

# Tratamento de erros

← Índice da documentação da API

Fontes: `controller/exception/GlobalExceptionHandler.java`, `mes/MesErrors.java`, `mes/MesEdgeSecurity.java`, `config/LoginRateLimitingFilter.java`, `config/PreflightAwareAuthenticationEntryPoint.java`, `util/MensagemErro.java`, e, no frontend, `src/axios.ts` e `src/utils/userFacingError.ts`.

## 1. Formato padrão de erro ( `ErrorResponseDTO` )

A maior parte dos erros sai pelo `GlobalExceptionHandler` neste formato:

```
{
  "timestamp": "2026-10-15T09:41:07.123456",
  "status": 400,
  "error": "Bad Request",
  "message": "Descrição legível do problema.",
  "path": "/api/produtos"
}
```

json

| CAMPO                  | TIPO                                | OBSERVAÇÃO                                                                                                                  |
|------------------------|-------------------------------------|-----------------------------------------------------------------------------------------------------------------------------|
| <code>timestamp</code> | <code>LocalDateTime</code> ISO-8601 | hora do servidor, <b>sem fuso</b>                                                                                           |
| <code>status</code>    | inteiro                             | mesmo valor do status HTTP                                                                                                  |
| <code>error</code>     | texto                               | descrição padrão do status HTTP em inglês ( <code>Bad Request</code> , <code>Conflict</code> ...)                           |
| <code>message</code>   | texto                               | <b>em português</b> , é o texto mostrado ao usuário pelo frontend; se vier vazio vira <code>Erro interno inesperado.</code> |
| <code>path</code>      | texto                               | URI da requisição                                                                                                           |

## 2. Mapeamento exceção → status (cadeia global)

| SITUAÇÃO / EXCEÇÃO                                                      | STATUS | MESSAGE                                                                                                                                                                                                                                                                                       |
|-------------------------------------------------------------------------|--------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Bean Validation falhou ( @Valid )                                       | 400    | mensagens dos campos separadas por ; (ou Dados inválidos. )                                                                                                                                                                                                                                   |
| IllegalArgumentException                                                | 400    | a mensagem da exceção (regra de negócio violada)                                                                                                                                                                                                                                              |
| DataIntegrityViolationException                                         | 400    | Já existe um produto com este código Wester ... (violação de codigo_sistema_wester ); Registro duplicado detectado. Verifique os campos únicos. ; Não foi possível concluir a operação por vínculo com outros registros. ; ou Violação de integridade de dados. Verifique os campos enviados. |
| InvalidCredentialsException                                             | 401    | Usuário ou senha inválidos. / Informe login e senha para continuar.                                                                                                                                                                                                                           |
| AuthenticationCredentialsNotFoundException                              | 401    | mensagem da exceção                                                                                                                                                                                                                                                                           |
| InactiveUserException                                                   | 403    | Usuário inativo. Procure um administrador.                                                                                                                                                                                                                                                    |
| AuthorizationDeniedException (falha de @PreAuthorize )                  | 403    | Você não tem permissão para executar esta ação.                                                                                                                                                                                                                                               |
| RuntimeException cuja mensagem contém "não encontrado(o/a)"             | 404    | a mensagem (comparação sem acento e sem gênero, via MensagemErro )                                                                                                                                                                                                                            |
| HttpRequestMethodNotSupportedException                                  | 405    | mensagem do Spring                                                                                                                                                                                                                                                                            |
| IllegalStateException                                                   | 409    | a mensagem (conflito de estado)                                                                                                                                                                                                                                                               |
| Conflito de versão otimista ( ObjectOptimisticLockingFailureException , | 409    | OPTIMISTIC_LOCK: 0 registro foi alterado por outro usuário. Atualize os dados e tente novamente.                                                                                                                                                                                              |

| SITUAÇÃO / EXCEÇÃO                                                                         | STATUS              | MESSAGE                                                 |
|--------------------------------------------------------------------------------------------|---------------------|---------------------------------------------------------|
| <code>OptimisticLockException</code><br><code>n )</code>                                   |                     |                                                         |
| <code>ResponseStatusException</code><br><code>n</code>                                     | o status da exceção | <code>reason</code> da exceção                          |
| Qualquer outra<br><code>RuntimeException</code> /<br><code>Exception</code>                | <code>500</code>    | mensagem da exceção; erro registrado em log             |
| Cliente desconectou no meio da resposta<br>( <code>AsyncRequestNotUsableException</code> ) | —                   | apenas log em nível <code>debug</code> ; não gera corpo |

Pontos a saber:

- O prefixo `OPTIMISTIC_LOCK:` na mensagem é um contrato com o frontend, que abre o `ConflictDialog` ("alterado por outro usuário"). Entidades com controle de versão ( `@Version` ) no código: `Fileira` , `Grade` , `Nivel` e `ItemEstoque` (campo `version` ).
- **404 depende do texto da mensagem.** Uma exceção genérica só vira `404` se a mensagem contiver "não encontrad..."; se o texto mudar, o status passa a `500` . Está documentado no próprio utilitário `MensagemErro` .
- O texto de `IllegalArgumentException` e `IllegalStateException` vai direto ao cliente. Por isso as mensagens dos serviços são escritas em português para o usuário final.

### 3. Formatos de erro diferentes do padrão

Nem todo erro usa o `ErrorResponseDTO` . Integradores devem tratar estes formatos:

| ORIGEM                                                                                            | STATUS                                                                               | CORPO                                                                                                                                                                 |
|---------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <code>UsuarioController</code> (criar, atualizar, alterar senha, status, excluir)                 | <code>400</code> , <code>403</code> , <code>404</code> , <code>500</code>            | texto puro ( <code>String</code> ) no corpo, por exemplo <code>Senha é obrigatória.</code> e <code>Você não tem permissão para alterar a senha de outra conta.</code> |
| MES — <code>MesErrors</code> (apenas rotas de <code>MesController</code> )                        | <code>400</code> , status de <code>ResponseStatusException</code> , <code>403</code> | <code>{"message": " ... "}</code> ( <code>Dados invalidos</code> , <code>Operacao recusada</code> , <code>Sem permissao</code> quando não há texto)                   |
| Edge — <code>MesEdgeSecurity</code>                                                               | <code>401</code> ou <code>503</code>                                                 | <code>{"message":"Integracao edge indisponivel ou credencial invalida"}</code>                                                                                        |
| Limite de taxa — <code>LoginRateLimitingFilter</code>                                             | <code>429</code>                                                                     | <code>{"status":429,"error":"Too Many Requests","message":"Muitas tentativas. Aguarde um momento antes de tentar novamente."}</code>                                  |
| Falta de autenticação na cadeia principal ( <code>PreflightAwareAuthenticationEntryPoint</code> ) | <code>401</code>                                                                     | sem corpo (apenas o status); um pré-voo CORS recebe <code>200</code>                                                                                                  |

`PreflightAwareAuthenticationEntryPoint` define apenas o status: `401` sem corpo quando falta autenticação em uma rota protegida (e `200` para pré-voo CORS). Um cliente **não deve** esperar `ErrorResponseDTO` nesse caso.

Os controllers do MES que respondem `ResponseStatusException` seguem `MesErrors`, não o handler global, porque o `@RestControllerAdvice` do MES tem `@Order(0)` e escopo `basePackageClasses=MesController.class`.

## 4. Códigos HTTP que a API pode devolver

| STATUS | QUANDO                                                                                                   |
|--------|----------------------------------------------------------------------------------------------------------|
| 200    | consulta ou alteração com corpo                                                                          |
| 201    | criação ( <code>POST</code> <code>/api/usuarios</code> e outras criações que declaram <code>201</code> ) |
| 204    | alteração sem corpo (por exemplo, troca de senha)                                                        |
| 400    | validação, regra de negócio, integridade de dados, JSON inválido                                         |
| 401    | token ausente/inválido/expirado, usuário ou senha inválidos, chave do Edge incorreta                     |
| 403    | usuário inativo, falta de permissão ( <code>SCREEN_*</code> / <code>FULL_ACCESS</code> /MES)             |
| 404    | recurso não encontrado                                                                                   |
| 405    | método HTTP não suportado na rota                                                                        |
| 409    | conflito de estado ou de versão (otimista); MES também usa para sequência de eventos do coletor          |
| 429    | excesso de tentativas de login/cadastro por IP                                                           |
| 500    | erro não tratado                                                                                         |
| 503    | integração Edge indisponível (chave não configurada)                                                     |

Cada ficha de endpoint ( `modulos/*.md` ) lista os códigos declarados e as mensagens de validação identificadas no serviço. Ela **não** garante que todas as mensagens possíveis estejam listadas: as mensagens dinâmicas e as que vêm de métodos auxiliares aparecem em documento interno de regras de negócio.

## 5. Como o frontend trata os erros (para o manual e para integradores)

- 401 com token anexado:** o `axios` dispara o *handler* de não autorizado, que faz logout e volta para a tela de Login (o token vale 8 horas).
- Mensagem exibida:** `getUserFacingErrorMessage` usa o `message` da API, **descartando** textos técnicos ( `network error` , `internal server error` , `not found` , `timeout` , `404` , `500` , HTML,

`stack trace` ...) e caindo na mensagem padrão da tela. Para `401` / `403` sem texto aproveitável, mostra `Você não tem permissão para concluir esta operação.`

- **Conflito** `409` : mostra a mensagem da API ou `Os dados foram alterados por outro usuário. Atualize a tela e tente novamente.`
- **Login:** `401` → `E-mail ou senha inválidos.` ; `403` → `Seu usuário não pode acessar o sistema.` ; sem resposta do servidor → `Não foi possível conectar ao servidor. Tente novamente.`

---

## 6. Recomendações para quem integra

1. Trate o corpo de erro como **texto ou JSON** dependendo da rota (seção 3); verifique o `Content-Type` .
2. Use o `status` HTTP como contrato; o `message` é para pessoas e pode mudar.
3. Em `409` com prefixo `OPTIMISTIC_LOCK:` , releia o recurso (incluindo `version` ) antes de tentar de novo.
4. Em `429` , espere um minuto antes de repetir.
5. Para o Edge, `503` significa que a chave **não está configurada no servidor** (não é erro do coletor), e `401` que a chave enviada não confere.