

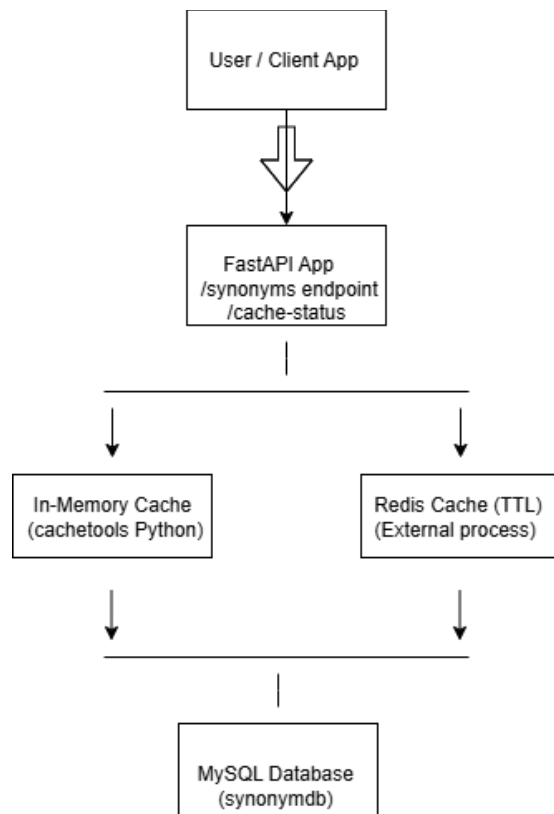
The Data Engine Synonym System is a FastAPI-based web service that retrieves synonym information from a MySQL database, using an efficient dual-layer caching mechanism (Redis + in-memory) with TTL and automatic expiration.

### === Folder Structure ===

synonym\_api/

- |— main.py
- |— models.py
- |— db.py
- |— cache/
  - | |— \_\_init\_\_.py
  - | |— base.py
  - | |— memory\_cache.py
  - | |— redis\_cache.py
- |— config.py
- |— .env

### === Architecture Overview ===



### === Features ===

- Bulk retrieval of synonyms from a MySQL database
- Dual caching system: Redis (distributed) + in-memory
- Configurable TTL for cache entries
- Secured via API key header (`X-API-KEY`)
- Swagger-based interactive documentation
- Simple to run and test locally using VSCode and virtual environment

### === Tech Stack ===

- FastAPI (web framework)
- SQLAlchemy (DB ORM layer using SQLAlchemy)
- MySQL (via MySQL Workbench)
- Redis (optional for distributed caching)
- In-memory caching (via `cachetools`)
- Environment management (`python-dotenv`)

### === Setup Instructions (Windows / VSCode / CMD) ===

# Clone and navigate

```
cd synonym_api
```

# Create virtual environment

```
python -m venv venv
```

```
.\venv\Scripts\activate
```

# Install dependencies

```
pip install fastapi uvicorn sqlalchemy pymysql cachetools redis python-dotenv cryptography
```

```
# My SQL Setup in mysql workbench
```

```
CREATE DATABASE synonymdb;
```

```
USE synonymdb;
```

```
CREATE TABLE synonym (  
    id INT PRIMARY KEY AUTO_INCREMENT,  
    word VARCHAR(255),  
    synonyms TEXT  
);
```

```
INSERT INTO synonym (word, synonyms)
```

```
VALUES
```

```
('fast', 'quick, rapid, speedy'),
```

```
('smart', 'intelligent, clever, bright');
```

### **=== Redis Setup (Windows) ===**

```
# Navigate to the Redis installation folder, where you have installed
```

```
cd C:\redis
```

```
# Start the Redis server with config
```

```
redis-server.exe redis.windows.conf
```

### **=== Security ===**

This API requires a custom header for authentication:

```
Header Name:  X-API-KEY
```

Header Value: wbcache

### === Running the App ===

In vs code:

```
uvicorn main:app --reload
```

The app will run at:

<http://127.0.0.1:8000>

Interactive Swagger docs (auto-generated):

**<http://127.0.0.1:8000/docs>**

You must set X-API-KEY (use: wbcache) in the Swagger UI to authorize requests.

### API Endpoints

GET /synonyms

Returns all synonym records

Checks Redis/in-memory cache first

Adds "source": "cache" or "source": "database" in response

GET /cache-status

Returns current cache backend in use (redis or in\_memory)

### === Caching Logic ===

- Client hits /synonyms API with API key.
- System first checks in-memory cache.
- If not found and Redis is enabled, it checks Redis cache.
- If still not found, it queries the MySQL database.
- Retrieved data is cached in both Redis and memory (with TTL).
- Response includes metadata: "source": "cache" or "database".

# Data Engine Synonym System 0.1.0 OAS 3.1

/openapi.json

default

GET

/synonyms

Get Synonyms

Parameters

Name

Description

X-API-KEY \* required

wbcache

string

(header)

Execute

Clear

Responses

Curl

```
curl -X 'GET' \
  'http://127.0.0.1:8000/synonyms' \
  -H 'accept: application/json' \
  -H 'X-API-KEY: wbcache'
```

Request URL

```
http://127.0.0.1:8000/synonyms
```

Server response

Code

Details

200

Response body

```
{
  "word": "fast",
  "synonyms": [
    "quick",
    "rapid",
    "speedy"
  ],
  "source": "cache"
},
{
  "word": "smart",
  "synonyms": [
    "intelligent",
    "clever",
    "bright"
  ],
  "source": "cache"
}
]
```

Download

Response headers

Author: Adhithya Kiran, adhithyakiran@gwu.edu

X-API-KEY \* required  
string  
(header)

wbcache

Execute

Clear

## Responses

Curl

```
curl -X 'GET' \
'http://127.0.0.1:8000/cache-status' \
-H 'accept: application/json' \
-H 'X-API-KEY: wbcache'
```

Request URL

http://127.0.0.1:8000/cache-status

Server response

Code	Details
------	---------

200

Response body

```
{
  "cache_backend": "redis"
}
```

Response headers



Download