

Multi-Tenant Metadata API Specification

Version: 1.0.0

Overview

The Multi-Tenant Metadata API provides endpoints for defining, updating, and retrieving configuration and metadata definitions in a multi-tenant SaaS environment. Each tenant's metadata is logically isolated.

Base URL

`https://api.example.com/v1/tenants/{tenant_id}/metadata`

Authentication

- API Key via `Authorization` header (Bearer <token>).
- Every request must be scoped to an authenticated tenant.

Endpoints Summary

Method	Endpoint	Description
GET	<code>/tenants/{tenant_id}/metadata</code>	List all metadata schemas for a tenant
POST	<code>/tenants/{tenant_id}/metadata</code>	Create a new metadata schema
GET	<code>/tenants/{tenant_id}/metadata/{schema_id}</code>	Get a specific metadata schema by ID
PUT	<code>/tenants/{tenant_id}/metadata/{schema_id}</code>	Update an existing metadata schema
DELETE	<code>/tenants/{tenant_id}/metadata/{schema_id}</code>	Delete a metadata schema

Data Model: Metadata Schema

```
{
  "id": "string",
  "name": "string",
  "fields": [
    {
      "name": "string",
      "type": "string",
      "required": "boolean",
      "default": "any"
    }
  ],
  "created_at": "ISO8601 timestamp",
  "updated_at": "ISO8601 timestamp"
}
```

Sample Requests

Create Metadata Schema

```
POST /tenants/acme/metadata
Content-Type: application/json
Authorization: Bearer <token>
{
  "name": "CustomerProfile",
  "fields": [
    {"name": "firstName", "type": "string", "required": true},
    {"name": "createdDate", "type": "datetime", "required": false}
  ]
}
```

Response

```
Status: 201 Created
{
  "id": "schema_45ac",
  "name": "CustomerProfile",
  "fields": [
    {"name": "firstName", "type": "string", "required": true},
    {"name": "createdDate", "type": "datetime", "required": false}
  ],
  "created_at": "2024-06-15T10:30:00Z",
  "updated_at": "2024-06-15T10:30:00Z"
}
```

Error Handling

```
{
  "error": "string",
  "message": "string",
  "status": integer
}
```

Status Codes

- 200 – Success
- 201 – Created
- 400 – Bad Request
- 401 – Unauthorized
- 403 – Forbidden
- 404 – Not Found
- 409 – Conflict
- 500 – Internal Server Error

End of sample. Fields and structures may be extended as per implementation needs.