

Setores e tipos de material

Índice de endpoints · Modelos

API-002 — Listar Todas Areas

GET `/api/areas`

GET `/api/setores`

Finalidade: Listar Todas Areas.

Fonte: `src/main/java/com/wester/storage/controller/AreaEstoqueController.java:46` — `AreaEstoqueController.listarTodasAreas`.

Autenticação, autorização e perfis: `hasAuthority('SCREEN_WAREHOUSE')` Consulte a correspondência entre autoridades e perfis em Autenticação.

Headers: `Authorization: Bearer <TOKEN>`. `Accept: application/json` recomendado.

Parâmetros:

Sem query parameters, path variables ou headers de negócio declarados.

Corpo: não declarado; não enviar corpo.

Resposta de sucesso: `List<AreaEstoque>`. Campos em Modelos.

Exemplo estrutural sintético, sem afirmar valores de produção:

```
[ ]
```

json

Códigos HTTP: `200`, `401`, `403`. Códigos de segurança dependem da cadeia de filtros; falhas não capturadas podem chegar ao tratamento global descrito em Erros.

Regras de negócio e dependências: `AreaEstoqueService.listarTodasAreas`. Consulte documento interno de regras de negócio para efeitos persistentes e validações compartilhadas.

cURL (Bash):

```
curl -X GET "${BASE_URL}/api/areas" -H "Authorization: Bearer ${TOKEN}"
```

bash

API-003 — Buscar Area Por Id

GET `/api/areas/{id}`

GET `/api/setores/{id}`

Finalidade: Buscar Area Por Id.

Fonte: `src/main/java/com/wester/storage/controller/AreaEstoqueController.java:53` — `AreaEstoqueController.buscarAreaPorId`.

Autenticação, autorização e perfis: `hasAuthority('SCREEN_WAREHOUSE')` Consulte a correspondência entre autoridades e perfis em Autenticação.

Headers: `Authorization: Bearer <TOKEN>`. `Accept: application/json` recomendado.

Parâmetros:

NOME	LOCAL	TIPO	OBRIGATÓRIO	PADRÃO
<code>id</code>	path	<code>Long</code>	Sim	—

Corpo: não declarado; não enviar corpo.

Resposta de sucesso: `AreaEstoque`. Campos em Modelos.

Exemplo estrutural sintético, sem afirmar valores de produção:

```
{
  "id": 1,
  "descricao": "Exemplo",
  "nome": "Exemplo",
  "ativo": false,
  "tipoEstrutura": "DINAMICO",
  "labelEstruturaPrimaria": "Exemplo",
  "labelEstruturaSecundaria": "Exemplo",
  "labelNivel": "Exemplo",
  "labelSlot": "Exemplo",
  "labelFileira": "Exemplo",
  "labelGrade": "Exemplo",
  "labelPosicao": "Exemplo"
}
```

json

Códigos HTTP: 200, 401, 403, 404. Códigos de segurança dependem da cadeia de filtros; falhas não capturadas podem chegar ao tratamento global descrito em Erros.

Regras de negócio e dependências: `AreaEstoqueService.buscarAreaPorId`. Consulte documento interno de regras de negócio para efeitos persistentes e validações compartilhadas.

cURL (Bash):

```
curl -X GET "${BASE_URL}/api/areas/1" -H "Authorization: Bearer ${TOKEN}"
```

bash

API-004 — Buscar Por Nome

GET `/api/areas/nome/{nome}`

GET `/api/setores/nome/{nome}`

Finalidade: Buscar Por Nome.

Fonte: `src/main/java/com/wester/storage/controller/AreaEstoqueController.java:62` — `AreaEstoqueController.buscarPorNome`.

Autenticação, autorização e perfis: `hasAuthority('SCREEN_WAREHOUSE')` Consulte a correspondência entre autoridades e perfis em Autenticação.

Headers: `Authorization: Bearer <TOKEN>`. `Accept: application/json` recomendado.

Parâmetros:

NOME	LOCAL	TIPO	OBRIGATÓRIO	PADRÃO
<code>nome</code>	path	<code>String</code>	Sim	—

Corpo: não declarado; não enviar corpo.

Resposta de sucesso: `AreaEstoque`. Campos em Modelos.

Exemplo estrutural sintético, sem afirmar valores de produção:

json

```
{
  "id": 1,
  "descricao": "Exemplo",
  "nome": "Exemplo",
  "ativo": false,
  "tipoEstrutura": "DINAMICO",
  "labelEstruturaPrimaria": "Exemplo",
  "labelEstruturaSecundaria": "Exemplo",
  "labelNivel": "Exemplo",
  "labelSlot": "Exemplo",
  "labelFileira": "Exemplo",
  "labelGrade": "Exemplo",
  "labelPosicao": "Exemplo"
}
```

Códigos HTTP: 200, 401, 403, 404. Códigos de segurança dependem da cadeia de filtros; falhas não capturadas podem chegar ao tratamento global descrito em Erros.

Regras de negócio e dependências: `AreaEstoqueService.buscarPorNome`. Consulte documento interno de regras de negócio para efeitos persistentes e validações compartilhadas.

cURL (Bash):

bash

```
curl -X GET "${BASE_URL}/api/areas/nome/1" -H "Authorization: Bearer ${TOKEN}"
```

API-005 — Criar Area

POST `/api/areas`

POST `/api/setores`

Finalidade: Criar Area.

Fonte: `src/main/java/com/wester/storage/controller/AreaEstoqueController.java:72` — `AreaEstoqueController.criarArea`.

Autenticação, autorização e perfis: `hasAuthority('SCREEN_WAREHOUSE')` and `hasRole('FULL_ACCESS')` Consulte a correspondência entre autoridades e perfis em Autenticação.

Headers: `Authorization: Bearer <TOKEN>`. `Accept: application/json` recomendado. `Content-Type: application/json` para o corpo.

Parâmetros:

Sem query parameters, path variables ou headers de negócio declarados.

Corpo da requisição: `AreaEstoque`; obrigatório. Validação Jakarta ativada por `@Valid`. Consulte os campos e restrições em Modelos.

Exemplo sintético (IDs devem existir e versões devem vir de consulta):

```
{
  "nome": "Setor de exemplo",
  "descricao": "Área de exemplo",
  "ativo": true,
  "tipoEstrutura": "DINAMICO"
}
```

json

Resposta de sucesso: `AreaEstoque`. Campos em Modelos.

Exemplo estrutural sintético, sem afirmar valores de produção:

```
{
  "id": 1,
  "descricao": "Exemplo",
  "nome": "Exemplo",
  "ativo": false,
  "tipoEstrutura": "DINAMICO",
  "labelEstruturaPrimaria": "Exemplo",
  "labelEstruturaSecundaria": "Exemplo",
  "labelNivel": "Exemplo",
  "labelSlot": "Exemplo",
  "labelFileira": "Exemplo",
  "labelGrade": "Exemplo",
  "labelPosicao": "Exemplo"
}
```

json

Códigos HTTP: `201`, `400`, `401`, `403`, `500`. Códigos de segurança dependem da cadeia de filtros; falhas não capturadas podem chegar ao tratamento global descrito em Erros.

Validações e erros identificados na cadeia local do serviço:

- Nome da área de estoque já existe: [mensagem pode receber complemento dinâmico]

Regras de negócio e dependências: `AreaEstoqueService.salvarArea`. Consulte documento interno de regras de negócio para efeitos persistentes e validações compartilhadas.

cURL (Bash):

```
curl -X POST "${BASE_URL}/api/areas" -H "Authorization: Bearer ${TOKEN}" -H 'Content-Type: application/json'
```

API-006 — Atualizar Area

PUT `/api/areas/{id}`

PUT `/api/setores/{id}`

Finalidade: Atualizar Area.

Fonte: `src/main/java/com/wester/storage/controller/AreaEstoqueController.java:86` — `AreaEstoqueController.atualizarArea`.

Autenticação, autorização e perfis: `hasAuthority('SCREEN_WAREHOUSE')` and `hasRole('FULL_ACCESS')` Consulte a correspondência entre autoridades e perfis em Autenticação.

Headers: `Authorization: Bearer <TOKEN>`. `Accept: application/json` recomendado. `Content-Type: application/json` para o corpo.

Parâmetros:

NOME	LOCAL	TIPO	OBRIGATÓRIO	PADRÃO
<code>id</code>	path	<code>Long</code>	Sim	—

Corpo da requisição: `AreaEstoque`; obrigatório. Validação Jakarta ativada por `@Valid`. Consulte os campos e restrições em Modelos.

Exemplo sintético (IDs devem existir e versões devem vir de consulta):

```
{
  "nome": "Setor de exemplo",
  "descricao": "Área de exemplo",
  "ativo": true,
  "tipoEstrutura": "DINAMICO"
}
```

json

Resposta de sucesso: `AreaEstoque` . Campos em Modelos.

Exemplo estrutural sintético, sem afirmar valores de produção:

```
{
  "id": 1,
  "descricao": "Exemplo",
  "nome": "Exemplo",
  "ativo": false,
  "tipoEstrutura": "DINAMICO",
  "labelEstruturaPrimaria": "Exemplo",
  "labelEstruturaSecundaria": "Exemplo",
  "labelNivel": "Exemplo",
  "labelSlot": "Exemplo",
  "labelFileira": "Exemplo",
  "labelGrade": "Exemplo",
  "labelPosicao": "Exemplo"
}
```

json

Códigos HTTP: `200` , `400` , `401` , `403` , `404` , `500` . Códigos de segurança dependem da cadeia de filtros; falhas não capturadas podem chegar ao tratamento global descrito em Erros.

Validações e erros identificados na cadeia local do serviço:

- Nome da área de estoque já existe: [mensagem pode receber complemento dinâmico]

Regras de negócio e dependências: `AreaEstoqueService.atualizarArea` . Consulte documento interno de regras de negócio para efeitos persistentes e validações compartilhadas.

cURL (Bash):

```
curl -X PUT "${BASE_URL}/api/areas/1" -H "Authorization: Bearer ${TOKEN}" -H 'Content-Type: application/json'
```

bash

API-007 — Atualizar Labels

PATCH `/api/areas/{id}/labels`

PATCH `/api/setores/{id}/labels`

Finalidade: Atualizar Labels.

Fonte: `src/main/java/com/wester/storage/controller/AreaEstoqueController.java:107` — `AreaEstoqueController.atualizarLabels`.

Autenticação, autorização e perfis: `hasAuthority('SCREEN_WAREHOUSE')` and `hasRole('FULL_ACCESS')` Consulte a correspondência entre autoridades e perfis em Autenticação.

Headers: `Authorization: Bearer <TOKEN>`. `Accept: application/json` recomendado. `Content-Type: application/json` para o corpo.

Parâmetros:

NOME	LOCAL	TIPO	OBRIGATÓRIO	PADRÃO
<code>id</code>	path	<code>Long</code>	Sim	—

Corpo da requisição: `AreaLabelsUpdateRequestDTO`; obrigatório. Validação Jakarta ativada por `@Valid`. Consulte os campos e restrições em Modelos.

Exemplo sintético (IDs devem existir e versões devem vir de consulta):

```
{
  "labelEstruturaPrimaria": "Exemplo",
  "labelEstruturaSecundaria": "Exemplo",
  "labelNivel": "Exemplo",
  "labelSlot": "Exemplo"
}
```

json

Resposta de sucesso: `AreaEstoque`. Campos em Modelos.

Exemplo estrutural sintético, sem afirmar valores de produção:

json

```
{
  "id": 1,
  "descricao": "Exemplo",
  "nome": "Exemplo",
  "ativo": false,
  "tipoEstrutura": "DINAMICO",
  "labelEstruturaPrimaria": "Exemplo",
  "labelEstruturaSecundaria": "Exemplo",
  "labelNivel": "Exemplo",
  "labelSlot": "Exemplo",
  "labelFileira": "Exemplo",
  "labelGrade": "Exemplo",
  "labelPosicao": "Exemplo"
}
```

Códigos HTTP: 200, 400, 401, 403, 404, 409, 500. Códigos de segurança dependem da cadeia de filtros; falhas não capturadas podem chegar ao tratamento global descrito em Erros.

Validações e erros identificados na cadeia local do serviço:

- Labels configuráveis estão disponíveis para setores FIXO.

Regras de negócio e dependências: `AreaEstoqueService.atualizarLabels`. Consulte documento interno de regras de negócio para efeitos persistentes e validações compartilhadas.

cURL (Bash):

bash

```
curl -X PATCH "${BASE_URL}/api/areas/1/labels" -H "Authorization: Bearer ${TOKEN}" -H 'Cont
```

API-008 — Ativar Area

PATCH /api/areas/{id}/activate ; **POST** /api/areas/{id}/activate

PATCH /api/areas/{id}/ativar ; **POST** /api/areas/{id}/ativar

PATCH /api/setores/{id}/activate ; **POST** /api/setores/{id}/activate

PATCH /api/setores/{id}/ativar ; **POST** /api/setores/{id}/ativar

Finalidade: Ativar Area.

Fonte: `src/main/java/com/wester/storage/controller/AreaEstoqueController.java:129` — `AreaEstoqueController.ativarArea`.

Autenticação, autorização e perfis: `hasAuthority('SCREEN_WAREHOUSE')` and `hasRole('FULL_ACCESS')` Consulte a correspondência entre autoridades e perfis em Autenticação.

Headers: `Authorization: Bearer <TOKEN>`. `Accept: application/json` recomendado.

Parâmetros:

NOME	LOCAL	TIPO	OBRIGATÓRIO	PADRÃO
<code>id</code>	path	<code>Long</code>	Sim	—

Corpo: não declarado; não enviar corpo.

Resposta de sucesso: `AreaEstoque`. Campos em Modelos.

Exemplo estrutural sintético, sem afirmar valores de produção:

```
{
  "id": 1,
  "descricao": "Exemplo",
  "nome": "Exemplo",
  "ativo": false,
  "tipoEstrutura": "DINAMICO",
  "labelEstruturaPrimaria": "Exemplo",
  "labelEstruturaSecundaria": "Exemplo",
  "labelNivel": "Exemplo",
  "labelSlot": "Exemplo",
  "labelFileira": "Exemplo",
  "labelGrade": "Exemplo",
  "labelPosicao": "Exemplo"
}
```

json

Códigos HTTP: `200`, `401`, `403`, `404`, `500`. Códigos de segurança dependem da cadeia de filtros; falhas não capturadas podem chegar ao tratamento global descrito em Erros.

Regras de negócio e dependências: `AreaEstoqueService.ativarArea`. Consulte documento interno de regras de negócio para efeitos persistentes e validações compartilhadas.

cURL (Bash):

```
curl -X PATCH "${BASE_URL}/api/areas/1/activate" -H "Authorization: Bearer ${TOKEN}"
```

API-009 — Inativar Area

PATCH /api/areas/{id}/inactivate ; **POST** /api/areas/{id}/inactivate

PATCH /api/areas/{id}/inativar ; **POST** /api/areas/{id}/inativar

PATCH /api/setores/{id}/inactivate ; **POST** /api/setores/{id}/inactivate

PATCH /api/setores/{id}/inativar ; **POST** /api/setores/{id}/inativar

Finalidade: Inativar Area.

Fonte: `src/main/java/com/wester/storage/controller/AreaEstoqueController.java:145` —
`AreaEstoqueController.inativarArea`.

Autenticação, autorização e perfis: `hasAuthority('SCREEN_WAREHOUSE')` and `hasRole('FULL_ACCESS')` Consulte a correspondência entre autoridades e perfis em Autenticação.

Headers: `Authorization: Bearer <TOKEN>`. `Accept: application/json` recomendado.

Parâmetros:

NOME	LOCAL	TIPO	OBRIGATÓRIO	PADRÃO
<code>id</code>	path	<code>Long</code>	Sim	—

Corpo: não declarado; não enviar corpo.

Resposta de sucesso: `AreaEstoque`. Campos em Modelos.

Exemplo estrutural sintético, sem afirmar valores de produção:

json

```
{
  "id": 1,
  "descricao": "Exemplo",
  "nome": "Exemplo",
  "ativo": false,
  "tipoEstrutura": "DINAMICO",
  "labelEstruturaPrimaria": "Exemplo",
  "labelEstruturaSecundaria": "Exemplo",
  "labelNivel": "Exemplo",
  "labelSlot": "Exemplo",
  "labelFileira": "Exemplo",
  "labelGrade": "Exemplo",
  "labelPosicao": "Exemplo"
}
```

Códigos HTTP: 200, 401, 403, 404, 500. Códigos de segurança dependem da cadeia de filtros; falhas não capturadas podem chegar ao tratamento global descrito em Erros.

Regras de negócio e dependências: `AreaEstoqueService.inativarArea`. Consulte documento interno de regras de negócio para efeitos persistentes e validações compartilhadas.

cURL (Bash):

bash

```
curl -X PATCH "${BASE_URL}/api/areas/1/inactivate" -H "Authorization: Bearer ${TOKEN}"
```

API-010 — Deletar Area

DELETE `/api/areas/{id}`

DELETE `/api/setores/{id}`

Finalidade: Deletar Area.

Fonte: `src/main/java/com/wester/storage/controller/AreaEstoqueController.java:162` — `AreaEstoqueController.deletarArea`.

Autenticação, autorização e perfis: `hasAuthority('SCREEN_WAREHOUSE')` and `hasRole('FULL_ACCESS')` Consulte a correspondência entre autoridades e perfis em Autenticação.

Headers: `Authorization: Bearer <TOKEN>`. `Accept: application/json` recomendado.

Parâmetros:

NOME	LOCAL	TIPO	OBRIGATÓRIO	PADRÃO
id	path	Long	Sim	—

Corpo: não declarado; não enviar corpo.

Resposta de sucesso: sem corpo quando 204.

Códigos HTTP: 204, 401, 403, 404, 409, 500. Códigos de segurança dependem da cadeia de filtros; falhas não capturadas podem chegar ao tratamento global descrito em Erros.

Validações e erros identificados na cadeia local do serviço:

- Não é possível deletar a área de estoque pois ela contém fileiras associadas.

Regras de negócio e dependências: `AreaEstoqueService.deletarArea`. Consulte documento interno de regras de negócio para efeitos persistentes e validações compartilhadas.

cURL (Bash):

```
curl -X DELETE "${BASE_URL}/api/areas/1" -H "Authorization: Bearer ${TOKEN}"
```

bash

API-011 — Listar

GET `/api/area-material-tags/area/{areaId}`

Finalidade: Listar.

Fonte: `src/main/java/com/wester/storage/controller/AreaMaterialTagController.java:26` — `AreaMaterialTagController.listar`.

Autenticação, autorização e perfis: `hasAuthority('SCREEN_WAREHOUSE')` Consulte a correspondência entre autoridades e perfis em Autenticação.

Headers: `Authorization: Bearer <TOKEN>`. `Accept: application/json` recomendado.

Parâmetros:

NOME	LOCAL	TIPO	OBRIGATÓRIO	PADRÃO
areaId	path	Long	Sim	—

Corpo: não declarado; não enviar corpo.

Resposta de sucesso: `List<MaterialType>`. Campos em Modelos.

Exemplo estrutural sintético, sem afirmar valores de produção:

```
[ ]
```

json

Códigos HTTP: `200`, `401`, `403`. Códigos de segurança dependem da cadeia de filtros; falhas não capturadas podem chegar ao tratamento global descrito em Erros.

Regras de negócio e dependências: `AreaMaterialTagService.listarTags`. Consulte documento interno de regras de negócio para efeitos persistentes e validações compartilhadas.

cURL (Bash):

```
curl -X GET "${BASE_URL}/api/area-material-tags/area/1" -H "Authorization: Bearer ${TOKEN}"
```

bash

API-012 — Configurar

PUT `/api/area-material-tags/area/{areaId}`

Finalidade: Configurar.

Fonte: `src/main/java/com/wester/storage/controller/AreaMaterialTagController.java:33` — `AreaMaterialTagController.configurar`.

Autenticação, autorização e perfis: `hasAuthority('SCREEN_WAREHOUSE')` and `hasRole('FULL_ACCESS')` Consulte a correspondência entre autoridades e perfis em Autenticação.

Headers: `Authorization: Bearer <TOKEN>`. `Accept: application/json` recomendado. `Content-Type: application/json` para o corpo.

Parâmetros:

NOME	LOCAL	TIPO	OBRIGATÓRIO	PADRÃO
areaId	path	Long	Sim	—

Corpo da requisição: `AreaMaterialTagsRequestDTO` ; obrigatório. Validação Jakarta ativada por `@Valid` . Consulte os campos e restrições em Modelos.

Exemplo sintético (IDs devem existir e versões devem vir de consulta):

```
{
  "materialTypes": [
    "MOLDE",
    "PRODUTO"
  ]
}
```

json

Resposta de sucesso: `List<MaterialType>` . Campos em Modelos.

Exemplo estrutural sintético, sem afirmar valores de produção:

```
[]
```

json

Códigos HTTP: `200` , `400` , `401` , `403` . Códigos de segurança dependem da cadeia de filtros; falhas não capturadas podem chegar ao tratamento global descrito em Erros.

Regras de negócio e dependências: `AreaMaterialTagService.configurarTags` . Consulte documento interno de regras de negócio para efeitos persistentes e validações compartilhadas.

cURL (Bash):

```
curl -X PUT "${BASE_URL}/api/area-material-tags/area/1" -H "Authorization: Bearer ${TOKEN}"
```

bash