

Guía rápida de usuario – API FERM Registry

Mini-manual para acceso y consumo del API (versión 2026-02-16)

URL de administración <https://ferm.fao.org/admin/api>

Base URL del API (consumo): <https://ferm.fao.org/api>

1. Requisitos

- Acceso al centro administrativo de FERM (usuario autorizado).
- Generar una API Key desde el panel administrativo.
- Herramienta para consumir el API (Postman/Insomnia) o línea de comandos (cURL).

2. Crear la API Key (clave de API)

1. Ingresar a <https://ferm.fao.org/admin/api>
2. En la sección de Clave de API, seleccionar “Crear clave”.
3. Copiar y almacenar la clave en un lugar seguro (Secret Manager/Vault).
4. Nota: solo puede existir 1 clave activa por usuario. Si se crea una nueva, la anterior se revoca.

3. Autenticación (cómo enviar la clave)

Todas las solicitudes deben incluir el siguiente header:

- `x-api-key: <SU_API_KEY>`

Ejemplo (header en cURL):

```
curl -H "x-api-key: <SU_API_KEY>" https://ferm.fao.org/health
```



4. Endpoints disponibles

Los endpoints se encuentran listados en la interfaz del API y se agrupan así:

Grupo	Método y ruta	Descripción
Health	GET /health	Verificación básica del servicio.
Health	GET /health/detailed	Verificación detallada con estado.
Projects	POST /api/projects	Crear un nuevo proyecto de restauración.
Projects	GET /api/projects/{id}	Consultar un proyecto por ID.
Samples	GET /api/samples	Listar ejemplos publicados (JSON/GeoJSON).

5. Prueba rápida recomendada

A) Validar conectividad (Health):

```
curl -X GET "https://ferm.fao.org/health" \
-H "x-api-key: <SU_API_KEY>"
```

B) Health detallado (opcional):

```
curl -X GET "https://ferm.fao.org/health/detailed" \
-H "x-api-key: <SU_API_KEY>"
```

6. Crear un proyecto (POST /api/projects)

El endpoint recibe un GeoJSON FeatureCollection:

- properties: metadatos del proyecto (institución, título, años, contactos, países, etc.).
- features[]: sitios/áreas del proyecto con geometry (Polygon) y properties del sitio.

Ejemplo (payload básico):

```

curl -X POST "https://ferm.fao.org/api/projects" \
-H "x-api-key: <SU_API_KEY>" \
-H "Content-Type: application/json" \
--data-binary @- <<'GEOJSON'

{
  "type": "FeatureCollection",
  "properties": {
    "institutionId": "ORG_SAMPLE_001",
    "title": "Demo Restoration Project (Basic)",
    "startingYear": 2026,
    "restorationTypes": [1],
    "pointsOfContact": [
      { "name": "Alex Rivera", "email": "alex.rivera@example.org" }
    ],
    "organizations": [
      {
        "name": "Restoration Demo Org",
        "role": "Lead implementer",
        "type": "NGO",
        "country": "PA"
      }
    ],
    "country": "PA",
    "countries": ["PA"]
  },
  "features": [
    {
      "type": "Feature",
      "geometry": {
        "type": "Polygon",
        "coordinates": [
          [
            [-80.0, 8.5],
            [-80.0, 8.6],
            [-79.9, 8.6],
            [-79.9, 8.5],
            [-80.0, 8.5]
          ]
        ],
        "properties": {
          "siteName": "Demo Site A",
          "area": 25.0
        }
      }
    }
  ]
}
GEOJSON

```



7. Consultar un proyecto por ID (GET /api/projects/{id})

Luego de crear un proyecto, el sistema retorna un identificador. Use ese ID para consultar:

```
curl -X GET "https://ferm.fao.org/api/projects/<PROJECT_ID>" \
-H "x-api-key: <SU_API_KEY>"
```

8. Consumo desde Postman / Insomnia (pasos rápidos)

1. Crear una solicitud (GET/POST) con la URL del endpoint.
2. En Headers agregar: x-api-key = <SU_API_KEY>.
3. Para POST agregar además: Content-Type = application/json.
4. En Body (raw/JSON), pegar el GeoJSON FeatureCollection.
5. Enviar y validar respuesta (status 200/201 y cuerpo de respuesta).

9. Recomendaciones operativas para instituciones

- No guardar la API Key en código fuente. Usar secretos (Secret Manager/Vault).
- Iniciar con dataset pequeño (1–10 proyectos) y escalar gradualmente.
- Registrar bitácora local: fecha/hora, payload, respuesta, y projectId devuelto.
- Automatización: ejecutar un job programado (cron/scheduler) que extraiga, transforme a GeoJSON, envíe y registre resultados.

10. Checklist de solución de problemas

- 401/403: API Key faltante o inválida. Verifique header x-api-key.
- 400: payload inválido. Revise estructura GeoJSON (FeatureCollection, properties, features).
- Clave anterior no funciona: si creó una nueva, la anterior queda revocada (solo 1 activa).

Aida Zapata Orozco

Forestry Software Developer

Restoration Flagship, NFOM

Food and Agriculture Organization of the United Nations (FAO)