

fireservice-adapter

1. Overview

Firestore Adapter is a RESTful microservice designed to abstract and simplify interactions with Google Firestore for other backend services.

It provides easy-to-use endpoints for common Firestore operations (CRUD), making it simple to integrate Firestore into your apps without directly handling Google’s SDKs or permissions.

2. Architecture

- **Language/Stack:** Python (Flask/FastAPI), Dockerized
- **Deployed on:** Google Cloud Run (Stateless, auto-scaled)
- **Authentication:** Service account for backend-to-backend communication, [optional: API key or OIDC for clients]
- **State:** Stateless (all data in Firestore)

3. Endpoints

Base URL:

```
https://<cloud-run-url>/
```

Method	Path	Description
GET	/health	Health check
GET	/documents	List all documents
GET	/documents/{id}	Retrieve document by ID
POST	/documents	Create a new document
PUT	/documents/{id}	Update an existing document

DELETE	/documents/{id}	Delete document by ID
--------	-----------------	-----------------------

4. Authentication

- **Backend-to-backend:**

Use GCP service accounts or OIDC token (if enabled).

- **API Key / Bearer Token:**

If public endpoints are enabled, use an API key in the header:

Authorization: Bearer <token>

(Omit if only private internal services call this.)

5. Request & Response Examples

GET /health

Request:

```
curl https://<cloud-run-url>/health
```

Response:

```
{"status": "ok"}
```

POST /documents

Request:

```
curl -X POST https://<cloud-run-url>/documents ¥  
-H "Content-Type: application/json" ¥  
-d '{"field1": "value1", "field2": "value2"}'
```

Response:

```
{
  "id": "12345",
  "field1": "value1",
  "field2": "value2",
  "createdAt": "2025-05-27T10:20:00Z"
}
```

GET /documents/{id}

Request:

```
curl https://<cloud-run-url>/documents/12345
```

Response:

```
{
  "id": "12345",
  "field1": "value1",
  "field2": "value2",
  "createdAt": "2025-05-27T10:20:00Z"
}
```

PUT /documents/{id}

Request:

```
curl -X PUT https://<cloud-run-url>/documents/12345 \
-H "Content-Type: application/json" \
-d '{"field1": "newvalue"}
```

Response:

```
{
  "id": "12345",
  "field1": "newvalue",
}
```

```
"field2": "value2",  
"updatedAt": "2025-05-27T10:25:00Z"  
}
```

DELETE /documents/{id}

Request:

```
curl -X DELETE https://<cloud-run-url>/documents/12345
```

Response:

```
{  
  "success": true  
}
```
