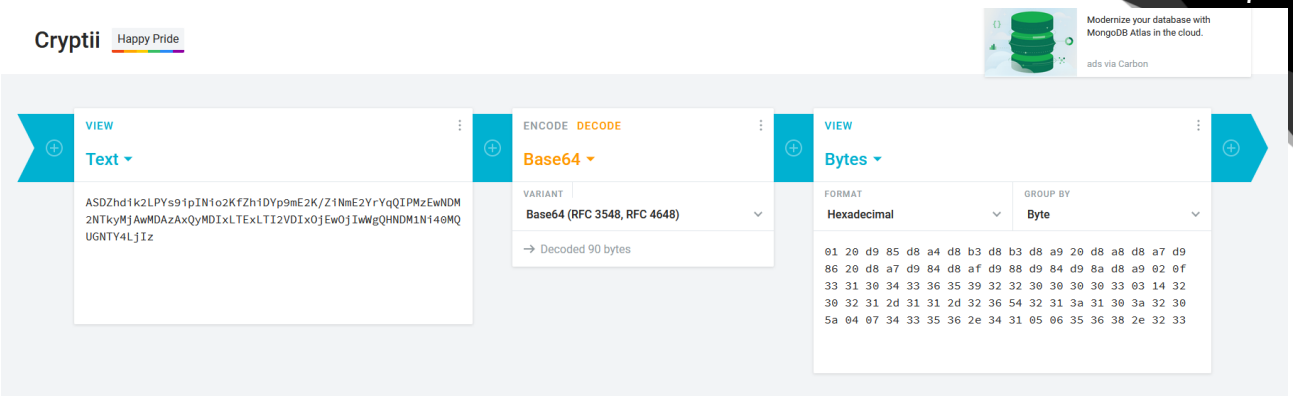
Now, it's time to check if the QRCode is compliant with ZATCA Rules.

First of all, we will read the QR Code and get the Base64 string. This string is the one in the Base64 Data Field.

```
ASDZhdik2LPYs9ipINio2KfZhiDYp9mE2K/ZiNmE2YrYqQIPMzEwNDM2NTkyMjAwMDAzAxQyMDIxLTExLTI2VDIxOjEwOjIwWgQHNDM1Ni40MQUGNTY4LjIz
```

This code we can decode to binary array in https://cryptii.com/ choosing text on first window and pasting the above code.

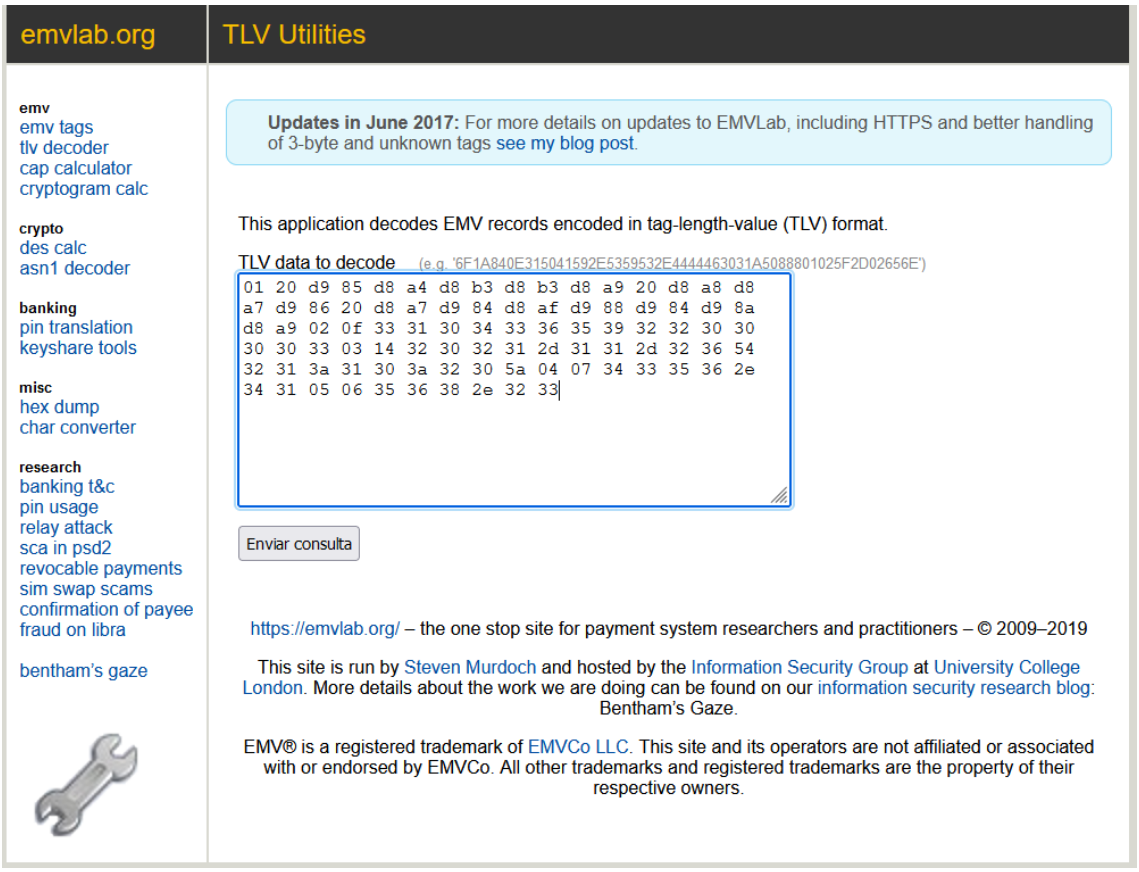
In second window we choose Decode and base64 encoding. And in third window we choose Bytes in hexadecimal group by Byte.



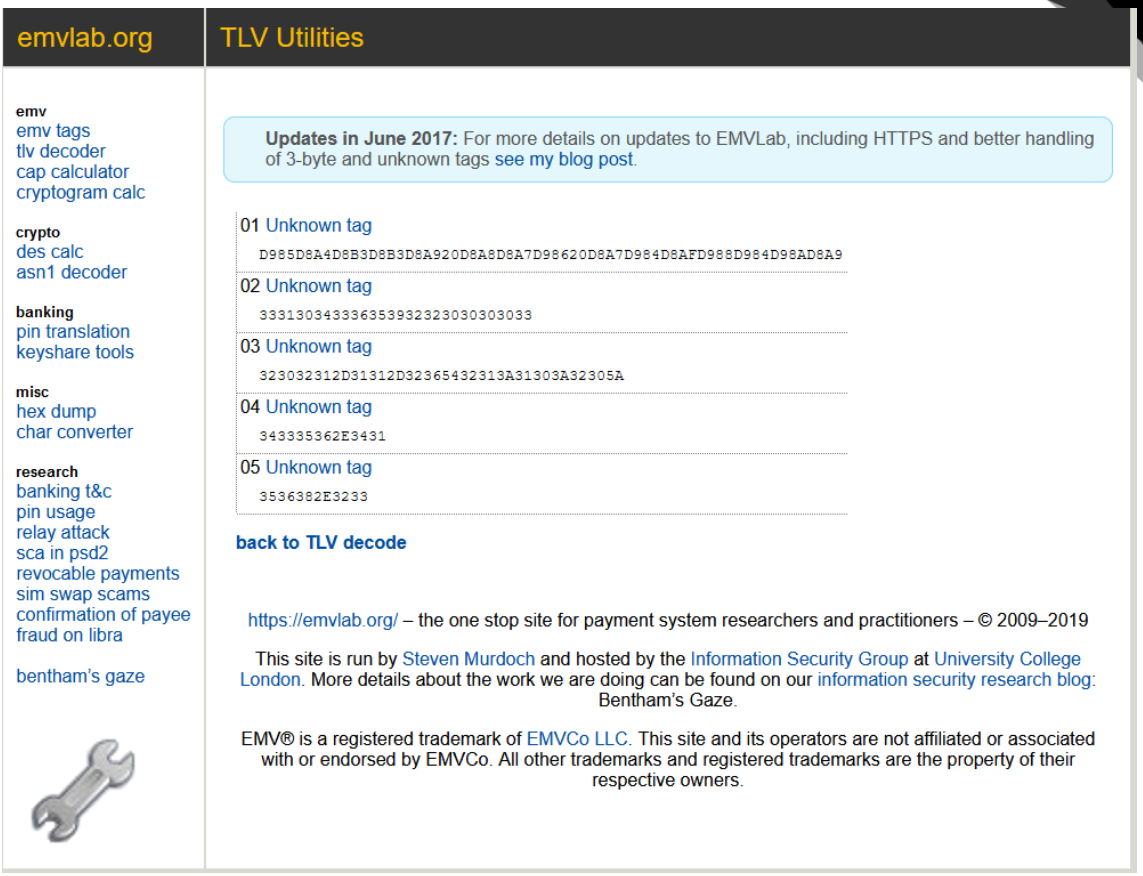
We get the decoded hexadecimal byte array

```
01 20 d9 85 d8 a4 d8 b3 d8 b3 d8 a9 20 d8 a8 d8 a7 d9 86 20 d8 a7 d9 84 d8 af d9 88 d9
84 d9 8a d8 a9 02 0f 33 31 30 34 33 36 35 39 32 32 30 30 30 30 33 03 14 32 30 32 31 2d
31 31 2d 32 36 54 32 31 3a 31 30 3a 32 30 5a 04 07 34 33 35 36 2e 34 31 05 06 35 36 38
2e 32 33
```

That can be read in a TLV reader as the one in https://emvlab.org/tlvutils/

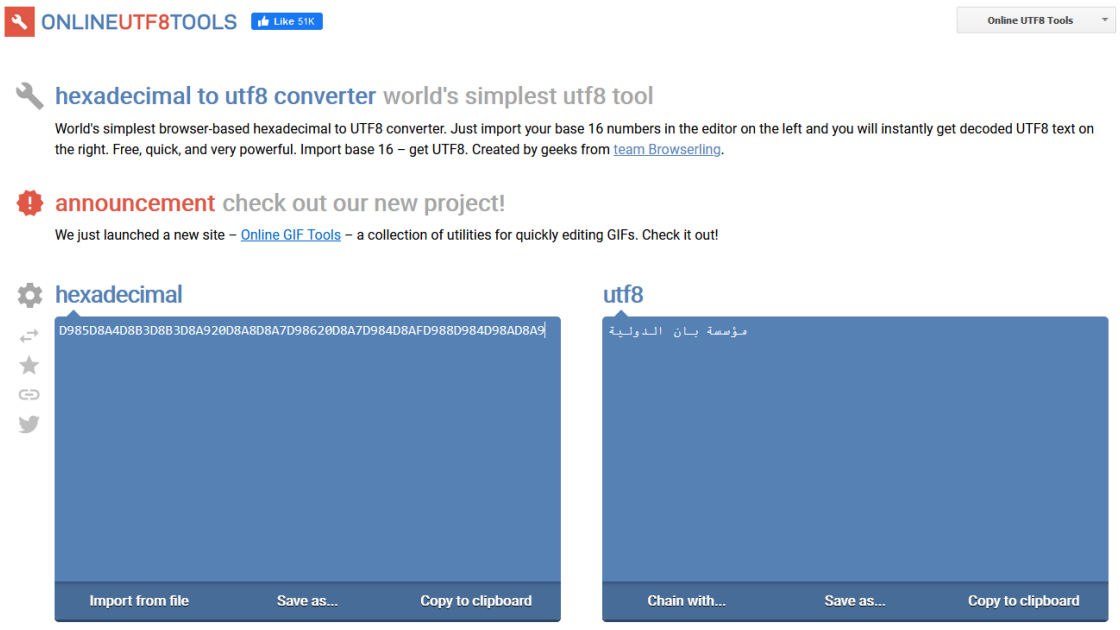


And if is correctly formed we obtain the tags.



We will decode the strings for each tag through https://onlineutf8tools.com/convert-hexadecimal-to-utf8 finally obtaining:

TAG 01 – Seller
D985D8A4D8B3D8B3D8A920D8A8D8A7D98620D8A7D984D8AFD988D984D98AD8A9



TAG 02 – SELLER's VAT NUMBER
333130343336353932323030303033

TAG 03 – INVOICE POSTING DATE
323032312D31312D32365432313A31303A32305A

TAG 04 – INVOICE TOTAL (WITH VAT)
343335362E3431

TAG 05 – VAT
3536382E3233

Thus, it has been demonstrated that the generated QRCode by pibicut_invoice is ZATCA Compliant.