

BE NodeJS


Tugas: Validation Pagination

Leonardho R. Sitanggang

Soal :

Lakukan request terhadap endpoint GET /products dengan parameter query seperti ini

1.

 #57-Sanbercode-NodeJS / Products / **Find All Products**

GET

⌵

http://localhost:3000/api/products?page=1&limit=1

Params ● Authorization Headers (6) Body Scripts Tests Settings

Query Params

<input type="checkbox"/>	Key	Value
<input checked="" type="checkbox"/>	page	1
<input checked="" type="checkbox"/>	limit	1
<input type="checkbox"/>	search	sepatu
	Key	Value

2.

GET
⌵
http://localhost:3000/api/products?page=2&limit=1

Params
Authorization
Headers (6)
Body
Scripts
Tests
Settings

Query Params

<input type="checkbox"/>	Key	Value
<input checked="" type="checkbox"/>	page	2
<input checked="" type="checkbox"/>	limit	1
<input type="checkbox"/>	search	sepatu
	Key	Value

3.

 #57-Sanbercode-NodeJS / Products / Find All Products

GET
⌵
http://localhost:3000/api/products?page=5&limit=10

Params
Authorization
Headers (6)
Body
Scripts
Tests
Settings

Query Params

<input type="checkbox"/>	Key	Value
<input checked="" type="checkbox"/>	page	5
<input checked="" type="checkbox"/>	limit	10
<div>⋮</div> <input type="checkbox"/>	search	sepatu
	Key	Value

Lakukan request terhadap endpoint POST /products dengan request body seperti ini

1.

```

{
  "name": "Kemeja Variant 1",
  "description": "Deskripsi kemeja variant 1",
  "images":
["https://res.cloudinary.com/five-code/image/upload/v1718805645/fxuurm45mt5talry7a29.png", "sepatu2.jpg", "sepatu3.jpg"],
  "price": 20000,

```

```
"qty": 120,  
"category": "6673eed48865a2d1991420bb",  
"slug": "variant-1-slug"  
},
```

2.

```
{  
  "name": "Kemeja Variant 2",  
  "description": "Deskripsi kemeja variant 2",  
  "images":  
  ["https://res.cloudinary.com/five-code/image/upload/v1718805645/fxuurm45mt5tal  
ry7a29.png", "sepatu2.jpg", "sepatu3.jpg"],  
  "price": 25000,  
  "qty": 150,  
  "category": "6673eed48865a2d1991420bb",  
  
},
```

3.

```
{  
  "name": "Kemeja Variant 3",  
  "description": "Deskripsi kemeja variant 3",  
  "slug": "variant-3-slug"  
}
```

- Pastikan sudah menyelesaikan terlebih dahulu langkah - langkah pada bagian Materi dan fiturnya berfungsi
- Pelajari dan analisa setiap kasus yang ada pada Latihan ini
- Buatlah ringkasan tentang 6 soal diatas sesuai dengan apa yang kamu pahami
- Tulis hasil ringkasan di Google Docs, kumpulkan **link dokumen nya saja**.

Jawab :

tugas-validation-pagination - Run results
 Ran today at 21:18:49 · [View all runs](#)

Source	Environment	Iterations	Duration	All tests	Avg. Resp. Time
Runner	none	1	4s 84ms	0	626 ms

All Tests Passed (0) Failed (0) Skipped (0)

Iteration 1

POST Product With Slug
 http://localhost:3000/api/products 201 Created

No tests found

POST Product Without Slug
 http://localhost:3000/api/products 201 Created

No tests found

POST Product Without Price
 http://localhost:3000/api/products 400 Bad Request

No tests found

GET All Products
 http://localhost:3000/api/products?page=1&limit=1 200 OK

No tests found

GET All Products (Page 2 Limit 1)
 http://localhost:3000/api/products?page=2&limit=1 200 OK

No tests found

GET All Products (Page 5 Limit 10)
 http://localhost:3000/api/products?page=5&limit=10 200 OK

No tests found

1 **POST** tugas-validation-pagination / Products / **Product With Slug**

Response Headers Request

Pretty

```

1 {
2   "data": {
3     "name": "Kemeja Variant 1",
4     "description": "Deskripsi kemeja variant 1",
5     "images": [
6       "https://res.cloudinary.com/five-code/im
7       "sepatu2.jpg",
8       "sepatu3.jpg"
9     ],
10    "price": 20000,
11    "qty": 120,
12    "slug": "variant-1-slug",
13    "category": "6673eed48865a2d1991420bb",
14    "_id": "673363ca6a2a860741ebf053",
15    "createdAt": "2024-11-12T14:18:50.174Z",
16    "updatedAt": "2024-11-12T14:18:50.174Z",
17    "__v": 0
18  },
19  "message": "Success create product"
20 }

```

Pada kasus ini, menggunakan endpoint dengan method POST untuk mengirimkan data product dengan parameter nama, description, images, price, qty, slug, dan category. Hasil dari response body seperti pada gambar diatas. menampilkan beberapa object seperti request yang telah dikirim serta object createdAt dan updatedAt.

tugas-validation-pagination - Run results
 Ran today at 21:18:49 · [View all runs](#)

Source	Environment	Iterations	Duration	All tests	Avg. Resp. Time
Runner	none	1	4s 84ms	0	626 ms

All Tests Passed (0) Failed (0) Skipped (0)

Iteration 1

POST Product With Slug

http://localhost:3000/api/products

201 Created

No tests found

POST Product Without Slug

http://localhost:3000/api/products

201 Created

No tests found

POST Product Without Price

http://localhost:3000/api/products

400 Bad Request

No tests found

GET All Products

http://localhost:3000/api/products?page=1&limit=1

200 OK

No tests found

GET All Products (Page 2 Limit 1)

http://localhost:3000/api/products?page=2&limit=1

200 OK

No tests found

GET All Products (Page 5 Limit 10)

http://localhost:3000/api/products?page=5&limit=10

200 OK

No tests found

1 POST tugas-validation-pagination / Products / Product Without Slug

Response Headers Request

Pretty

```

1 {
2   "data": {
3     "name": "Kemeja Variant 2",
4     "description": "Deskripsi kemeja variant 2",
5     "images": [
6       "https://res.cloudinary.com/five-code/ima
7       sepatu2.jpg",
8       "sepatu3.jpg"
9     ],
10    "price": 25000,
11    "qty": 150,
12    "category": "6673eed48865a2d1991420bb",
13    "_id": "673363ca6a2a860741ebf055",
14    "createdAt": "2024-11-12T14:18:50.233Z",
15    "updatedAt": "2024-11-12T14:18:50.233Z",
16    "slug": "kemeja-variant-2",
17    "__v": 0
18  },
19  "message": "Success create product"
20 }
  
```

Pada kasus ini, menggunakan endpoint dengan method POST untuk mengirimkan data product dengan parameter nama, description, images, price, qty, dan category. Hasil dari response body seperti pada gambar diatas. menampilkan beberapa object seperti request yang telah dikirim serta object createdAt dan updatedAt. Selain itu ditemukan juga slug yang merupakan generated value yang sebelumnya telah ditambahkan di model, jika pada request pengguna tidak memberikan input slug. Maka aplikasi akan membuat slug secara otomatis.

tugas-validation-pagination - Run results

Ran today at 21:18:49 · [View all runs](#)

Source	Environment	Iterations	Duration	All tests	Avg. Resp. Time
Runner	none	1	4s 84ms	0	626 ms

All Tests Passed (0) Failed (0) Skipped (0)

Iteration 1

POST Product With Slug

http://localhost:3000/api/products

201 Created

No tests found

POST Product Without Slug

http://localhost:3000/api/products

201 Created

No tests found

POST Product Without Price

http://localhost:3000/api/products

400 Bad Request

No tests found

1 **POST** tugas-validation-pagination / Products / Product Without Price

Response Headers Request

Pretty

```
1 {
2   "data": [
3     "qty is a required field"
4   ],
5   "message": "Failed create product"
6 }
```

Pada kasus ini, menggunakan endpoint dengan method POST untuk mengirimkan data product dengan parameter nama, description, dan slug. Hasil dari response body seperti pada gambar diatas. menampilkan sebuah array validasi gagal. Dimana pada model dari product telah mewajibkan kolom qty untuk diisi (pada request tidak terdapat qty). Agar dapat berhasil menginput data, qty harus disertakan pada request

tugas-validation-pagination - Run results
 Ran today at 21:18:49 · [View all runs](#)

Source	Environment	Iterations	Duration	All tests	Avg. Resp. Time
Runner	none	1	4s 84ms	0	626 ms

All Tests Passed (0) Failed (0) Skipped (0)

Iteration 1

POST Product With Slug
 http://localhost:3000/api/products 201 Created
 No tests found

POST Product Without Slug
 http://localhost:3000/api/products 201 Created
 No tests found

POST Product Without Price
 http://localhost:3000/api/products 400 Bad Request
 No tests found

GET All Products
 http://localhost:3000/api/products?page=1&limit=1 200 OK
 No tests found

GET All Products (Page 2 Limit 1)
 http://localhost:3000/api/products?page=2&limit=1 200 OK
 No tests found

GET All Products (Page 5 Limit 10)
 http://localhost:3000/api/products?page=5&limit=10 200 OK
 No tests found

Response Headers Request

Pretty

```

1  {
2    "data": [
3      {
4        "_id": "673363ca6a2a860741ebf055",
5        "name": "Kemeja Variant 2",
6        "description": "Deskripsi kemeja variant 2",
7        "images": [
8          "https://res.cloudinary.com/five-code/ima
9          sepatu2.jpg",
10         sepatu3.jpg"
11       ],
12       "price": 25000,
13       "qty": 150,
14       "category": null,
15       "createdAt": "2024-11-12T14:18:50.233Z",
16       "updatedAt": "2024-11-12T14:18:50.233Z",
17       "slug": "kemeja-variant-2",
18       "__v": 0
19     }
20   ],
21   "message": "Success get all products",
22   "page": 1,
23   "limit": 1,
24   "total": 2,
25   "totalPages": 2
26 }
  
```

Pada kasus ini, menggunakan endpoint dengan method GET untuk mengambil data pada halaman pertama product. Dengan jumlah maksimal item yang ditampilkan perhalaman yaitu 1 item. Halaman yang dikunjungi, total keseluruhan halaman, serta total keseluruhan item juga dicantumkan pada response. Sehingga response ini dapat digunakan secara iterative.

Secara urutan keseluruhan (tanpa pagination) item yang ditampilkan merupakan item pertama (index 0) yang ada pada list product yang ingin dicari

tugas-validation-pagination - Run results

Ran today at 21:18:49 · [View all runs](#)

Source	Environment	Iterations	Duration	All tests	Avg. Resp. Time
Runner	none	1	4s 84ms	0	626 ms

All Tests
Passed (0)
Failed (0)
Skipped (0)

Iteration 1
1
GET tugas-validation-pagination / Products / All Products (Page 2 Limit 1)

POST Product With Slug
http://localhost:3000/api/products
201 Created

No tests found

POST Product Without Slug
http://localhost:3000/api/products
201 Created

No tests found

POST Product Without Price
http://localhost:3000/api/products
400 Bad Request

No tests found

GET All Products
http://localhost:3000/api/products?page=1&limit=1
200 OK

No tests found

GET All Products (Page 2 Limit 1)
http://localhost:3000/api/products?page=2&limit=1
200 OK

No tests found

GET All Products (Page 5 Limit 10)
http://localhost:3000/api/products?page=5&limit=10
200 OK

No tests found

Response
Headers
Request

Pretty

```

1 {
2   "data": [
3     {
4       "_id": "673363ca6a2a860741ebf053",
5       "name": "Kemeja Variant 1",
6       "description": "Deskripsi kemeja variant 1",
7       "images": [
8         "https://res.cloudinary.com/five-code/imag
9         sepatu2.jpg",
10        "sepatu3.jpg"
11      ],
12      "price": 20000,
13      "qty": 120,
14      "slug": "variant-1-slug",
15      "category": null,
16      "createdAt": "2024-11-12T14:18:50.174Z",
17      "updatedAt": "2024-11-12T14:18:50.174Z",
18      "__v": 0
19    }
20  ],
21  "message": "Success get all products",
22  "page": 2,
23  "limit": 1,
24  "total": 2,
25  "totalPages": 2
26  }

```

Pada kasus ini, menggunakan endpoint dengan method GET untuk mengambil data pada halaman kedua product. Dengan jumlah maksimal item yang ditampilkan perhalaman yaitu 1 item. Halaman yang dikunjungi, total keseluruhan halaman, serta total keseluruhan item juga dicantumkan pada response. Sehingga response ini dapat digunakan secara iterative.

Secara urutan keseluruhan (tanpa pagination) item yang ditampilkan merupakan item kedua (index 1) yang ada pada list product yang ingin dicari

tugas-validation-pagination - Run results
 Ran today at 21:18:49 · [View all runs](#)

Source	Environment	Iterations	Duration	All tests	Avg. Resp. Time
Runner	none	1	4s 84ms	0	626 ms

All Tests Passed (0) Failed (0) Skipped (0)

Iteration 1

POST Product With Slug

http://localhost:3000/api/products 201 Created

No tests found

POST Product Without Slug

http://localhost:3000/api/products 201 Created

No tests found

POST Product Without Price

http://localhost:3000/api/products 400 Bad Request

No tests found

GET All Products

http://localhost:3000/api/products?page=1&limit=1 200 OK

No tests found

GET All Products (Page 2 Limit 1)

http://localhost:3000/api/products?page=2&limit=1 200 OK

No tests found

GET All Products (Page 5 Limit 10)

http://localhost:3000/api/products?page=5&limit=10 200 OK

No tests found

GET tugas-validation-pagination / Products / All Products (Page 5 Limit 10)

Response Headers Request

Pretty

```

1 {
2   "data": [],
3   "message": "Success get all products",
4   "page": 5,
5   "limit": 10,
6   "total": 2,
7   "totalPages": 1
8 }
```

Pada kasus ini, menggunakan endpoint dengan method GET untuk mengambil data pada halaman kelima product. Dengan jumlah maksimal item yang ditampilkan perhalaman yaitu 10 item. Halaman yang dikunjungi, total keseluruhan halaman, serta total keseluruhan item juga dicantumkan pada response. Sehingga response ini dapat digunakan secara iterative.

Namun tidak ada data yang ditemukan dan ditampilkan pada array. Hal ini dikarenakan secara urutan keseluruhan (tanpa pagination) item yang ditampilkan merupakan item yang berada pada urutan 41-50 (index 40-49) yang ada pada list product yang ingin dicari.

Sementara berdasarkan object total dan totalPage. Ditemukan total item hanya 2. Dan halaman valid yang masih memiliki item berada maksimal pada halaman 1.

Repository :

- Github <https://github.com/FlazeFy/sanbercode-nodejs-xpress/tree/main/tugas-validation-pagination>
- Postman Documentation <https://documenter.getpostman.com/view/18882129/2sAY547Jug>

Refrensi :

- Materi Pekan 2.5 Moongose
<https://sanbercode.com/bootcamp/class/684/materi/tugas-2-5-mongoose/74e9a398-83f3-4e62-97be-38aa517919ce>
- Materi Pekan 3.1 Validation, Mongoose Middleware dan Pagination Query
<https://sanbercode.com/bootcamp/class/684/materi/3-1-validation-mongoose-middleware-dan-pagination-query/84f7aa1a-485b-4380-b3ea-7eb5415497f2#>