



DOCUMENTACIÓN TÉCNICA – API Registro Digital

Versión: 1.0

Última actualización: 02/12/2025

Responsable: Dirección de Tecnología

1. Descripción General

El endpoint **Registro Digital** permite a entidades autorizadas crear un registro inicial de un usuario para iniciar un proceso de validación de identidad digital, **sin notificaciones al consumidor**.

El servicio:

1. Valida credenciales del usuario API.
 2. Genera un **código único**.
 3. Retorna una **URL única** para continuar el proceso.
-

2. URL del Servicio

POST <https://consultorid.com/api/digital/crear/registro.php>

3. Autenticación

La autenticación se realiza enviando:

- **Username** → Número de documento del usuario API
 - **Password** → Contraseña del usuario API
 - El usuario debe estar **activo**
 - La entidad del usuario debe estar **activa**
-

4. Método HTTP

POST

5. Headers requeridos

**Header****Valor**

Content-Type application/json

6. Body del Request (JSON)Enviar el cuerpo de la petición en formato **JSON**:

```
{  
    "Username": "123456789",  
    "Password": "MiClaveSegura",  
    "Dni": "987654321",  
    "TipoDni": "1",  
    "Nombres": "Jorge Salamanca"  
}
```

7. Descripción de Parámetros**Campo****Tipo****Requerido****Descripción**

Username string Sí Número de documento del usuario API.

Password string Sí Contraseña del usuario API.

Dni string Sí Número de documento del ciudadano a validar.

TipoDni entero Sí ID del tipo de documento, debe existir en la tabla tipo_documento.

Nombres string Sí Nombres completos del ciudadano.



8. Tipos de Documento disponibles

ID Tipo de Documento

- 1 Cédula de ciudadanía
 - 2 Tarjeta de identidad
 - 3 Registro Civil
 - 4 Cédula de extranjería
 - 5 Doc. definido por la DIAN
 - 6 NIT
 - 7 P.E.P.
 - 8 Pasaporte
 - 9 Visa
-

9. Ejemplo de Request en Postman

Método

POST

URL

<https://consultorid.com/api/digital/crear/registro.php>

Headers

Key	Value
-----	-------

Content-Type application/json

Body → Raw → JSON

{

 "Username": "123456789",

 "Password": "ClaveAaignada",

 "Dni": "987654321",

 "TipoDni": "1",



"Nombres": "Jorge Salamanca"

}

10. Ejemplo de Response exitoso (200)

{

 "status": 200,

 "message": "Registro generado exitosamente",

 "data": {

 "Codigo": "A8F121PIO1R7I",

 "Url": "https://decrimlawyers.com/digital/index.php?cod=A8F121PIO1R7I",

 "Fecha": "2025-12-05 10:46:29"

 }

}

11. Ejemplos de Responses de Error

400 – Formato incorrecto

{

 "status": 400,

 "message": "Formato de solicitud incorrecto."

}

400 – Falta un campo obligatorio

{

 "status": 400,

 "message": "El campo 'Nombres' es obligatorio."

}

**400 – Tipo de documento no válido**

```
{  
  "status": 400,  
  "message": "Error al crear registro, tipo de documento incorrecto."  
}
```

401 – Usuario no encontrado

```
{  
  "status": 401,  
  "message": "Usuario no encontrado."  
}
```

401 – Contraseña incorrecta

```
{  
  "status": 401,  
  "message": "Error de autenticación."  
}
```

403 – Usuario sin permisos API

```
{  
  "status": 403,  
  "message": "El usuario no tiene atributos programáticos."  
}
```

403 – Usuario inactivo

```
{  
  "status": 403,
```



"message": "El usuario se encuentra bloqueado."

}

403 – Entidad inactiva

{

"status": 403,

"message": "La entidad asociada al usuario se encuentra suspendida."

}

500 – Error interno del servidor

{

"status": 500,

"message": "Error del servidor: <detalle>"

}

12. Consideraciones Técnicas

- La API debe consumirse **exclusivamente por HTTPS**.
 - El cuerpo debe enviarse en formato **JSON**.
 - El código generado es **único** y no se repite.
 - La URL generada permite continuar el proceso digital.
-