

Testes

Iteracao 2 (testes de integração [Postman] e testes unitários [Mocha])

- medicoObtemReceitas
- utenteNaoObtemReceitas
- medicoCriaPrescricoes
- obtemSeusUtentes
- naoObtemOutrosUtentes
- verificaNIFExistenteNoRegisto
- apenasMedicoCriaReceita
- receitaTemMedicoCorretoAssociado
- utenteVeSuasPrescricoesApenas
- medicaAlteraPrescricaoQdNAviada
- farmaceuticoAviaTotalmenteReceita
- farmaceuticoAviaParcialmenteReceita
- prescricoesAteDataX
- prescricoesPorAviarUtente

Testes de Integração (Restfull - API)

Ferramenta utilizada: POSTMAN (cliente REST)

Testes efetuados: todos os testes mencionados abaixo sendo que para efeitos de documentação demonstramos um por acção rest implementada (get, post, put, delete).

- GET

api/Medicamento

GET

http://localhost:50647/api/Medicamento

Params

Send

Authorization

Headers (2)

Body

Pre-request Script

Tests

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

//----- GET TEST ----

tests["Codigo com estado 200"]=responseCode.code===200;

tests["Codigo para falhar"]= !(responseCode.code===400 || responseCode.code===404);

var obj= '{"id":1,"nome":"Aspirina Prevent"}'; //presente na base de dados

var obj2= '{"id":2,"nome":"Aspirina Prevent"}'; //não presente na base de dados

var jsonObj=JSON.parse(obj);

var jsonObj=JSON.parse(obj2);

tests["Lista Medicamentos (json contem medicamento com id)"]= responseBody.has(obj);

tests["Lista Medicamentos (nao contem medicamento com id)"]=!responseBody.has(obj2);

Test scripts are written in
run after the response is

Learn more about tests

SNIPPETS

Clear a global variable

Clear an environment var

Response body: Contains

Response body: Convert:
Object

Response body: Is equal t

Response body: JSON vali

Response headers: Conte

Body

Cookies

Headers (6)

Test Results (4/4)

Status: 200

All

Passed

Failed

PASS

Codigo com estado 200

PASS

Codigo para falhar

PASS

Lista Medicamentos (json contem medicamento com id)

PASS

Lista Medicamentos (nao contem medicamento com id)

api/Farmaco

api/Posologia

api/Apresentacao

api/Medicamento/{id}

api/Posologia/{id}

api/Apresentacao/{id}

GET ▾

http://localhost:50647/api/Apresentacao/1

Params

Authorization

Headers (2)

Body

Pre-request Script

Tests ●

```

1 //----- GET TEST ({id}) -----
2
3 tests["Codigo com estado 200"]=responseCode.code===200;
4 tests["Codigo para falhar"]= !(responseCode.code===400 || responseCode.code===404);
5
6 var obj= '{"id":1,"forma_adm":"Xarope","concentracao":"25mg","qtd":"1ml"}'; //presente
7   na base de dados
8 var obj2= '{"id":2,"nome":"Aspirina Prevent"}'; //não presente na base de dados
9 var jsonObj=JSON.parse(obj);
10 var jsonObj2=JSON.parse(obj2);
11
12 tests["Lista Apresentacoes (json contem id 1)"]= responseBody.has(obj);
13 tests["Lista Apresentacoes (nao contem id 1)"]=!responseBody.has(obj2);
14
15

```

Test scrip

run after

Learn m

SNIPPETS

Clear a g

Clear an

Respons

Respons

Object

Respons

Respons

Respons

Body

Cookies

Headers (6)

Test Results (4/4)

All

Passed

Failed

PASS

Codigo com estado 200

PASS

Codigo para falhar

PASS

Lista Apresentacoes (json contem id 1)

PASS

Lista Apresentacoes (nao contem id 1)

api/Medicamento/{id}/Apresentacoes

api/Medicamentos/{id}/Posologias

api/Farmaco/{id}/Medicamentos

api/Farmaco/{id}/Posologias

api/Farmaco/{id}/Apresentacoes

api/Medicamento/nome={nome}

- POST

api/Medicamento

GET

http://localhost:50647/api/Medicamento

Params

Send

Authorization

Headers (2)

Body

Pre-request Script

Tests

1 //----- POST TEST -----
2
3 var obj= '{"id":8,"nome":"TESTE"}'; //não presente na base de dados
4 var jsonObj=JSON.parse(obj);
5
6 tests["Lista Medicamentos ainda nao contem medicamento teste"]=!responseBody.has(obj);
7
8
9
10

Test scripts are written in JavaScript and run after the response is received.
[Learn more about tests](#)

SNIPPETS
[Clear a global variable](#)
[Clear an environment variable](#)
[Response body: Contains](#)
[Response body: Convert to Object](#)
[Response body: Is equal to](#)
[Response body: JSON value](#)
[Response headers: Contains](#)

Body

Cookies

Headers (6)

Test Results (1/1)

Status: 200

All

Passed

Failed

PASS

Lista Medicamentos ainda nao contem medicamento teste

POST

http://localhost:50647/api/Medicamento

Authorization

Headers (2)

Body

Pre-request Script

Tests

1 //----- POST TEST -----
2
3 tests["POST com sucesso"]= responseCode.code===201 || responseCode.code===202;
4
5 var obj= '{"id":8,"nome":"TESTE"}'; // presente na base de dados
6 var jsonObj=JSON.parse(obj);
7
8
9 tests["Lista Medicamentos contem medicamento teste"]=responseBody.has(obj);
10
11
12
13

Body

Cookies

Headers (7)

Test Results (1/2)

All

Passed

Failed

PASS

POST com sucesso

http://localhost:50647

http://localhost:50647

+

...

GET

http://localhost:50647/api/Medicamento/

Authorization

Headers (2)

Body

Pre-request Script

Tests

1

//----- POST TEST -----

2

3

4

tests["GET com sucesso"] = responseCode.code===200;

5

6

var obj= '{"id":8,"nome":"TESTE"}'; // presente na base de dados

7

var jsonObj=JSON.parse(obj);

8

9

tests["Lista Medicamentos contem medicamento teste"]=responseBody.has(obj);

10

Body

Cookies

Headers (6)

Test Results (2/2)

All

Passed

Failed

PASS

GET com sucesso

PASS

Lista Medicamentos contem medicamento teste

api/Farmaco

api/Posologia

api/Apresentacao

api/Account/

api/Account/Login

api/Account/Token

POST http://localhost:50647/api/Medicamento

Authorization Headers (2) Body Pre-request Script Tests

```
1
2 tests["Permissoes sem token inserido"] = responseCode.code===400 || responseCode.code===401 || responseCode.code===402;
3
```

Body Cookies Headers (6) Test Results (1/1)

All Passed Failed

PASS Permissoes sem token inserido

http://localhost:50647 + ... Test

GET http://localhost:50647/api/Medicamento/11 Params Send

Authorization Headers (2) Body Pre-request Script Tests

Key	Value	Description	...	Bulk Ed
<input checked="" type="checkbox"/> Content-Type	application/json			
<input type="checkbox"/> Authorization	Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJqdGkiOiI3Y2Y2O...			
<input type="checkbox"/>				
New key	Value	Description		

Body Cookies Headers (6) Test Results (1/5) Status: 200 OK

Pretty Raw Preview

```
{"id":11,"nome":null}
```

- PUT/DELETE

api/Medicamento/{id}

api/Farmaco/{id}

api/Posologia/{id}

api/Apresentacao/{id}

Put

http://localhost:50647

http://localhost:50647

+

...

Test

GET

http://localhost:50647/api/Posologia

Params

Send

AuthorizationHeaders (2)BodyPre-request ScriptTests

1 // put

2

3 /*tests["PUT com sucesso"]= responseCode.code===201 || responseCode.code===202 ||

responseCode.code===204;

4 */

5 var obj= '{"id":1,"principio_ativo":"Acido acetil-salicilico"}';

6 var jsonObj=JSON.parse(obj);

7 tests["Posologia PUT - antigo nao encontrado "]=responseBody.has(obj);

8

9 var obj2= '{"id":1,"principio_ativo":"TESTE Acido acetil-salicilico"}';

10 var jsonObj=JSON.parse(obj2);

11 tests["Farmaco PUT - objeto modificado encontrado "]=responseBody.has(obj2);

12

Test scripts are written i

run after the response i:

[Learn more about tests](#)

SNIPPETS

[Clear a global variable](#)

[Clear an environment va](#)

[Response body: Contain](#)

[Response body: Convert](#)

[Object](#)

[Response body: Is equal](#)

[Response body: JSON va](#)

[Response headers: Cont](#)

BodyCookiesHeaders (6)Test Results (0/2)Status: 2

PrettyRawPreview

[{"id":1,"dose":100,"via_administracao":"via oral","intervalo_tempo_horas":8,"periodo_tempo_dias":3}, {"id":2,"dose":400,"via_administracao":"via oral","intervalo_tempo_horas":6,"periodo_tempo_dias":5}, {"id":3,"dose":40,"via_administracao":"via oral","intervalo_tempo_horas":8,"periodo_tempo_dias":8}, {"id":4,"dose":100,"via_administracao":"via injeccao","intervalo_tempo_horas":24,"periodo_tempo_dias":3}, {"id":5,"dose":10,"via_administracao":"via retal","intervalo_tempo_horas":6,"periodo_tempo_dias":7}]

http://localhost:50647

http://localhost:50647

+

...

Test

PUT

http://localhost:50647/api/Farmaco/1

Params

Send

AuthorizationHeaders (2)BodyPre-request ScriptTests

form-data

x-www-form-urlencoded

raw

binary

JSON (application/json)

1 {

2 "FarmacoId": "1",

3 "principio_ativo": "TESTE Acido acetil-salicilico"

4 }

All Passed Failed

PASS PUT com sucesso

http://localhost:50647 http://localhost:50647 + ... Test

GET http://localhost:50647/api/Farmaco/ Pa

Authorization Headers (2) Body Pre-request Script Tests

```
1 // put
2
3 /*tests["PUT com sucesso"]= responseCode.code===201 || responseCode.code===202 ||
   responseCode.code===204;
4 */
5 var obj= '{"id":1,"principio_ativo":"Acido acetil-salicilico"}';
6 var jsonObj=JSON.parse(obj);
7 tests["Posologia PUT - antigo nao encontrado "]=responseBody.has(obj);
8
9 var obj2= '{"id":1,"principio_ativo":"TESTE Acido acetil-salicilico"}';
10 var jsonObj=JSON.parse(obj2);
11 tests["Farmaco PUT - objeto modificado encontrado "]=responseBody.has(obj2);
12
```

All Passed Failed

FAIL Posologia PUT - antigo nao encontrado

PASS Farmaco PUT - objeto modificado encontrado

PUT

http://localhost:50647/api/Farmaco/1

Authorization

Headers (2)

Body

Pre-request Script

Tests

form-data

x-www-form-urlencoded

raw

binary

JSON (application/json)

1

{

2

"FarmacoId": "1",

3

"principio_ativo": "TESTEAcido acetil-salilico"

4

}

Body

Cookies

Headers (6)

Test Results (0/1)

Pretty

Raw

Preview

JSON

1

{

2

"id": 1,

3

"principio_ativo": "TESTEAcido acetil-salilico"

4

}

Delete

The screenshot shows the Postman interface with the following details:

- Method:** GET
- URL:** http://localhost:50647/api/Apresentacao
- Authorization:** No Auth
- Headers:** 2
- Body:** JSON (selected)
- Test Results:** 0/2

The JSON body contains an array of 8 objects, each representing a medication item with fields: id, forma_adm, concentracao, and qtd.

```
[{"id":1,"forma_adm":"Xarope","concentracao":"25mg","qtd":"1ml"}, {"id":2,"forma_adm":"Comprimido","concentracao":"10mg","qtd":"10ml"}, {"id":3,"forma_adm":"Supositorio","concentracao":"10mg","qtd":"2ml"}, {"id":4,"forma_adm":"Comprimido","concentracao":"25mg","qtd":"10ml"}, {"id":5,"forma_adm":"Comprimido","concentracao":"25mg","qtd":"8ml"}, {"id":6,"forma_adm":"Comprimido","concentracao":"25mg","qtd":"8ml"}, {"id":7,"forma_adm":"Liquido","concentracao":"50mg","qtd":"3ml"}, {"id":8,"forma_adm":"Comprimido","concentracao":"25mg","qtd":"8ml"}]
```

Authorization

Tests ●

No Auth

Body

Headers (6)

Pretty

Preview

```
[{"id":1,"forma_adm":"Xarope","concentracao":"25mg","qtd":"1ml"},
{"id":2,"forma_adm":"Comprimido","concentracao":"10mg","qtd":"10ml"},
{"id":3,"forma_adm":"Supositorio","concentracao":"10mg","qtd":"2ml"},
{"id":4,"forma_adm":"Comprimido","concentracao":"25mg","qtd":"10ml"},
{"id":5,"forma_adm":"Comprimido","concentracao":"25mg","qtd":"8ml"},
{"id":6,"forma_adm":"Comprimido","concentracao":"25mg","qtd":"8ml"},
{"id":7,"forma_adm":"Liquido","concentracao":"50mg","qtd":"3ml"},
{"id":8,"forma_adm":"Comprimido","concentracao":"25mg","qtd":"8ml"}]
```

Max Rows: 1000

[illegible]

GET

http://localhost:50647/api/Posologia/

Params

Send

Save

AuthorizationHeaders (2)BodyPre-request ScriptTestsCod

1//----- DELETE tests -----

2

3var obj= '{"id":2,"dose":400,"via_administracao":"via oral","intervalo_tempo_horas":6,"periodo_tempo_dias":5}';

4var jsonObj=JSON.parse(obj);

5tests["Posologia DELETE - objeto existente em pre-delete "]=responseBody.has(obj);

6

7/*

8tests["DELETE com sucesso"]= responseCode.code===201 || responseCode.code===202 || responseCode.code===204;

9

10var obj= '{"id":3,"dose":40,"via_administracao":"via oral","intervalo_tempo_horas":8,"periodo_tempo_dias":8}';

11var jsonObj=JSON.parse(obj);

12tests["Posologia DELETE - objeto nao existente pos-delete "]=responseBody.has(obj);

13*/

Test scripts are written in JavaScript, and are run after the response is received.

Learn more about tests

SNIPPETS

Clear a global variable

Clear an environment variable

Response body: Contains string

Response body: Convert XML body to a JSON Object

Response body: Is equal to a string

Response body: JSON value check

Response headers: Content-Type header check

BodyCookiesHeaders (6)Test Results (1/1)Status: 200 OKTime: 46 ms

AllPassedFailed

PASSPosologia DELETE - objeto existente em pre-delete

DELETE

http://localhost:50647/api/Posologia/2

Params

Send

AuthorizationHeaders (2)BodyPre-request ScriptTests

1//----- DELETE tests -----

2

3/*

4var obj= '{"id":2,"dose":400,"via_administracao":"via oral","intervalo_tempo_horas":6,"periodo_tempo_dias":5}';

5var jsonObj=JSON.parse(obj);

6tests["Posologia DELETE - objeto existente em pre-delete "]=responseBody.has(obj);

7*/

8tests["DELETE com sucesso"]= responseCode.code===200;

9

10var obj= '{"id":2,"dose":400,"via_administracao":"via oral","intervalo_tempo_horas":6,"periodo_tempo_dias":5}';

11var jsonObj=JSON.parse(obj);

12tests["Posologia DELETE FAIL - objeto nao existente pos-delete "]=!responseBody.has(obj);

13

Test scripts are written in JavaScript, and are run after the response is received.

Learn more about tests

SNIPPETS

Clear a global variable

Clear an environment variable

Response body: Contains string

Response body: Convert XML body to a JSON Object

Response body: Is equal to a string

Response body: JSON value check

Response headers: Content-Type header check

BodyCookiesHeaders (6)Test Results (2/2)Status: 200 OK

AllPassedFailed

PASSDELETE com sucesso

PASSPosologia DELETE FAIL - objeto nao existente pos-delete

GET ▼ http://localhost:50647/api/Apresentacao Params Send

Authorization Headers (2) Body Pre-request Script Tests ●

Type No Auth ▼

Body Cookies Headers (6) Test Results (2/2) Status

Pretty Raw Preview

```
[{"id":3,"forma_adm":"Supositorio","concentracao":"10mg","qtd":"2ml"}, {"id":5,"forma_adm":"Comprimido","concentracao":"25mg","qtd":"8ml"}, {"id":6,"forma_adm":"Comprimido","concentracao":"25mg","qtd":"8ml"}, {"id":7,"forma_adm":"Liquido","concentracao":"50mg","qtd":"3ml"}, {"id":8,"forma_adm":"Comprimido","concentracao":"25mg","qtd":"8ml"}]
```

SQL Server Object Explorer

- SQL Server
 - (localdb)\MSSQLLocalDB (SQL Server 13.0.4001)
 - Databases
 - System Databases
 - MedicamentosAPIContext-ca95e933-595
 - Tables
 - System Tables
 - External Tables
 - dbo._EFMigrationsHistory
 - dbo.Apresentacao**
 - dbo.AspNetRoleClaims
 - dbo.AspNetRoles
 - dbo.AspNetUserClaims
 - dbo.AspNetUserLogins
 - dbo.AspNetUserRoles
 - dbo.AspNetUsers
 - dbo.AspNetUserTokens
 - dbo.Farmaco
 - dbo.Medicamento
 - dbo.Posologia
 - Views
 - System Views
 - Synonyms
 - Programmability
 - External Resources
 - Service Broker
 - Storage
 - Security
 - SementeContext-7e0f97ac-ea8a-48b6-8c
 - Security
 - Server Objects

dbo.Apresentacao [Data]

Max Rows: 1000

	Apresentac...	Farmacold	Medicame...	Posologia...	dosagem	forma_adm	quantidade
▶	3	3	5	5	10	Supositorio	2
	5	3	2	1	25	Comprimido	8
	6	3	3	3	25	Comprimido	8
	7	2	8	4	50	Liquido	3
	8	2	6	3	25	Comprimido	8
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Server Explorer | Toolbox | SQL Server Object Explorer | Connection Ready | (localdb)\MSSQLLocalDB | LAPTOP-VSM9ACD

Testes Unitários

testarGETMedicamentoOK()

testarGETIdMedicamentoCorretoDevolvido()

testarGETMedicamentoDevolveLista()

testarPUTMedicamentoOk()

testarPUTMedicamentoQuandoNaoExiste()

testarDELETEQuandoExiste()

testarDELETEQuandoNaoExiste()

testarDefinicaoBD()

testarMedicamentosCarregadosBD()