

# LoadController Documentation

Controller responsável por gerenciar as operações relacionadas às cargas (loads).

## Endpoints

### GET /load

Retorna uma lista de cargas.

Parâmetro	Descrição	Tipo	Obrigatório
term	Termo de pesquisa	String	Não
offset	Offset para paginação (default: 0)	Int	Não
limit	Limite de resultados (default: 10000)	Int	Não
companies	Lista de IDs de empresas	String	Não

```
@PreAuthorize("hasAnyRole('ROLE_USER', 'ROLE_ADMIN', 'ROLE_MANAGER', 'ROLE_ROOT')")

@GetMapping

public ResponseEntity<Response<List<LoadModel>>> findAll(...)
```

### GET /load/company/{companyId}

Retorna uma lista de cargas para uma empresa específica.

Parâmetro	Descrição	Tipo	Obrigatório
term	Termo de pesquisa	String	Não
offset	Offset para paginação (default: 0)	Int	Não
limit	Limite de resultados (default: 10000)	Int	Não
companyId	ID da empresa	Long	Sim

```
@PreAuthorize("hasAnyRole('ROLE_USER', 'ROLE_ADMIN', 'ROLE_MANAGER', 'ROLE_ROOT')")

@GetMapping("/company/{companyId}")

public ResponseEntity<Response<List<LoadModel>>> findAllByCompanyId(...)
```

### GET /load/{id}

Retorna uma carga específica pelo seu ID.

Parâmetro	Descrição	Tipo	Obrigatório
id	ID da carga	Long	Sim

```
@PreAuthorize("hasAnyRole('ROLE_USER', 'ROLE_ADMIN', 'ROLE_MANAGER', 'ROLE_ROOT')")

@GetMapping("/{id}")

public ResponseEntity<Response<LoadModel>> findById(...)
```

## POST /load

Cadastra uma nova carga.

Parâmetro	Descrição	Tipo	Obrigatório
body	Dados da carga	Map	Sim

```
@PreAuthorize("hasAnyRole('ROLE_USER', 'ROLE_ADMIN', 'ROLE_MANAGER', 'ROLE_ROOT')")

@PostMapping

public ResponseEntity<Response<LoadModel>> save(...)
```

## PUT /load/{id}

Atualiza uma carga existente.

Parâmetro	Descrição	Tipo	Obrigatório
id	ID da carga	Long	Sim
item	Dados atualizados da carga	LoadModel	Sim

```
@PreAuthorize("hasAnyRole('ROLE_USER', 'ROLE_ADMIN', 'ROLE_MANAGER', 'ROLE_ROOT')")

@PutMapping("/{id}")

public ResponseEntity<Response<LoadModel>> update(...)
```

## DELETE /load/{id}

Remove uma carga pelo seu ID.

Parâmetro	Descrição	Tipo	Obrigatório
id	ID da carga	Long	Sim

```
@PreAuthorize("hasAnyRole('ROLE_USER', 'ROLE_ADMIN', 'ROLE_MANAGER', 'ROLE_ROOT')")

@DeleteMapping("/{id}")

public ResponseEntity<Response<LoadModel>> delete(...)
```