

**FGA 0238 - Testes de Software – T01****Semestre:** 2024.1**Nome:** João Eduardo Pereira Rabelo**Matrícula:** 180053299**Equipe:** NeverMind

Atividade 3 – Desenvolver Testes de Unidade

1 MEC Energia API

O projeto "MEC Energia API" no GitLab, desenvolvido pelo LAPPIS-UNB, é uma aplicação voltada para a gestão e monitoramento de dados de energia elétrica. A API oferece funcionalidades que permitem a coleta, processamento e disponibilização de informações relacionadas ao consumo de energia, buscando proporcionar maior eficiência e controle no uso dos recursos energéticos. O sistema pode ser integrado a outras plataformas e ferramentas, facilitando a análise e visualização de dados energéticos, com o objetivo de promover a sustentabilidade e a otimização do consumo de energia.

2 Cobertura de testes

Conforme detalhado na imagem abaixo, após ser feita a varredura do projeto, atualmente, se tem uma cobertura total de 59% acerca das suas classes e seus métodos, tendo espaço ainda para uma melhora na área de testes.

tariffs/serializers.py	80	2	2	0	98%
tariffs/urls.py	8	0	0	0	100%
tariffs/views.py	149	50	28	7	64%
universities/apps.py	4	0	0	0	100%
universities/models.py	134	69	34	1	45%
universities/recommendation.py	44	18	12	0	57%
universities/serializers.py	54	0	0	0	100%
universities/urls.py	5	0	0	0	100%
universities/views.py	159	98	14	1	36%
users/apps.py	4	0	0	0	100%
users/authentications.py	191	108	20	2	42%
users/managers.py	40	18	12	2	50%
users/models.py	145	69	26	2	48%
users/requests_permissions.py	32	5	8	3	80%
users/serializers.py	59	0	0	0	100%
users/urls.py	5	0	0	0	100%
users/views.py	101	51	14	1	48%
utils/cnpj_validator_util.py	29	0	15	0	100%
utils/date_util.py	8	0	0	0	100%
utils/email/send_email.py	33	22	0	0	33%
utils/email/templates_email/password_templates_email.py	14	9	0	0	36%
utils/email/valid_email.py	7	0	0	0	100%
utils/endpoints_util.py	8	3	0	0	62%
utils/energy_bill_util.py	84	1	14	1	98%
utils/subgroup_util.py	20	1	10	0	97%
utils/tariff_util.py	9	8	4	0	8%
utils/user/authentication.py	23	14	8	0	29%
utils/user/user_type_util.py	18	2	10	2	86%

TOTAL	2850	1063	481	45	59%
Coverage HTML written to dir reports/cov					

3 Método a ser testado

- Nome do Método: *generate_link_to_reset_password*
- O método “*generate_link_to_reset_password*” tem como objetivo gerar um link específico para a redefinição de senha de um usuário dentro do sistema MEC Energia. Esse link é construído com base no tipo de redefinição de senha que está sendo solicitada (*first_access*, *admin_reset* ou *user_reset*), e inclui um token de autenticação e o primeiro nome do usuário como parâmetros de consulta na URL.
- O método se encontra na classe de autenticação, não possui testes criados anteriormente, mas o método em si, pode ser acessada através do seguinte caminho: [mec-energia-api/utils/user/authentication.py](https://gitlab.com/JoaoEduardoP/mec-energia-api/-/blob/develop/utils/user/authentication.py)
- Pode ser visualizada no seguinte link do fork do projeto: https://gitlab.com/JoaoEduardoP/mec-energia-api/-/blob/develop/utils/user/authentication.py?ref_type=heads

```
30 def generate_link_to_reset_password(token, user_first_name, password_status):
31     if password_status == 'first_access':
32         endpoint = settings.MEC_ENERGIA_PASSWORD_ENDPOINT_FIRST_ACCESS
33     elif password_status == 'admin_reset':
34         endpoint = settings.MEC_ENERGIA_PASSWORD_ENDPOINT_ADMIN_RESET
35     elif password_status == 'user_reset':
36         endpoint = settings.MEC_ENERGIA_PASSWORD_ENDPOINT_USER_RESET
37     else:
38         raise ValueError("Invalid password_status")
39
40     endpoint_string = f'{settings.MEC_ENERGIA_URL}/{endpoint}/'
41     user_first_name = f'?nome={user_first_name}'
42     token_string = f'&token={token}'
43
44     return endpoint_string + user_first_name + token_string
```

4 Testes existentes

- O método apresentado não apresenta nenhuma cobertura de código até então, outras partes do projeto possuem alguma cobertura, mas pelo que foi verificado, essa classe de autenticação ainda não foi criada os testes de maneira coerente.
- Link para acesso do repositório: https://gitlab.com/JoaoEduardoP/mec-energia-api/-/blob/develop/utils/user/authentication.py?ref_type=heads

5 Tabela de decisões/condições

Tabela 01: Decisões e Condições

ID	Decisão (Linha)	Condição	Situação para verdadeiro	Situação para falso
01	31	password_status == 'first_access'	O status da senha que será redefinida é 'first_access'.	O status da senha que será redefinida é qualquer um diferente de 'first_access'.
02	33	password_status == 'admin_reset'	O status da senha que será redefinida é 'admin_reset'.	O status da senha que será redefinida é qualquer um diferente de 'admin_reset'.
03	35	password_status == 'user_reset'	O status da senha que será redefinida é 'user_reset'.	O status da senha que será redefinida é qualquer um diferente de 'user_reset'.
04	37	password_status != 'first_access' && password_status != 'admin_reset' && password_status != 'user_reset'	O status da senha que será redefinida é inválido (não atende a nenhuma das condições anteriores).	O status da senha que será redefinida é válido.

6 Tabelas verdade, pares de independência e combinações de condições MC/DC

Tabela 02: Tabela Verdade (MC/DC)

Teste	password_status == 'first_access'	password_status == 'admin_reset'	password_status == 'user_reset'	Resultado
00	Falso	Falso	Falso	Falso
01	Falso	Falso	Verdadeiro	Verdadeiro
02	Falso	Verdadeiro	Falso	Verdadeiro
03	Verdadeiro	Falso	Falso	Verdadeiro

- Pares de Independência
- Para password_status == 'first_access' -> 03 e 00
- Para password_status == 'admin_reset' -> 02 e 00
- Para password_status == 'user_reset' -> 01 e 00

7 Especificação dos Casos de Testes

Tabela 03: Casos de Teste

Casos de Teste	Entrada	Saída Esperada
1	<code>password_status != 'first_access'</code> <code>&&</code> <code>password_status != 'admin_reset'</code> <code>&&</code> <code>password_status != 'user_reset'</code>	Com estes parâmetros, a função irá retornar um valor de error como senha inválida.
2	<code>password_status != 'first_access'</code> <code>&&</code> <code>password_status != 'admin_reset'</code> <code>&&</code> <code>password_status == 'user_reset'</code>	Com estes parâmetros, a função vai retornar o endpoint do “user_reset”.
3	<code>password_status != 'first_access'</code> <code>&&</code> <code>password_status == 'admin_reset'</code> <code>&&</code> <code>password_status != 'user_reset'</code>	Com estes parâmetros, a função vai retornar o endpoint do “admin_reset”.
4	<code>password_status == 'first_access'</code> <code>&&</code> <code>password_status != 'admin_reset'</code> <code>&&</code> <code>password_status != 'user_reset'</code>	Com estes parâmetros, a função vai retornar o endpoint do “first_access”.

8 Implementação dos Casos de Teste

- Para os testes da classe de autenticação, foram implementados os seguintes testes, para cobertura do método em questão:
 - Caso de Teste 1

```
49 def test_generate_link_to_reset_password_invalid_status():
50     token = "dummy_token"
51     user_first_name = "John"
52     password_status = "invalid_status"
53
54     with pytest.raises(ValueError, match="Invalid password_status"):
55         generate_link_to_reset_password(token, user_first_name, password_status)
56
```

- Caso de Teste 2

```
36 v def test_generate_link_to_reset_password_user_reset(monkeypatch):
37     token = "dummy_token"
38     user_first_name = "John"
39     password_status = "user_reset"
40
41     monkeypatch.setattr(settings, "MEC_ENERGIA_URL", "http://example.com")
42     monkeypatch.setattr(settings, "MEC_ENERGIA_PASSWORD_ENDPOINT_USER_RESET", "user_reset")
43
44     link = generate_link_to_reset_password(token, user_first_name, password_status)
45     expected_link = "http://example.com/user_reset/?nome=John&token=dummy_token"
46
47     assert link == expected_link
```

- Caso de Teste 3

```
23 def test_generate_link_to_reset_password_admin_reset(monkeypatch):
24     token = "dummy_token"
25     user_first_name = "John"
26     password_status = "admin_reset"
27
28     monkeypatch.setattr(settings, "MEC_ENERGIA_URL", "http://example.com")
29     monkeypatch.setattr(settings, "MEC_ENERGIA_PASSWORD_ENDPOINT_ADMIN_RESET", "admin_reset")
30
31     link = generate_link_to_reset_password(token, user_first_name, password_status)
32     expected_link = "http://example.com/admin_reset/?nome=John&token=dummy_token"
33
34     assert link == expected_link
```

- Caso de Teste 4

```
10 def test_generate_link_to_reset_password_first_access(monkeypatch):
11     token = "dummy_token"
12     user_first_name = "John"
13     password_status = "first_access"
14
15     monkeypatch.setattr(settings, "MEC_ENERGIA_URL", "http://example.com")
16     monkeypatch.setattr(settings, "MEC_ENERGIA_PASSWORD_ENDPOINT_FIRST_ACCESS", "first_access")
17
18     link = generate_link_to_reset_password(token, user_first_name, password_status)
19     expected_link = "http://example.com/first_access/?none=John&token=dummy_token"
20
21     assert link == expected_link
```

- Para visualizar as alterações realizadas, pode-se acessar o link do commit feito:
<https://gitlab.com/JoaoEduardoP/mec-energia-api/-/commit/6c622b7a0395234de088ccbe997bbab0a4926478>
- Para visualizar de outra forma, apenas acessar o fork no seguinte link:
<https://gitlab.com/JoaoEduardoP/mec-energia-api>
 - Acessar a branch “Teste_Autenticacao” e o seguinte diretório para acessar o arquivo de testes criado: **tests/utils/user/test_authentication.py**

9 Análises e Resultados

- Realizado a chamada do “pytest” e feitas as varreduras nos códigos e seus respectivos testes, utilizando o seguinte comando: ***docker exec mec-energia-api bash -c "python -m pytest"***.
- Foram obtidos os seguintes resultados após serem criados os testes:
 - Sendo os números apresentados na análise, vistos seguindo o seguinte padrão:

```
----- coverage: platform linux, python 3.10.5-final-0 -----
Name                               Stmts   Miss Branch BrPart  Cover
-----
```

- Name: Caminho do arquivo que precisa ser testado.
- Stmts: Quantidade de Linhas de código
- Miss: Quantidade de linhas que não foram executadas durante os testes, ou seja, quantas linhas não foram cobertas pelo teste.
- Branch: Quantidade de ramos, ou seja, possíveis decisões que existem dentro do código.
- Cover: A porcentagem de código que foi testada no total

- Varredura antes dos testes criados:

tariffs/serializers.py	80	2	2	0	98%
tariffs/urls.py	8	0	0	0	100%
tariffs/views.py	149	50	28	7	64%
universities/apps.py	4	0	0	0	100%
universities/models.py	134	69	34	1	45%
universities/recommendation.py	44	18	12	0	57%
universities/serializers.py	54	0	0	0	100%
universities/urls.py	5	0	0	0	100%
universities/views.py	159	98	14	1	36%
users/apps.py	4	0	0	0	100%
users/authentications.py	191	108	20	2	42%
users/managers.py	40	18	12	2	50%
users/models.py	145	69	26	2	48%
users/requests_permissions.py	32	5	8	3	80%
users/serializers.py	59	0	0	0	100%
users/urls.py	5	0	0	0	100%
users/views.py	101	51	14	1	48%
utils/cnpj_validator_util.py	29	0	15	0	100%
utils/date_util.py	8	0	0	0	100%
utils/email/send_email.py	33	22	0	0	33%
utils/email/templates_email/password_templates_email.py	14	9	0	0	36%
utils/email/valid_email.py	7	0	0	0	100%
utils/endpoints_util.py	8	3	0	0	62%
utils/energy_bill_util.py	84	1	14	1	98%
utils/subgroup_util.py	20	1	10	0	97%
utils/tariff_util.py	9	8	4	0	8%
utils/user/authentication.py	23	14	8	0	29%
utils/user/user_type_util.py	18	2	10	2	86%

TOTAL	2850	1063	481	45	59%
Coverage HTML written to dir reports/cov					

- Varredura após os testes criados:

tariffs/serializers.py	80	2	2	0	98%
tariffs/urls.py	8	0	0	0	100%
tariffs/views.py	149	50	28	7	64%
universities/apps.py	4	0	0	0	100%
universities/models.py	134	69	34	1	45%
universities/recommendation.py	44	18	12	0	57%
universities/serializers.py	54	0	0	0	100%
universities/urls.py	5	0	0	0	100%
universities/views.py	159	98	14	1	36%
users/apps.py	4	0	0	0	100%
users/authentications.py	191	108	20	2	42%
users/managers.py	40	18	12	2	50%
users/models.py	145	69	26	2	48%
users/requests_permissions.py	32	5	8	3	80%
users/serializers.py	59	0	0	0	100%
users/urls.py	5	0	0	0	100%
users/views.py	101	51	14	1	48%
utils/cnpj_validator_util.py	29	0	15	0	100%
utils/date_util.py	8	0	0	0	100%
utils/email/send_email.py	33	22	0	0	33%
utils/email/templates_email/password_templates_email.py	14	9	0	0	36%
utils/email/valid_email.py	7	0	0	0	100%
utils/endpoints_util.py	8	3	0	0	62%
utils/energy_bill_util.py	84	1	14	1	98%
utils/subgroup_util.py	20	1	10	0	97%
utils/tariff_util.py	9	8	4	0	8%
utils/user/authentication.py	23	3	8	0	84%
utils/user/user_type_util.py	18	2	10	2	86%

TOTAL	2850	1052	481	45	59%
Coverage HTML written to dir reports/cov					



10 Links

Fork do projeto: <https://gitlab.com/JoaoEduardoP/mec-energia-api>

Commit da implementação: <https://gitlab.com/JoaoEduardoP/mec-energia-api/-/commit/6c622b7a0395234de088ccb997bbab0a4926478>