

# Demostración funcional y visual del desarrollo para el tema 2

En relación con I002 – Tema 2 (#2)

## Contenido

Introducción .....	2
Capturas de pantalla .....	2
CRM .....	2
IoT .....	7

# Introducción

En este documento se encuentra la demostración visual (capturas de pantalla) del funcionamiento de los endpoints desarrollados para los sistemas CRM mini e IoT mini.

Las demostraciones se han realizado usando Colecciones de Postman como herramienta para automatizar las peticiones. A continuación, se presentan una serie de pruebas que cuentan con una pequeña explicación cuando sea necesario.

## Capturas de pantalla

### CRM

- Entorno de demostración (Postman Collection)

The screenshot shows the Postman interface with the following details:

- Collection:** Pruebas CRM-IoT
- Run Details:** Ran today at 12:24:02 PM · View all runs
- Iteration 1:** 1 iteration, Duration: 920ms, All tests: 0, Avg. Resp. Time: 18 ms
- Test Cases:** All Tests · Passed (0) · Failed (0) · Skipped (0) · View Summary
- API Requests:**
  - GET CRM / Clientes 200 OK (http://localhost:8001/clientes) · 200 · 58 ms · 5.529 KB
  - GET CRM / Clientes por nombre 200 OK (http://localhost:8001/clientes?q=Consultoria y Auditoria) · 200 · 5 ms · 398 B
  - GET CRM / Paginación 200 OK (http://localhost:8001/clientes?page=2&pageSize=3) · 200 · 8 ms · 891 B
  - GET CRM / Clientes por ubicación 200 OK (http://localhost:8001/clientes?ubicacionId=Calle Real 45, Valencia) · 200 · 4 ms · 398 B
  - GET CRM / Clientes por id 200 OK (http://localhost:8001/clientes/P006) · 200 · 51 ms · 354 B

- Clientes: Podemos observar que obtenemos todos los clientes (contamos con 200 clientes generados por IA) que son expuestos en rondas de 25 por el parámetro default de paginación.

The screenshot shows the Postman interface with a successful API call. The URL is `GET {{crmPythonURL}} /clientes`. The response body is a JSON object representing a page of clients:

```

1  {
2    "total": 200,
3    "page": 1,
4    "pageSize": 25,
5    "data": [
6      {
7        "id": "C001",
8        "nombre": "Acme Suministros S.L.",
9        "tipo": "cliente",
10       "direccion": "Calle Falsa 123, Madrid",
11       "nif": "B12345678",
12       "correo_electronico": "acme.suministros@example.com",
13       "numero_telefono": "+34 91 123 4567",
14       "transacciones_detalladas": []
15     },
16     {
17       "id": "P001",
18       "nombre": "GlobalTech Proveedores",
19     }
20   ]
21 }

```

- Clientes por nombre

The screenshot shows the Postman interface with a successful API call. The URL is `GET {{crmPythonURL}} /clientes?q=Consultoría y Auditoría`. The response body is a JSON object representing a page of clients filtered by name:

```

1  {
2    "total": 1,
3    "page": 1,
4    "pageSize": 25,
5    "data": [
6      {
7        "id": "P006",
8        "nombre": "Consultoría y Auditoría",
9        "tipo": "proveedor",
10       "direccion": "Calle Real 45, Valencia",
11       "nif": "A40302010",
12       "correo_electronico": "auditoria@consul.net",
13       "transacciones_detalladas": [
14         "S-MOV-ENT-09",
15         "S-ENERGIA-01"
16       ]
17     }
18   ]
19 }

```

- Paginación en consulta

Pruebas CRM-IoT / CRM / Página 200 OK

GET {{crmPythonURL}} /clientes?page=2&pageSize=3

Key	Value	Description
page	2	
pageSize	3	

```

1 {
  "total": 200,
  "page": 2,
  "pageSize": 3,
  "data": [
    {
      "id": "P002",
      "nombre": "Materias Primas del Sur",
      "tipo": "proveedor",
      "direccion": "Polígono Industrial, Sevilla",
      "nif": "A11223344",
      "correo_electronico": "primosur@materiales.net",
      "numero_telefono": "+34654789012",
      "transacciones_detalladas": [
        "S-PH-05"
      ]
    }
  ]
}

```

- Ubicación como filtro

Pruebas CRM-IoT / CRM / Clientes por ubicación 200 OK

GET {{crmPythonURL}} /clientes?ubicacionId=Calle Real 45, Valencia

Key	Value	Description
ubicacionId	Calle Real 45, Valencia	

```

1 {
  "total": 1,
  "page": 1,
  "pageSize": 25,
  "data": [
    {
      "id": "P006",
      "nombre": "Consultoría y Auditoría",
      "tipo": "proveedor",
      "direccion": "Calle Real 45, Valencia",
      "nif": "A40302010",
      "correo_electronico": "auditoria@consul.net",
      "transacciones_detalladas": [
        "S-MOV-ENT-09",
        "S-ENERGIA-01"
      ]
    }
  ]
}

```

- ID como filtro

The screenshot shows the Postman interface with a collection named "Pruebas CRM-IoT / CRM". A specific request titled "GET Clientes por id 200 OK" is selected. The URL is set to `{{crmPythonURL}}/clientes/P006`. The "Params" tab is active, showing a single parameter "Key" with value "P006". The "Body" tab shows a JSON response with the following content:

```

1 {
2     "id": "P006",
3     "nombre": "Consultoria y Auditoria",
4     "tipo": "proveedor",
5     "direccion": "Calle Real 45, Valencia",
6     "nif": "A40302010",
7     "correo_electronico": "auditoria@consul.net",
8     "transacciones_detalladas": [
9         "S-MOV-ENT-09",
10        "S-ENERGIA-01"
11    ]
12 }

```

The response status is 200 OK, with a duration of 7 ms and a size of 354 B. The response body is displayed in JSON format.

- Comprobación de errores: Hasta el momento, hemos realizado peticiones exitosas (código 200). Ahora toca validar que las peticiones no exitosas devuelven códigos de respuesta acordes a su error.

- ID como filtro (404)

The screenshot shows the Postman interface with the same collection. A request titled "GET Clientes por id 404 Not Found" is selected. The URL is set to `{{crmPythonURL}}/clientes/id_inexistente`. The "Params" tab is active, showing a single parameter "Key" with value "id\_inexistente". The "Body" tab shows a JSON response with the following content:

```

1 {
2     "detail": "Cliente no encontrado"
3 }

```

The response status is 404 Not Found, with a duration of 7 ms and a size of 166 B. The response body is displayed in JSON format.

- Parámetro no válido (400)

Screenshot of the Postman application interface showing a failed API request.

The URL entered is: `GET {{crmPythonURL}}/clientes?pageSize=200`

The response status is: **400 Bad Request**

The response body is:

```
1 {  
2     "detail": [  
3         {  
4             "type": "less_than_equal",  
5             "loc": [  
6                 "query",  
7                 "pageSize"  
8             ],  
9             "msg": "Input should be less than or equal to 100",  
10            "input": "200",  
11            "ctx": {  
12                "le": 100  
13            }  
14        }  
15    ],  
16    "body": null  
17}
```

# IoT

- Entorno de demostración (Postman Collection)

The screenshot shows the Postman interface with the following details:

- Collections:** Pruebas CRM-IoT
- Environments:** none
- Iterations:** 1
- Duration:** 1s 356ms
- All tests:** Passed (0) Failed (0) Skipped (0)
- Iteration 1:**
  - GET /Sensores 200 OK**: http://localhost:8002/sensores. Response: 200 • 25 ms • 2.636 KB. No tests found.
  - GET /Sensores por tipo 200 OK**: http://localhost:8002/sensores?tipo=temperatura. Response: 200 • 6 ms • 753 B. No tests found.
  - GET /Sensores por ubicación 200 OK**: http://localhost:8002/sensores?ubicacionId=Sala%20de%20Servidores%201. Response: 200 • 4 ms • 583 B. No tests found.
  - GET /Lecturas 200 OK**: http://localhost:8002/lecturas. Response: 200 • 71 ms • 8.183 KB. No tests found.
  - GET /Lecturas por sensor 200 OK**: http://localhost:8002/lecturas/sensorId=S-TEMP-04. Response: 200 • 7 ms • 828 B. No tests found.
- Cloud View**, **Find and replace**, **Console**
- Runner**, **Start Proxy**, **Cookies**, **Vault**, **Trash**

- Sensores

The screenshot shows the Postman interface with the following details:

- Collections:** Pruebas CRM-IoT
- Environments:** {{iotPythonURL}}
- API Endpoint:** /IoT / Sensores 200 OK
- Method:** GET
- Query Params:**

Key	Value	Description
Key	Value	Description
- Body:** {} JSON
- Headers:** (4)
- Test Results:** 200 OK • 22 ms • 2.57 KB
- Visualizer:** JSON, Preview, Visualize
- Body Content:**

```
1 {
2     "status": "success",
3     "message": "Sensores recuperados correctamente",
4     "params": [
5         {"tipo": null,
6          "ubicacionId": null
7     },
8     "total": 10,
9     "sensores": [
10         {
11             "id": "S-TEMP-OUT-01",
12             "nombre": "Sensor de Temperatura Exterior",
13             "tipo": "temperatura",
14             "ubicacion": "Tech del Edificio A",
15             "modelo": "DHT-22-Pro",
16             "fabricante": "TechSense Co.",
17             "unidad medida": "C"
18         }
19     ]
20 }
```
- Cloud View**, **Find and replace**, **Console**
- Runner**, **Start Proxy**, **Cookies**, **Vault**, **Trash**

- Sensores por tipo: Filtrados por sensores de temperatura

The screenshot shows the Postman interface with a collection named "Pruebas CRM-IoT". A specific request titled "Sensores por tipo 200 OK" is selected. The request URL is `http://{{iotPythonURL}}/sensores?tipo=temperatura`. The "Params" tab is active, showing a parameter "tipo" with the value "temperatura". The "Body" tab shows a JSON response with two sensors, both of which are temperature sensors located on the roof of Building A. The response status is 200 OK.

```
1 {
  "status": "success",
  "message": "Sensores recuperados correctamente",
  "params": [
    {
      "tipo": "temperatura",
      "ubicacionId": null
    }
  ],
  "total": 2,
  "sensores": [
    {
      "id": "S-TEMP-OUT-01",
      "nombre": "Sensor de Temperatura Exterior",
      "tipo": "temperatura",
      "ubicacion": "Techo del Edificio A",
      "modelo": "DHT-22-Pro",
      "fabricante": "TechSense Co.",
      "unidad medida": "C"
    }
  ]
}
```

- Sensores por ubicación

The screenshot shows the Postman interface with the same collection "Pruebas CRM-IoT". A request titled "Sensores por ubicación 200 OK" is selected. The request URL is `http://{{iotPythonURL}}/sensores?ubicacionId=Sala%20de%20Servidores%201`. The "Params" tab is active, showing a parameter "ubicacionId" with the value "Sala%20de%20Servidores%201". The "Body" tab shows a JSON response with one sensor, which is a humidity sensor located in the first server room. The response status is 200 OK.

```
1 {
  "status": "success",
  "message": "Sensores recuperados correctamente",
  "params": [
    {
      "tipo": null,
      "ubicacionId": "Sala de Servidores 1"
    }
  ],
  "total": 1,
  "sensores": [
    {
      "id": "S-HUM-INT-02",
      "nombre": "Sensor de Humedad Interior",
      "tipo": "humedad",
      "ubicacion": "Sala de Servidores 1",
      "modelo": "HH-V3",
      "fabricante": "EnviroGuard",
      "unidad medida": "%RH"
    }
  ]
}
```

- Lecturas de los sensores

Pruebas CRM-IoT / IoT Lecturas 200 OK

GET {{iotPythonURL}} /lecturas

Query Params

Key	Value	Description
Key	Value	Description

Body Cookies Headers (4) Test Results

```

1 {
2   "status": "success",
3   "message": "lecturas recuperadas correctamente",
4   "params": {},
5   "total": 60,
6   "lecturas": [
7     {
8       "id_lectura": "L-20251027-001",
9       "id_sensor": "S-TEMP-04",
10      "valor": 22.5,
11      "unidad": "C"
12    }
13  ]
14 }
15 }
16 }
17
  
```

200 OK 77 ms 7.99 KB Save Response

- Lecturas por sensor

Pruebas CRM-IoT / IoT Lecturas por sensor 200 OK

GET {{iotPythonURL}} /lecturas?sensorId=S-TEMP-04

Query Params

Key	Value	Description
<input checked="" type="checkbox"/> sensorId	S-TEMP-04	
Key	Value	Description

Body Cookies Headers (4) Test Results

```

1 {
2   "status": "success",
3   "message": "lecturas recuperadas correctamente",
4   "params": {},
5   "total": 4,
6   "lecturas": [
7     {
8       "id_lectura": "L-20251027-001",
9       "id_sensor": "S-TEMP-04",
10      "valor": 22.5,
11      "unidad": "C"
12    }
13  ]
14 }
15 }
16 }
17
  
```

200 OK 12 ms 828 B Save Response

- Lecturas por ubicación

The screenshot shows the Postman application interface. The top navigation bar includes 'Home', 'Workspaces', 'API Network', 'Search Postman', 'Ctrl + K', 'invite', 'Upgrade', and environment selection. The main workspace title is 'Pruebas CRM-IoT / IoT / Lecturas por ubicación 200 OK'. A GET request is selected with the URL `({{iotPythonURL}}) /lecturas?ubicacionId=Tanque%20de%20Agua%20Potable%20P1`. The 'Params' tab is active, showing a single parameter 'ubicacionId' with the value 'Tanque%20de%20Agua%20Potable%20P1'. Below the request, the response status is '200 OK' with a response time of '12 ms' and a size of '1.17 KB'. The response body is displayed as JSON:

```

1 {
2   "status": "success",
3   "message": "lecturas recuperadas correctamente",
4   "params": [
5     {
6       "sensorId": null,
7       "ubicacionId": "Tanque de Agua Potable P1",
8       "from": null,
9       "to": null,
10      "limit": 100
11    },
12    "total": 7,
13    "lecturas": [
14      {
15        "id_lectura": "L-003",
16        "id_sensor": "S-PH-05",
17        "valor": 7.05,
18        "unidad": "mH"
19      }
20    ]
21  }
22}

```

- Lecturas en un rango de fechas

The screenshot shows the Postman application interface. The top navigation bar includes 'Home', 'Workspaces', 'API Network', 'Search Postman', 'Ctrl + K', 'invite', 'Upgrade', and environment selection. The main workspace title is 'Pruebas CRM-IoT / IoT / Lecturas en rango de fechas 200 OK'. A GET request is selected with the URL `({{iotPythonURL}}) /lecturas?from=2025-10-27T10:00:00Z&to=2025-10-27T12:00:00Z`. The 'Params' tab is active, showing two parameters: 'from' with the value '2025-10-27T10:00:00Z' and 'to' with the value '2025-10-27T12:00:00Z'. Below the request, the response status is '200 OK' with a response time of '69 ms' and a size of '8.03 KB'. The response body is displayed as JSON:

```

1 {
2   "status": "success",
3   "message": "lecturas recuperadas correctamente",
4   "params": [
5     {
6       "sensorId": null,
7       "ubicacionId": null,
8       "from": "2025-10-27T10:00:00Z",
9       "to": "2025-10-27T12:00:00Z",
10      "limit": 100
11    },
12    "total": 60,
13    "lecturas": [
14      {
15        "id_lectura": "L-20251027-001",
16        "id_sensor": "S-TEMP-04",
17        "valor": 22.5,
18        "unidad": "C"
19      }
20    ]
21  }
22}

```

- Lecturas limitadas

The screenshot shows the Postman interface with a successful API call. The URL is `({iotPythonURL}) /lecturas?limit=5`. The response status is 200 OK, with a response time of 11 ms and a size of 983 B. The response body is a JSON object:

```
1 {
2     "status": "success",
3     "message": "lecturas recuperadas correctamente",
4     "params": [
5         {
6             "sensorId": null,
7             "ubicacionId": null,
8             "from": null,
9             "to": null,
10            "limit": 5
11        },
12        "total": 60,
13        "lecturas": [
14            {
15                "id_lectura": "L-20251027-001",
16                "id_sensor": "S-TEMP-04",
17                "valor": 22.5,
18                "unidad": "C"
19            }
20        ]
21 }
```

- Comprobación de errores

- Lecturas limitadas fuera de rango (400)

The screenshot shows the Postman interface with an error response. The URL is `({iotPythonURL}) /lecturas?limit=0`. The response status is 400 Bad Request, with a response time of 6 ms and a size of 185 B. The response body is a JSON object:

```
1 {
2     "detail": "'limit' debe ser entero entre 1 y 1000"
3 }
```

- Lecturas con fechas inválidas

The screenshot shows the Postman interface with a collection named "Pruebas". A GET request is being tested against the URL `({{iotPythonURL}}) /lecturas?from=fecha_invalida`. The "Params" tab is selected, showing a parameter named "from" with the value "fecha\_invalida". The response status is 400 Bad Request, with a duration of 5 ms and a size of 182 B. The response body is a JSON object with one key, "detail": "Fecha ISO inválida: fecha\_invalida".

- Lectura en rango de fechas incorrectas

The screenshot shows the Postman interface with a collection named "Pruebas". A GET request is being tested against the URL `({{iotPythonURL}}) /lecturas?from=2025-12-31T23:59:59Z&to=2024-01-01T00:00:00Z`. The "Params" tab is selected, showing parameters "from" with value "2025-12-31T23:59:59Z" and "to" with value "2024-01-01T00:00:00Z". The response status is 400 Bad Request, with a duration of 16 ms and a size of 181 B. The response body is a JSON object with one key, "detail": "'from' no puede ser mayor que 'to'".