

Application Programming Interface

Pertemuan 14

Gelar Aditya Pratama, S.T., M.Kom.
gelar@stmik-amikbandung.ac.id



laravel api & mysql – swagger api doc

- Jalankan command
composer require "darkaonline/l5-swagger"
- Nanti akan ada package baru tertulis di composer.json

```
{ } composer.json M X
{ } composer.json > { } require
1  {
2      .... "name": "laravel/laravel",
3      .... "type": "project",
4      .... "description": "The Laravel Framework.",
5      .... "keywords": ["framework", "laravel"],
6      .... "license": "MIT",
7      .... "require": {
8          .... "php": "^8.0",
9          .... "darkaonline/l5-swagger": "^8.5",
10         .... "fruitcake/laravel-cors": "^2.0.5",
11         .... "guzzlehttp/guzzle": "^7.7",
12         .... "laravel/framework": "^9.0",
```

Swagger



- Jalankan command
php artisan vendor:publish --provider "L5Swagger\L5SwaggerServiceProvider"
- Nanti akan ada file baru config/l5-swagger.php
edit line 15 menjadi 'api' => ''.

```
composer.json M  l5-swagger.php U X
config > l5-swagger.php
1  <?php
2
3  return [
4      .... 'default' => 'default',
5      .... 'documentations' => [
6          .... 'default' => [
7              .... 'api' => [
8                  .... 'title' => 'L5 Swagger UI',
9              ],
10         ],
11         .... 'routes' => [
12             .... /*
13              .... *Route for accessing api documentation interface
14              .... */
15             .... 'api' => '',
16         ],
```

- Edit file routes/web.php
comment bagian route /

```
routes > web.php > ...
2
3 use Illuminate\Support\Facades\Route;
4
5 /*
6 |-----
7 | Web Routes
8 |-----
9 |
10 | Here is where you can register web routes for your application. These
11 | routes are loaded by the RouteServiceProvider within a group which
12 | contains the "web" middleware group. Now create something great!
13 |
14 */
15
16 //Route::get('/', function() {
17 //    ...return view('welcome');
18 //});
19
```

Edit file app/Controllers/Controller.php



Controller.php M X

app > Http > Controllers > Controller.php > Controller

```
10
11 /**
12  * @OA\Info(
13  *     description="Made with ❤ by STMIK AMIK Bandung.",
14  *     version="1.0.0",
15  *     title="Laravel API Documentation",
16  *     termsOfService="http://swagger.io/terms/",
17  *     @OA\Contact(
18  *         email="akatgelar@gmail.com"
19  *     ),
20  *     @OA\License(
21  *         name="Apache 2.0",
22  *         url="http://www.apache.org/licenses/LICENSE-2.0.html"
23  *     )
24  * )
25  *
26  * @OA\SecurityScheme(
27  *     securityScheme="bearerAuth",
28  *     in="header",
29  *     name="bearerAuth",
30  *     type="http",
31  *     scheme="bearer",
32  *     bearerFormat="JWT",
33  * )
34  */
35 class Controller extends BaseController
```

Edit file app/Controllers/Api/AuthController.php



```
AuthController.php M X
app > Http > Controllers > Api > AuthController.php > AuthController
10
17 ...../**
18 .....* @OA\Post(
19 .....* .....path="/api/login",
20 .....* .....tags={"Auth"},
21 .....* .....summary="",
22 .....* .....description="Login",
23 .....* .....operationId="auth_login",
24 .....* .....@OA\RequestBody(
25 .....* .....@OA\MediaType(
26 .....* .....mediaType="multipart/form-data",
27 .....* .....@OA\Schema(
28 .....* .....@OA\Property(
29 .....* .....property="email",
30 .....* .....type="string"
31 .....* .....),
32 .....* .....@OA\Property(
33 .....* .....property="password",
34 .....* .....type="string"
35 .....* .....)
36 .....* .....)
37 .....* .....)
38 .....* .....),

AuthController.php M X
app > Http > Controllers > Api > AuthController.php > AuthController
39 .....* .....@OA\Response(
40 .....* .....response="default",
41 .....* .....description="OK",
42 .....* .....@OA\MediaType(
43 .....* .....mediaType="application/json",
44 .....* .....example={
45 .....* .....    "success"=True,
46 .....* .....    "message"="Login Successfull",
47 .....* .....    "data"={}
48 .....* .....}
49 .....* .....)
50 .....* .....)
51 .....* .....)
52 .....*/
53 .....public function login(Request $request)
54 .....{
55 .....    $request->validate([
56 .....        'email'=> 'required|string|email',
57 .....        'password'=> 'required|string',
58 .....    ]);
59 .....    $credentials = $request->only('email','password');
60 .....
61 .....    $token = Auth::attempt($credentials);
```

Edit file app/Controllers/Api/ProductController.php



```
ProductController.php M X
app > Http > Controllers > Api > ProductController.php > ProductController > index

17  ..../**
18  ....*  @OA\Get(
19  ....*      path="/api/product",
20  ....*      tags={"Product"},
21  ....*      summary="",
22  ....*      description="Get all data",
23  ....*      operationId="product_index",
24  ....*      @OA\Parameter(
25  ....*          name="per_page",
26  ....*          description="Per Page value is number. ex.: ?per_page=10",
27  ....*          required=false,
28  ....*          in="query",
29  ....*          @OA\Schema(
30  ....*              type="number"
31  ....*          )
32  ....*      ),
33  ....*      @OA\Parameter(
34  ....*          name="page",
35  ....*          description="Page value is number. ex.: ?page=2",
36  ....*          required=false,
37  ....*          in="query",
38  ....*          @OA\Schema(
39  ....*              type="number"
40  ....*          )
41  ....*      ),
```


Edit file app/Controllers/Api/ProductController.php



```
ProductController.php M X
app > Http > Controllers > Api > ProductController.php > ProductController > index
42 .....@OA\Parameter(
43 .....*.....name="sort",
44 .....*.....description="Sort value is string with rule column-name:order..ex:..?sort=id:asc",
45 .....*.....required=false,
46 .....*.....in="query",
47 .....*.....@OA\Schema(
48 .....*.....type="string"
49 .....*.....)
50 .....*.....),
51 .....*.....@OA\Parameter(
52 .....*.....name="where",
53 .....*.....description="Where value is object..ex:..?where={'name':'john','dob':'1990-12-31'}",
54 .....*.....required=false,
55 .....*.....in="query",
56 .....*.....@OA\Schema(
57 .....*.....type="string"
58 .....*.....)
59 .....*.....),
60 .....*.....@OA\Parameter(
61 .....*.....name="count",
62 .....*.....description="Count value is boolean..ex:..?count=true",
63 .....*.....required=false,
64 .....*.....in="query",
65 .....*.....@OA\Schema(
66 .....*.....type="boolean"
67 .....*.....)
68 .....*.....),
```

Edit file app/Controllers/Api/ProductController.php



ProductController.php M X

app > Http > Controllers > Api > ProductController.php > ProductController > index

```
69 .....@OA\Response(  
70 .....response="default",  
71 .....description="OK",  
72 .....@OA\MediaType(  
73 .....mediaType="application/json",  
74 .....example={  
75 .....    "status"=true,  
76 .....    "message"="Get Data Successful",  
77 .....    "data"={}  
78 .....}  
79 .....)  
80 .....)  
81 .....)  
82 ...../  
83 .....public function index(Request $request)  
84 .....{  
85 .....    // parameter  
86 .....    $where = $request->has('where') ? $request->get('where') : '{}';  
87 .....    $sort = $request->has('sort') ? $request->get('sort') : 'id:asc';  
88 .....    $per_page = $request->has('per_page') ? $request->get('per_page') : 2;  
89 .....    $page = $request->has('page') ? $request->get('page') : 1;  
90 .....    $count = $request->has('count') ? $request->get('count') : false;  
91 .....    $search = $request->has('search') ? $request->get('search') : '';
```

Edit file app/Controllers/Api/ProductController.php



ProductController.php M X

app > Http > Controllers > Api > ProductController.php > ProductController

```
155  ..../**
156  .... * @OA\Get(
157  .... * .....path="/api/product/{id}",
158  .... * .....tags={"Product"},
159  .... * .....summary="",
160  .... * .....description="Get data by id",
161  .... * .....operationId="product_show",
162  .... * .....@OA\Parameter(
163  .... * .....name="id",
164  .... * .....description="id",
165  .... * .....required=true,
166  .... * .....in="path",
167  .... * .....@OA\Schema(
168  .... * .....type="number"
169  .... * .....),
170  .... * .....),
```

ProductController.php M X

app > Http > Controllers > Api > ProductController.php > ProductController > show

```
171  .... * .....@OA\Response(
172  .... * .....response="default",
173  .... * .....description="OK",
174  .... * .....@OA\MediaType(
175  .... * .....mediaType="application/json",
176  .... * .....example={
177  .... * .....    "success"=true,
178  .... * .....    "message"="Get Data Successfull",
179  .... * .....    "data"={}
180  .... * .....})
181  .... * .....),
182  .... * .....),
183  .... * .....),
184  .... */
185  .... public function show($id)
186  .... {
187  ....     // query get by id
```

Edit file app/Controllers/Api/ProductController.php



```
ProductController.php M X
app > Http > Controllers > Api > ProductController.php > ProductController > store

199 ...../**
200 .....* @OA\Post(
201 .....* .....path="/api/product",
202 .....* .....tags={"Product"},
203 .....* .....summary="",
204 .....* .....description="Insert data",
205 .....* .....operationId="product_store",
206 .....* .....security={{"bearerAuth":{}}},
207 .....* .....@OA\RequestBody(
208 .....* .....@OA\MediaType(
209 .....* .....mediaType="application/json",
210 .....* .....@OA\Schema(
211 .....* .....@OA\Property(
212 .....* .....property="name",
213 .....* .....type="string"
214 .....* .....),
215 .....* .....@OA\Property(
216 .....* .....property="image",
217 .....* .....type="string"
218 .....* .....),
219 .....* .....@OA\Property(
220 .....* .....property="description",
221 .....* .....type="string"
222 .....* .....),

ProductController.php M X
app > Http > Controllers > Api > ProductController.php > ProductController > store

223 .....* .....@OA\Property(
224 .....* .....property="type",
225 .....* .....type="string"
226 .....* .....),
227 .....* .....@OA\Property(
228 .....* .....property="price",
229 .....* .....type="string"
230 .....* .....),
231 .....* .....),
232 .....* .....),
233 .....* .....),
234 .....* .....@OA\Response(
235 .....* .....response="default",
236 .....* .....description="OK",
237 .....* .....@OA\MediaType(
238 .....* .....mediaType="application/json",
239 .....* .....example={
240 .....* ....."success"=true,
241 .....* ....."message"="Insert Data Successfull",
242 .....* ....."data"={}
243 .....* .....})
244 .....* .....),
245 .....* .....),
246 .....* .....),
247 .....*/
248 .....public function store(Request $request)
249 .....{
250 .....    // query insert
251 .....    $query = Product::create($request->all());
```

Edit file app/Controllers/Api/ProductController.php



ProductController.php M X

app > Http > Controllers > Api > ProductController.php > ProductController > update

```
201 ...../**
202 .....*
203 .....* @OA\Put(
204 .....* .....path="/api/product/{id}",
205 .....* .....tags={"Product"},
206 .....* .....summary="",
207 .....* .....description="Update data",
208 .....* .....operationId="product_update",
209 .....* .....security={{"bearerAuth":{}}},
210 .....* .....@OA\Parameter(
211 .....* .....name="id",
212 .....* .....description="id",
213 .....* .....required=true,
214 .....* .....in="path",
215 .....* .....@OA\Schema(
216 .....* .....type="number"
217 .....* .....),
218 .....* .....),
219 .....* .....@OA\RequestBody(
220 .....* .....@OA\MediaType(
221 .....* .....mediaType="application/json",
222 .....* .....@OA\Schema(
223 .....* .....@OA\Property(
224 .....* .....property="name",
225 .....* .....type="string"
226 .....* .....),
227 .....* .....@OA\Property(
228 .....* .....property="image",
229 .....* .....type="string"
230 .....* .....),
```

ProductController.php M X

app > Http > Controllers > Api > ProductController.php > ProductController > update

```
291 .....* .....@OA\Property(
292 .....* .....property="description",
293 .....* .....type="string"
294 .....* .....),
295 .....* .....@OA\Property(
296 .....* .....property="type",
297 .....* .....type="string"
298 .....* .....),
299 .....* .....@OA\Property(
300 .....* .....property="price",
301 .....* .....type="string"
302 .....* .....),
303 .....* .....),
304 .....* .....),
305 .....* .....),
306 .....* .....@OA\Response(
307 .....* .....response="default",
308 .....* .....description="OK",
309 .....* .....@OA\MediaType(
310 .....* .....mediaType="application/json",
311 .....* .....example={
312 .....* ....."success"=true,
313 .....* ....."message"="Update Data Successfull",
314 .....* ....."data"={}
315 .....* .....}
316 .....* .....),
317 .....* .....),
318 .....* .....),
319 .....*/
320 .....public function update(Request $request, $id)
321 .....{
```

Edit file app/Controllers/Api/ProductController.php



ProductController.php M X

app > Http > Controllers > Api > ProductController.php > ProductController

```
338     .../**
339     ...* @OA\Delete(
340     ...*     path="/api/product/{id}",
341     ...*     tags={"Product"},
342     ...*     summary="",
343     ...*     description="Delete data",
344     ...*     operationId="product_destroy",
345     ...*     security={{"bearerAuth":{}}},
346     ...*     @OA\Parameter(
347     ...*         name="id",
348     ...*         description="id",
349     ...*         required=true,
350     ...*         in="path",
351     ...*         @OA\Schema(
352     ...*             type="number"
353     ...*         )
354     ...*     ),
```

ProductController.php M X

app > Http > Controllers > Api > ProductController.php > ProductController > destroy

```
355     ...* @OA\Response(
356     ...*     response="default",
357     ...*     description="OK",
358     ...*     @OA\MediaType(
359     ...*         mediaType="application/json",
360     ...*         example={
361     ...*             "success"=true,
362     ...*             "message"="Delete Data Successfull",
363     ...*             "data"={}
364     ...*         }
365     ...*     )
366     ...* )
367     ...*/
368
369     public function destroy($id)
370     {
371         ...// query insert
372         ...$query = Product::findOrFail($id);
```

- Jalankan command
php artisan l5-swagger:generate
- Akan terbentuk file /storage/api-docs/api-docs/json

```
{ } api-docs.json U X
storage > api-docs > { } api-docs.json > ...
1  {
2    .... "openapi": "3.0.0",
3    .... "info": {
4      .... "title": "Laravel API Documentation",
5      .... "description": "Made with ❤ by STMIK AMIK Bandung.",
6      .... "termsOfService": "http://swagger.io/terms/",
7      .... "contact": {
8        .... "email": "akatgelar@gmail.com"
9      .... },
10     .... "license": {
11       .... "name": "Apache 2.0",
12       .... "url": "http://www.apache.org/licenses/LICENSE-2.0.html"
13     .... },
14     .... "version": "1.0.0"
15   .... },
16   .... "paths": {
17     .... "/api/login": {
18       .... "post": {
19         .... "tags": [
20           .... "Auth"
21         .... ],

```

Buka url <http://localhost:8000/>



<http://localhost:8000/docs/api-docs.json>

Explore

Laravel API Documentation

1.0.0 OAS 3.0

<http://localhost:8000/docs/api-docs.json>

Made with ❤ by STMIK AMIK Bandung.

Authorize

Auth

POST /api/login

POST /api/register

Product

GET /api/product

POST /api/product

GET /api/product/{id}

PUT /api/product/{id}

DELETE /api/product/{id}

Login



Auth

POST

/api/login

^

Login

Parameters

Cancel

Reset

No parameters

Request body

multipart/form-data

^

email

string

email

☒

Send empty value

password

string

password

☒

Send empty value

Execute

Responses

Code

Description

Links

Header Token



Authorize



Filter by tag

Auth

POST /api/login

POST /api/register

Product

GET /api/product

POST /api/product

GET /api/product/{id}

PUT /api/product/{id}

DELETE /api/product/{id}

Available authorizations

x

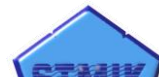
bearerAuth (http, Bearer)

Value:

Authorize

Close

Try it out



GET /api/product/{id}

Get data by id

Parameters

Cancel

Name	Description
------	-------------

id * required

number
(path)

1

Execute

Clear

Responses

Curl

```
curl -X 'GET' \
  'http://localhost:8000/api/product/1' \
  -H 'accept: application/json' \
  -H 'X-CSRF-TOKEN: '
```



Request URL

http://localhost:8000/api/product/1

Server response

Code Details

Response Data



Responses

Curl

```
curl -X 'GET' \
  'http://localhost:8000/api/product' \
  -H 'accept: application/json' \
  -H 'X-CSRF-TOKEN: '
```



Request URL

```
http://localhost:8000/api/product
```

Server response

Code

Details

200

Response body

```
{
  "success": true,
  "message": "Get Data Successfull",
  "data": [
    {
      "id": 1,
      "name": "kertas a4",
      "image": "",
      "description": "kertas 1 rim",
      "type": "alat kantor",
      "price": 40000,
      "created_at": "2023-05-24T11:27:31.000000Z",
      "updated_at": "2023-05-24T11:27:33.000000Z",
      "created_date_indo": "Rabu, 24 Mei 2023 11:27:31"
    },
    {
      "id": 3,
      "name": "pensil",
      "image": "",
      "description": "",
      "type": "",
      "price": 0,
      "created_at": null,
      "updated_at": null,
      "created_date_indo": "Selasa, 13 Juni 2023 21:17:54"
    }
  ]
}
```



Download