US002 - Login

Sendo um vendedor de uma loja com cadastro já realizado
Gostaria de poder me autenticar no Marketplace da ServeRest
Para poder cadastrar, editar, atualizar e excluir meus produtos

DoR

- Banco de dados e infraestrutura para desenvolvimento disponibilizados;
- API de cadastro de usuários implementada;
- Ambiente de testes disponibilizado.

DoD

- Autenticação com geração de token Bearer implementada;
- Análise de testes cobrindo a rota de login;
- Matriz de rastreabilidade atualizada;
- Automação de testes baseado na análise realizada;

Acceptance Criteria

- Usuários não cadastrados não deverão conseguir autenticar;
- Usuários com senha inválida não deverão conseguir autenticar;
- No caso de não autenticação, deverá ser retornado um status code `401` (Unauthorized);
- Usuários existentes e com a senha correta deverão ser autenticados;
- A autenticação deverá gerar um token Bearer;
- A duração da validade do token deverá ser de 10 minutos;
- Os testes executados deverão conter evidências;
- A cobertura de testes deve se basear no Swagger e ir além, cobrindo cenários alternativos.

Cenários de Teste

CT-015 – Login com credenciais válidas

Pré-condição: Usuário válido cadastrado (ex: `joao@valido.com`, senha `123456`

Passos:

1. Realizar uma requisição POST para o endpoint `/login`
2. Enviar o seguinte payload:

```
{
  "email": "joao@valido.com",
  "password": "123456"
}
```

Resultado Esperado:

- Status code `200`

- Mensagem: `"Login realizado com sucesso"`
- Token do tipo Bearer é retornado

CT-016 – Login com usuário não cadastrado

Pré-condição: E-mail não existe na base

Passos:

1. Realizar uma requisição POST para `/login` com um e-mail inexistente
2. Enviar senha válida qualquer

Resultado Esperado:

- Status code `401`
- Mensagem: `"Email e/ou senha inválidos"`

CT-017 – Login com senha inválida

Pré-condição: E-mail existente com senha correta na base

Passos:

1. Realizar POST para `/login` com senha incorreta

Resultado Esperado:

- Status code `401`
- Mensagem: `"Email e/ou senha inválidos"`

CT-018 – Login com e-mail mal formatado

Passos:

- Realizar POST para `/login` com e-mail `"joaogmail.com"`
- Enviar qualquer senha válida

Resultado Esperado:

- Status code `400` ou validação de erro (a depender do tratamento da API)

CT-019 – Login sem fornecer senha

Passos:

1. Realizar POST para `/login` omitindo o campo `"password"`

Resultado Esperado:

- Status code 400
- Mensagem de erro indicando campo obrigatório ausente

CT-020 – Login sem fornecer e-mail

Passos:

1. Realizar POST para `/login` omitindo o campo `"email"`

Resultado Esperado:

- Status code `400`
- Mensagem de erro indicando campo obrigatório ausente

CT-021 – Login com campo extra no payload

Passos:

1. Realizar POST para `/login` com payload contendo um campo adicional, ex `admin`:

```
{
  "email": "joao@valido.com",
  "password": "123456",
  "admin": true
}
```

Resultado Esperado:

- Status code `200` (se a API ignora campos extras)
- Ou status `400` se a API bloquear o payload inválido

CT-022 – Validade do token (10 minutos

Pré-condição: Usuário autenticado com token gerado

Passos:

1. Fazer login e guardar token
2. Esperar 10 minutos
3. Realizar GET em endpoint protegido (ex: `/produtos`) com esse token

Resultado Esperado:

- Após 10 minutos, status code `401 Unauthorized`
- Mensagem indicando expiração do token

Pós-condição: Sessão expirada

CT-023 – Reutilização de token após logout ou expiração

Pré-condição: Usuário autenticado, token expirado ou invalido após logout

Passos:

1. Tentar acessar `/produtos` com token expirado ou inválido

Resultado Esperado:

- Status code `401 Unauthorized`

CT-024 – Login com e-mail em letras maiúsculas

Pré-condição: Usuário cadastrado com letras minúsculas no e-mail

Passos:

1. Realizar POST `/login` com e-mail `"JOAO@VALIDO.COM"` (variação em caixa alta)

Resultado Esperado:

- Status code `401` (se case-sensitive)
- Ou 200 OK (se API não diferencia)

> **Observação:** Avalia sensibilidade do campo de e-mail

Priorização dos Testes

ID	Endpoint	Cenário	Tipo	Prioridade	Status
CT-015	`/login`	Login com e-mail e senha corretos	Funcional / Automatizado	Alta	EXECUTADO
CT-016	`/login`	Login com usuário inexistente	Funcional / Automatizado	Alta	NÃO EXECUTADO
CT-017	`/login`	Login com senha incorreta	Funcional / Automatizado	Alta	EXECUTADO
CT-018	`/login`	Login com e-mail mal formatado	Particionamento / Funcional	Média	EXECUTADO
CT-019	`/login`	Login sem campo `email`	Funcional / Negativo	Alta	NÃO EXECUTADO
CT-020	`/login`	Login sem campo `password`	Funcional / Negativo	Alta	NÃO EXECUTADO
CT-022	`/login`	Token expira após 10 minutos	E2E / Segurança	Crítica	NÃO EXECUTADO