

7. API Documentation for Each Operation

- **Base URL:** `http://localhost:5000`
- **Endpoints:**
 - **Get All CVEs** (`GET /cves`)
 - **Description:** Fetches all CVEs with filtering, sorting, and pagination.
 - **Query Parameters:**
 - `cve_id` (string) – Filter by specific CVE ID.
 - `year` (int) – Filter CVEs from a specific year.
 - `min_score`, `max_score` (float) – Filter by CVE base score.
 - `search` (string) – Search in descriptions or CVE ID.
 - `status` (string) – Filter by vulnerability status.
 - `source_identifier` (string) – Filter by source.
 - `sort_by` (string) – Field to sort by (`published_date`, `base_score`, etc.).
 - `sort_order` (string) – `ASC` or `DESC`.
 - `limit` (int) – Number of records per page (default: 10).
 - `offset` (int) – Pagination offset (default: 0).

Response Example:

```
{
  "total_records": 1000,
  "current_page": 1,
  "total_pages": 100,
  "page_range": "1 to 10",
  "cves": [
    {
      "id": "CVE-2024-0001",
      "published_date": "01 Jan 2024",
      "last_modified_date": "10 Jan 2024",
      "base_score": 7.5,
      "description": "Sample CVE description",
      "vuln_status": "Under Investigation"
    }
  ]
}
```

- **Get CVE by ID** (GET /cve/<string:cve_id>)
 - **Description:** Fetches details of a specific CVE.

Response Example:

```
{
  "id": "CVE-2024-0001",
  "published_date": "01 Jan 2024",
  "last_modified_date": "10 Jan 2024",
  "base_score": 7.5,
  "description": "Detailed CVE description",
  "access_vector": "Network",
  "severity": "High"
}
```