

# FIPAG MAPUTO API REFERENCE

## Technical Specifications

Version 1.0

## Table of Contents

1.	FIPAG maputo RECHARGE API .....	1
2.	Web Services .....	2
2.1.	Generate Token .....	2
2.1.1.	Input Parameters .....	2
2.1.2.	Input Sample .....	2
2.1.3.	Output Fields.....	2
2.1.4.	Output Sample .....	3
2.2.	Get Transaction Details .....	3
2.2.1.	Input Parameters .....	3
2.2.2.	Input sample .....	4
2.2.3.	Output Parameters .....	4
2.2.4.	Output Sample .....	5
2.3.	Process Transaction .....	5
2.3.1.	Input Parameters .....	6
2.3.2.	Input Sample .....	6
2.3.3.	Output Fields.....	6
2.3.4.	Output Sample .....	7
2.4.	Get Last Recharge .....	7
2.4.1.	Input Parameters .....	7
2.4.2.	Input Sample .....	7
2.4.3.	Output Fields.....	8
2.4.4.	Output Sample .....	8
2.5.	Error Codes Description .....	9
2.6.	Reporting and Recon Services.....	10
2.7.	Sample collection .....	10

## 1. FIPAG MAPUTO RECHARGE API

FIPAG MAPUTO provides a set of web services that can be externally used to purchase water recharge for pre-paid meters, as well as consultation of meter details and get last recharge generated.

This document contains technical information that will guide the external entity to use the API and consume the services.

In all API calls a set of mandatory parameters must be considered:

- **eld**

Entity ID is a unique and identifies the company on the request. eld is provided by the owner of the API after registration.

Type is alphanumeric and maximum length is 25.

- **eKey**

Entity Key identifies the type of operation of the request. Each type of operation has his own entity key. Operations can be either create, update, read or delete. This is also provided by the API owner.

Type is alphanumeric and maximum length is 50.

- **userId**

User ID is used to authenticate the application and for logging purpose. The API owner provides user ID. Type is alphanumeric and maximum length is 30.

- **token**

Token is a unique id used to authenticate the request. A token has validity, meaning that expires after a time. Default Expiry time is 5min.

Token should periodically generated by the entity and used before it expires.

Type is alphanumeric and maximum length is 50.

## 2. WEB SERVICES

### 2.1. Generate Token

HTTP Method: GET

Request Timeout: 60s

Endpoint: </fipagmaputo/api/auth/>

#### 2.1.1. Input Parameters

This service only needs the basic authentication parameters: [eld](#), [apiKey](#) and [userId](#).

#### 2.1.2. Input Sample

GET	http://localhost:8080/fipagmaputo/api/auth/key?eld=FIPAG_MAPUTO&eKey=71C10719B10B4EC0B408ACC5E1F82211&userId=apiuser	
Params	Authorization	Headers (8)
Query Params		
	KEY	VALUE
<input checked="" type="checkbox"/>	eld	FIPAG_MAPUTO
<input checked="" type="checkbox"/>	eKey	71C10719B10B4EC0B408ACC5E1F82211
<input checked="" type="checkbox"/>	userId	apiuser
	Key	Value

#### 2.1.3. Output Fields

Format: JSON

[tokenId](#) – alphanumeric.

[expiryDate](#) – datetime (Format yyyy-MM-dd HH:m).

[errorCode](#) – generate token can be null, meaning there is no error or can return messages described in error section (see details in [Error Codes Description](#)):

#### 2.1.4. Output Sample

```
Pretty Raw Preview Visualize JSON ↺
1 {
2   "tokenId": "4daa5485-14ec-48eb-8183-2fc90e527602",
3   "expiryDate": "2021-09-08 00:50",
4   "errorCode": null
5 }
```

## 2.2. Get Transaction Details

This service returns all transactions details such as commissions, debts, water units and customer details.

Method: GET

Request timeout: 60s

Endpoint: </fipagmaputo/api/recharge/simulate>

### 2.2.1. Input Parameters

Authentication parameters: [eld](#), [apiKey](#) and [userId](#).

[meterNumber](#) – meter number already registered on FIPAG MAPUTO. Type is alphanumeric and must have 11 or 13 characters.

[amount](#) – total amount to be converted in water unit or quantity of water. Type is Numeric. The precision must be to two decimal places and separated by a dot.

[token](#) – previous got through the service [Generate Token](#).

### 2.2.2. Input sample

GET

http://localhost:8080/fipagmaputo/api/recharge/simulate?eld=FIPAG\_MAPUTO&eKey=71C10719B10B4EC0B408ACC5E1F82211&userId=apiuser&meterNumber=0120027001102&amount=1000&token=FBD714DC-3F43-4558-BC2D-B2001F87BF42

Params

Authorization

Headers (8)

Body

Pre-request Script

Tests

Settings

Query Params

	KEY	VALUE	DESCRIPTION
<input checked="" type="checkbox"/>	eld	FIPAG_MAPUTO	
<input checked="" type="checkbox"/>	eKey	71C10719B10B4EC0B408ACC5E1F82211	
<input checked="" type="checkbox"/>	userId	apiuser	
<input checked="" type="checkbox"/>	meterNumber	0120027001102	
<input checked="" type="checkbox"/>	amount	1000	
<input checked="" type="checkbox"/>	token	FBD714DC-3F43-4558-BC2D-B2001F87BF42	

### 2.2.3. Output Parameters

Format: JSON

Encoding: UTF-8

[availabilityService](#) – Availability service amount. Type is numeric.

[category](#) – category of the customer, possible values are: Comercial, Doméstico, Industrial, Fontenário, Municipal and Público)

[customerAddress](#) – address where the meter is placed. Type is alphanumeric

[customerDistrict](#) – district where the meter is placed. Type is alphanumeric

[customerName](#) – name of the owner of the meter. Type is alphanumeric

[debtAmount](#) – total debt amount to be discounted on purchase. Type numeric.

[maxUnitAllowed](#)- maximum units allowed for the customer. This field is combined with errorCode MAX\_UNIT\_REACHED. Type is numeric.

[meterNumber](#) – meter number provided in the request. Type is alphanumeric

[minAmountRequired](#) – minimum required amount to proceed with the transaction. This field is combined with errorCode MIN\_AMOUNT\_REQUIRED). Type is Numeric.

[paidAmount](#) – amount paid. Provided in the request. Type is numeric.

[scale](#) – last scale of the purchase. Type is alphanumeric.

[simulationId](#) – unique id, to be used for to commit the transaction. Type is alphanumeric. Maximum length is 50.

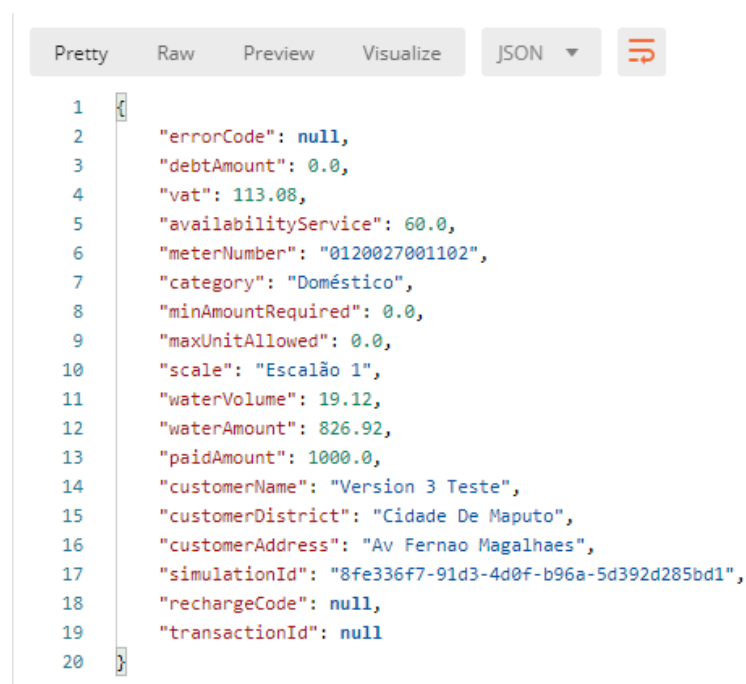
**vat** – VAT amount. Type is numeric.

**waterAmount** – remaining amount to be converted to water after all discounts. Type is numeric.

**waterVolume** – total water volume. Type is numeric.

**errorCode** – this field comes with value if there is an error. Possible errors: (See details in [Error Codes Description](#))

#### 2.2.4. Output Sample

A screenshot of a code editor showing a JSON object. The editor has tabs for 'Pretty', 'Raw', 'Preview', and 'Visualize', with 'Pretty' selected. A dropdown menu shows 'JSON' and a button with a red icon. The JSON object is formatted with line numbers 1 through 20 on the left. The object contains fields for error handling, payment, and customer information.

```
1 {
2   "errorCode": null,
3   "debtAmount": 0.0,
4   "vat": 113.08,
5   "availabilityService": 60.0,
6   "meterNumber": "0120027001102",
7   "category": "Doméstico",
8   "minAmountRequired": 0.0,
9   "maxUnitAllowed": 0.0,
10  "scale": "Escalaõ 1",
11  "waterVolume": 19.12,
12  "waterAmount": 826.92,
13  "paidAmount": 1000.0,
14  "customerName": "Version 3 Teste",
15  "customerDistrict": "Cidade De Maputo",
16  "customerAddress": "Av Fernao Magalhaes",
17  "simulationId": "8fe336f7-91d3-4d0f-b96a-5d392d285bd1",
18  "rechargeCode": null,
19  "transactionId": null
20 }
```

### 2.3. Process Transaction

This service is responsible to generate the recharge code.

HTTP Method: POST

Request timeout: 60s

Format: JSON

Encoding: UTF-8

Endpoint: </fipagmaputo/api/recharge/generate>

### 2.3.1. Input Parameters

Authentication parameters: [eld](#), [apiKey](#) and [userId](#)

[simulationId](#) – unique id previously obtained by get transaction details web service.

[token](#) – token previously obtained by generate token web service.

### 2.3.2. Input Sample



### 2.3.3. Output Fields

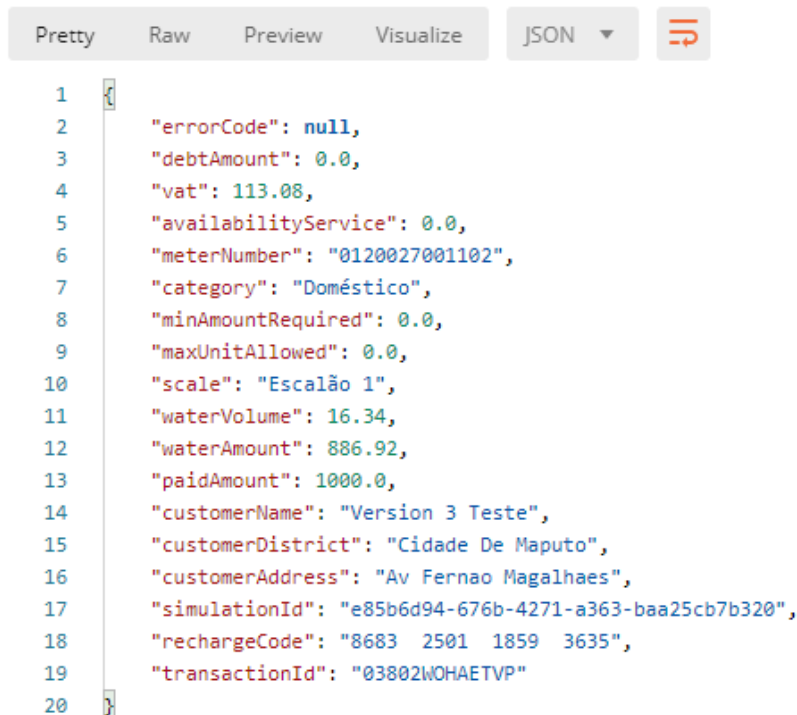
Process Transaction output contains same fields as get transaction details but two fields:

[rechargeCode](#) – the actually code to be introduced in a meter. Type is alphanumeric, maximum length is 29.

[transactionId](#) – unique ID that identifies the transaction. Type is alphanumeric, maximum length is 25.



### 2.3.4. Output Sample



The image shows a JSON output sample in a code editor. The editor has tabs for 'Pretty', 'Raw', 'Preview', and 'Visualize', with 'Pretty' selected. A dropdown menu shows 'JSON' and a refresh icon. The JSON is formatted and spans 20 lines. It contains fields for error code, debt amount, VAT, availability service, meter number, category, minimum and maximum unit allowed, scale, water volume and amount, paid amount, customer name and address, simulation ID, recharge code, and transaction ID.

```
1 {
2   "errorCode": null,
3   "debtAmount": 0.0,
4   "vat": 113.08,
5   "availabilityService": 0.0,
6   "meterNumber": "0120027001102",
7   "category": "Doméstico",
8   "minAmountRequired": 0.0,
9   "maxUnitAllowed": 0.0,
10  "scale": "Escalão 1",
11  "waterVolume": 16.34,
12  "waterAmount": 886.92,
13  "paidAmount": 1000.0,
14  "customerName": "Version 3 Teste",
15  "customerDistrict": "Cidade De Maputo",
16  "customerAddress": "Av Fernao Magalhaes",
17  "simulationId": "e85b6d94-676b-4271-a363-baa25cb7b320",
18  "rechargeCode": "8683 2501 1859 3635",
19  "transactionId": "03802WOHAETVP"
20 }
```

## 2.4. Get Last Recharge

This service will return the last generated recharge code. Note that the service return recharges generated within a month.

Method: GET

Request timeout: 60s

Endpoint: </fipagmaputo/api/recharge/last>

### 2.4.1. Input Parameters

Authentication parameters: [eld](#), [apiKey](#), and [userId](#).

[token](#) – token previously obtained by Generate Token web service

[meterNumber](#) – meter number already registered on FIPAG MAPUTO. Type is alphanumeric and must have 11 or 13 characters.

### 2.4.2. Input Sample

GET	http://localhost:8080/fipagmaputo/api/recharge/last?eld=FIPAG_MAPUTO&eKey=71C10719B10B4EC0B408ACC5E1F82211&userId=apiuser&meterNumber=0120027001102&token=CC99F695-BA9E-49F0-818E-3A464325915A		
Params	Authorization	Headers (8)	Body
Pre-request Script	Tests	Settings	
Query Params			
KEY	VALUE	DESCRIPTION	
<input checked="" type="checkbox"/> eld	FIPAG_MAPUTO		
<input checked="" type="checkbox"/> eKey	71C10719B10B4EC0B408ACC5E1F82211		
<input checked="" type="checkbox"/> userId	apiuser		
<input checked="" type="checkbox"/> meterNumber	0120027001102		
<input checked="" type="checkbox"/> token	CC99F695-BA9E-49F0-818E-3A464325915A		

### 2.4.3. Output Fields

**meterNumber** – meter number from the request. Type is alphanumeric.

**rechargeCode** – last recharge generated. Type is alphanumeric.

**transactionDate** – transaction date time. Format is yyyy-MM-dd HH:m

**transactionId** - unique identifier of the transaction. Type is Alphanumeric

**paidAmount** - amount paid. Type is Numeric.

**errorCode** - this field comes with value if there is an error. Possible errors: (See details in [Error Codes Description](#))

### 2.4.4. Output Sample

```

Pretty  Raw  Preview  Visualize  JSON
{
  "meterNumber": "0120027001102",
  "rechargeCode": "8683 2501 1859 3635",
  "transactionDate": "2021-09-09 18:10",
  "transactionId": "03802WOHAETVP",
  "paidAmount": 1000.0,
  "errorCode": null
}

```

## 2.5. Error Codes Description

Code	Description
DEBT_TOO_HIGH	The debt amount is too high, customer cannot buy water through web service
GENERIC_ERROR	Generic Exception
INVALID_METER	The meter does not exists in FIPAG MAPUTO database
INVALID_SIMULATION_ID	Simulation id is not valid
INVALID_TOKEN	Token is invalid or expired
MAX_UNIT_REACHED	The requested amount to buy exceeded max allowed units after conversion
MIN_AMOUNT_REQUIRED	The requested amount to buy is too low for the purchase
NO_TRANSACTION_FOUND	The transaction requested does not exists or is no longer available in live
SIMULATION_ALREADY_USED	The simulation id in the request has already used
INVALID_ENTITY	Entity Id provided in request is invalid
APP_DISABLED	Entity is disabled by the API Owner
USER_NOT_FOUND	User ID provided in request is invalid
INVALID_USER_TYPE	User ID provided cannot make requests through API
INVALID_PARAMETERS_LENGTH	Parameter length is invalid
EMPTY_PARAMETER	Empty parameter provided in request
INVALID_KEY	Entity Key provided is invalid
NUMBER_MALFORMED	The amount in request is not well formatted
INVALID_AMOUNT	Provided amount is not numeric
METER_NUMBER_MALFORMED	The provider meter number has bad format

## 2.6. Reporting and Recon Services

FIPAG provides transaction reports periodically. These reports that will serve for reconciliation can be sent daily, weekly or monthly.

The choice of the reception base depends on the will of the entity that will consume the service.

The report will be sent to the entity's email in csv format and contain the fields below:

`transactionId`, `meterNumber`, `customerName`, `customerDistrict`, `paidAmount`, `waterUnits`, `transactionDate` and `transactionStatus`.

## 2.7. Sample collection

With this document is attached a sample collection, so that all the services can be imported in a REST Client.

File name: FIPAG API.postman\_collection.json