

1. Tabel Data Base Untuk User dan Karyawan

The screenshot shows the pgAdmin 4 interface. On the left, the 'Object Explorer' pane displays a database schema with tables 'm_admin', 'm_employee', 'm_role', 'm_user', and 'm_user_credentials'. The main query editor shows a SQL query: `SELECT * FROM public.m_employee ORDER BY id ASC`. The 'Data Output' pane displays the results of the query, showing 13 rows of employee data.

id	email	is_member	name	phone
1	musoviet@viet.com	false	Muso	0336789988
2	samsang@viet.com	false	Samsan	0336789245
3	slaweg@viet.com	false	Slawente	0336664545
4	sukarno@viet.com	true	Ir Soekarno	03346565676
5	sukarno@viet.com	false	Sukarno	033478785676
6	kami@viet.com	true	Kami Ilyas (LIC)	033488885676
7	olive@viet.com	true	Oliver	033455885676
8	yamal@viet.com	false	Yamal	033488885676
9	malaka@viet.com	true	Getar Datuk Ibrahim Sutan Malaka	033486665676
10	yasin@viet.com	true	Yasin	033480085176
11	yasin@viet.com	false	Lev yasin	033677887665
12	yasin@viet.com	false	Lev yasin	033677887665
13	yasin@viet.com	false	Lev yasin	033677887664

2. End Point CRUD, Registrasi, Login, dan Pagination

- EndPoint CRUD

The screenshot shows the Swagger UI for a REST API. The 'Servers' section shows the base URL as `http://localhost:8080 - Generated server url`. The API is organized into three sections: 'Authentication', 'Employee', and 'auth-controller'. The 'Employee' section lists four endpoints for CRUD operations: GET /api/employees, PUT /api/employees, POST /api/employees, and DELETE /api/employees/{id}. The 'auth-controller' section lists three endpoints for authentication: POST /api/auth/register/employees, POST /api/auth/register/admin, and POST /api/auth/login.

-Registrasi

Activities Firefox Web Browser Aug 24 20:22

localhost:8080/swagger-ui/index.html#/auth-controller/registerAdmin

auth-controller

POST /api/auth/register/employees

POST /api/auth/register/admin

Parameters

No parameters

Try it out

Request body **required** application/json

Example Value Schema

```
{
  "username": "Cv8008q11hs1",
  "password": "string"
}
```

Responses

Code	Description	Links
200	OK	No links

Media type

application/json

Controls Accept header:

Example Value Schema

```
{}
```

POST /api/auth/login

-login

Activities Firefox Web Browser Aug 24 20:31

localhost:8080/swagger-ui/index.html#/employee-controller/getAllEmployee

POST /api/auth/login

Parameters

No parameters

Try it out

Reset

Request body **required** application/json

Example Value Schema

```
{
  "username": "amlr",
  "password": "ppp"
}
```

Responses

Curl

```
curl -X 'POST' \
  http://localhost:8080/api/auth/login \
  -H 'accept: */*' \
  -H 'content-type: application/json' \
  -d '{
    "username": "amlr",
    "password": "ppp"
  }'
```

Request URL

http://localhost:8080/api/auth/login

Server response

Code	Details
200	<p>Response body</p> <pre>{ "message": "successfully login", "statusCode": 200, "data": null, "login": null }</pre>

Download

-Pagination

The screenshot shows the Swagger UI for the 'Employee' API. The endpoint is a GET request to '/api/employees'. The parameters section lists the following query parameters:

Name	Description
page	Default value : 1
integer(int32) (query)	1
size	Default value : 10
integer(int32) (query)	10
direction	Default value : asc
string (query)	asc
sortBy	Default value : name
string (query)	name

The responses section shows a 200 OK response with no links.

3. menggunakan security jwt dan Authorization Bearer Token

The screenshot shows the Swagger UI for the 'EmployApp API'. The API is documented for managing employee information and user authentication. The endpoints are listed under the 'employee-controller' section:

Method	Endpoint
GET	/api/employees
PUT	/api/employees
POST	/api/employees
GET	/api/employees/{id}
DELETE	/api/employees/{id}

An 'Available authorizations' dialog box is open, showing 'employ-app-security (http, Bearer)' with a 'Value:' field and 'Authorize' and 'Close' buttons.