

## Transfer

/test/rest/external/transferExternal.test.js

### POST /transfers

/test/rest/external/transferExternal.test.js

🕒

29ms

📄

5

✓

5

✓

Quando informo valores válidos eu tenho sucesso (201 CREATED)

10ms

🕒

```
const postTransfer = require('../fixture/requisicoes/transferencias/postTransfer.json');
const resposta = await request(process.env.BASE_URL_REST)
  .post('/transfers')
  .set('Authorization', `Bearer ${token}`)
  .send(postTransfer);
expect(resposta.status).to.equal(201);

// Validação com um Fixture
const respostaEsperada = require('../fixture/respostas/quandoInformoValoresValidosEuTenhoSucesso.json');
expect(resposta.body)
  .excluding('date')
  .to.deep.equal(respostaEsperada);
```

✓

Quando tento transferir sem token, recebo erro (401 UNAUTHORIZED)

4ms

🕒

```
const postTransfer = require('../fixture/requisicoes/transferencias/postTransfer.json');
const resposta = await request(process.env.BASE_URL_REST)
  .post('/transfers')
  .send(postTransfer);
expect(resposta.status).to.equal(401);
expect(resposta.body).to.have.property('message', 'Token não fornecido.');
```

✓

Testando a regra relacionada a Saldo insuficiente

5ms

🕒

```
const postTransfer = require('../fixture/requisicoes/transferencias/postTransfer.json');
const resposta = await request(process.env.BASE_URL_REST)
  .post('/transfers')
  .set('Authorization', `Bearer ${token}`)
  .send(teste.postTransfer);
expect(resposta.status).to.equal(teste.statusCode);
expect(resposta.body).to.have.property('error', teste.mensagemEsperada);
```

✓

Testando a regra relacionada a Remetente não encontrado

5ms

🕒

mts20t2-d3-g8

🕒 298ms

📄 7

📄 13



✓ Testando a regra relacionada a Destinatário não encontrado

5ms 🕒

## GET /transfers

/test/rest/external/transferExternal.test.js

🕒 37ms 📄 2 ✓ 2

✓ Quando consulto a lista de transferências autenticado, recebo um array

33ms 🕒

```
const postLogin = require('../fixture/requisicoes/login/postLogin.json');
const respostaLogin = await request(process.env.BASE_URL_REST)
  .post('/users/login')
  .send(postLogin);
const token = respostaLogin.body.token;
const resposta = await request(process.env.BASE_URL_REST)
  .get('/transfers')
  .set('Authorization', `Bearer ${token}`);
expect(resposta.status).toEqual(200);
expect(resposta.body).toBe.an('array');
```

✓ Quando consulto a lista de transferências sem token, recebo erro (401 UNAUTHORIZED)

4ms 🕒

```
const resposta = await request(process.env.BASE_URL_REST)
  .get('/transfers');
expect(resposta.status).toEqual(401);
expect(resposta.body).to.have.property('message', 'Token não fornecido.');
```

## User

/test/rest/external/userExternal.test.js

## POST /users/register

/test/rest/external/userExternal.test.js

🕒 27ms 📄 2 ✓ 2

✓ Quando informo dados válidos, registro o usuário com sucesso (201 CREATED)

24ms 🕒

```
const novoUsuario = {
  username: `usuario_teste_${Date.now()}`,
  password: 'senha123',
  favoritos: ['julio']
};
const resposta = await request(process.env.BASE_URL_REST)
  .post('/users/register')
```

mts20t2-d3-g8

🕒 298ms

📄 7

📄 13



```
const usuarioExistente = {
  username: 'julio',
  password: 'senha123',
  favoritos: ['priscila']
};
const resposta = await request(process.env.BASE_URL_REST)
  .post('/users/register')
  .send(usuarioExistente);
expect(resposta.status).to.equal(400);
expect(resposta.body).to.have.property('error');
```

## POST /users/login

/test/rest/external/userExternal.test.js

🕒 51ms 📄 3 ✓ 3

✓ Quando informo credenciais válidas, faço login com sucesso (200 OK)

25ms 🕒

```
const resposta = await request(process.env.BASE_URL_REST)
  .post('/users/login')
  .send({ username: 'julio', password: '123456' });
expect(resposta.status).to.equal(200);
expect(resposta.body).to.have.property('token');
```

✓ Quando informo senha inválida, recebo erro (400 BAD REQUEST)

23ms 🕒

```
const resposta = await request(process.env.BASE_URL_REST)
  .post('/users/login')
  .send({ username: 'julio', password: 'senha_errada' });
expect(resposta.status).to.equal(400);
expect(resposta.body).to.have.property('error');
```

✓ Quando informo usuário inexistente, recebo erro (400 BAD REQUEST)

3ms 🕒

```
const resposta = await request(process.env.BASE_URL_REST)
  .post('/users/login')
  .send({ username: 'usuario_inexistente', password: 'qualquer_senha' });
expect(resposta.status).to.equal(400);
expect(resposta.body).to.have.property('error');
```

## GET /users

/test/rest/external/userExternal.test.js

🕒 2ms 📄 1 ✓ 1

mts20t2-d3-g8

🕒 298ms

📄 7

📄 13

