

API Security

1.successfully logged out

The screenshot shows a Postman interface with a workspace named 'My Workspace'. The left sidebar shows a collection of APIs, including 'VacQ' and 'Authentication'. The main panel displays a GET request to 'VacQ / Authentication / Get me' with the URL '({URL})/api/v1/auth/me'. The request headers are set to 'Content-Type: application/json' and 'Authorization: Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ...'. The response status is '401 Unauthorized' with a time of 8 ms and size of 305 B. The response body is a JSON object:

```
{  "success": false,  "msg": "Not authorize to access this route"}
```

2.prevent injection

The screenshot shows a Postman interface with a workspace named 'My Workspace'. The left sidebar shows a collection of APIs, including 'VacQ' and 'Authentication'. The main panel displays a POST request to 'VacQ / Authentication / Login' with the URL '({URL})/api/v1/auth/login'. The request body is a JSON object:

```
{  "email": {"$gt": ""},  "password": "1234456"}
```

. The response status is '401 Unauthorized' with a time of 28 ms and size of 313 B. The response body is a JSON object:

```
{  "success": false,  "msg": "Cannot convert email or password to string"}
```

3.apply helmet

Postman interface showing a GET request to `http://localhost:5000/api/v1/hospitals`. The response status is 200 OK. The response headers are displayed, showing various security headers (CSP, COOP, CORS, etc.) and the content type is `application/json; charset=utf-8`.

Key	Value	Description
Content-Security-Policy	default-src 'self';base-uri 'self';font-src 'self' https://data:...form-action 'self';frame...	
Cross-Origin-Opener-Policy	same-origin	
Cross-Origin-Resource-Policy	same-origin	
Origin-Agent-Cluster	?1	
Referrer-Policy	no-referrer	
Strict-Transport-Security	max-age=15552000; includeSubDomains	
X-Content-Type-Options	nosniff	
X-DNS-Prefetch-Control	off	
X-Download-Options	noopen	
X-Frame-Options	SAMEORIGIN	
X-Permitted-Cross-Domain-Policies	none	
X-XSS-Protection	0	
Content-Type	application/json; charset=utf-8	
Content-Length	10534	

4.prevent XSS

Postman interface showing a POST request to `{{URL}}/api/v1/hospitals`. The request body is a JSON object containing hospital details. The response status is 201 Created. The response body is a JSON object indicating success.

```
1 {
2   "name": "Hack Hospital <script>alert(1)</script>",
3   "address": "Ratchathewi",
4   "district": "Bangkok",
5   "province": "Bangkok",
6   "postalcode": "10400",
7   "tel": "02-1234567",
8   "region": "Bangkok"
9 }
```

```
1 {
2   "success": true,
3   "data": {
4     "name": "Hack Hospital",
5     "address": "Ratchathewi",
6     "district": "Bangkok",
7     "province": "Bangkok",
8     "postalcode": "10400",
9     "tel": "02-1234567",
10    "region": "Bangkok",
11    "_id": "65dfb1bfcf895e14dcbfef9d",
12    "createdAt": "2023-07-14T12:34:56.789Z"
13  }
14 }
```

5.apply express-rate-limit

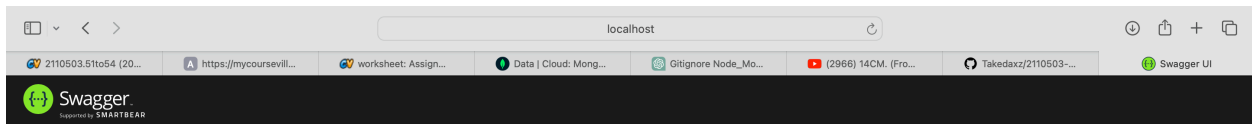
The screenshot shows the Postman interface with a workspace named 'VacQ'. A collection 'VacQ' is expanded, showing a folder 'Hospitals' with a 'GET ONE' endpoint. The endpoint URL is `{{URL}}/api/v1/hospitals/65d36c23969f8fef2a22d9bf`. The request is a GET method. The response status is 429 Too Many Requests, with a message: 'Too many requests, please try again later.'

Query Params table:

Key	Value	Description
Key	Value	Description

OpenAPI

1.Screen



Library API ^{1.0.0} ^{OAS 3.0}

A simple Express VacQ API

Hospitals The hospitals managing API

GET	/hospitals	Returns the list of all the hospitals
POST	/hospitals	Create a new hospital
GET	/hospitals/{id}	Get the hospital by id
PUT	/hospitals/{id}	Update the hospital by the id
DELETE	/hospitals/{id}	Remove the hospital by id

Schemas

Hospital >

2.add server

2110503.5f1054 (20...https://mycoursevill...worksheet: Assign...Data | Cloud: Mong...Gitignore Node_Mo...(2066) 14CM. (Fro...Takedaxz/2110503-...Swagger UI

localhost

Responses

Curl

```
curl -X 'GET' \
'http://localhost:5000/api/v1/hospitals' \
-H 'accept: application/json'
```

Request URL

```
http://localhost:5000/api/v1/hospitals
```

Server response

CodeDetails

200

Response body

```
{
  "success": true,
  "count": 25,
  "pagination": {
    "next": {
      "page": 2,
      "limit": 25
    }
  },
  "data": [
    {
      "_id": "65df1b1fcf895e14dcbfef9d",
      "name": "Hack Hospital",
      "address": "Ratchathewi",
      "district": "Bangkok",
      "province": "Bangkok",
      "postalcode": "10400",
      "tel": "02-1234567",
      "region": "Bangkok",
      "__v": 0,
      "appointments": [],
      "id": "65df1b1fcf895e14dcbfef9d"
    },
    {
      "id": "65df1a295dc5b516c2d4bbbf"
    }
  ]
}
```

Response headers

```
access-control-allow-origin: *
connection: keep-alive
```