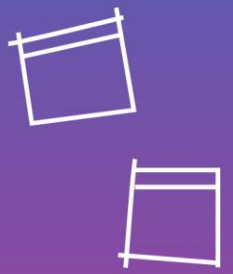
DBC
DIGITAL BUSINESS COMPANY®

VEM SER
DBC



# Spring Security

## Aula 3 – Regras e Permissionamento

# Sumário

- Recuperar Usuário do Token
- Criação de estrutura de tabelas para permissionamento
- Criação de regras

# Recuperar Usuário do Token

```
public Integer getIdLoggedUser() {
    Integer findUserId = (Integer) SecurityContextHolder.getContext().getAuthentication().getPrincipal();
    return findUserId;
}
```

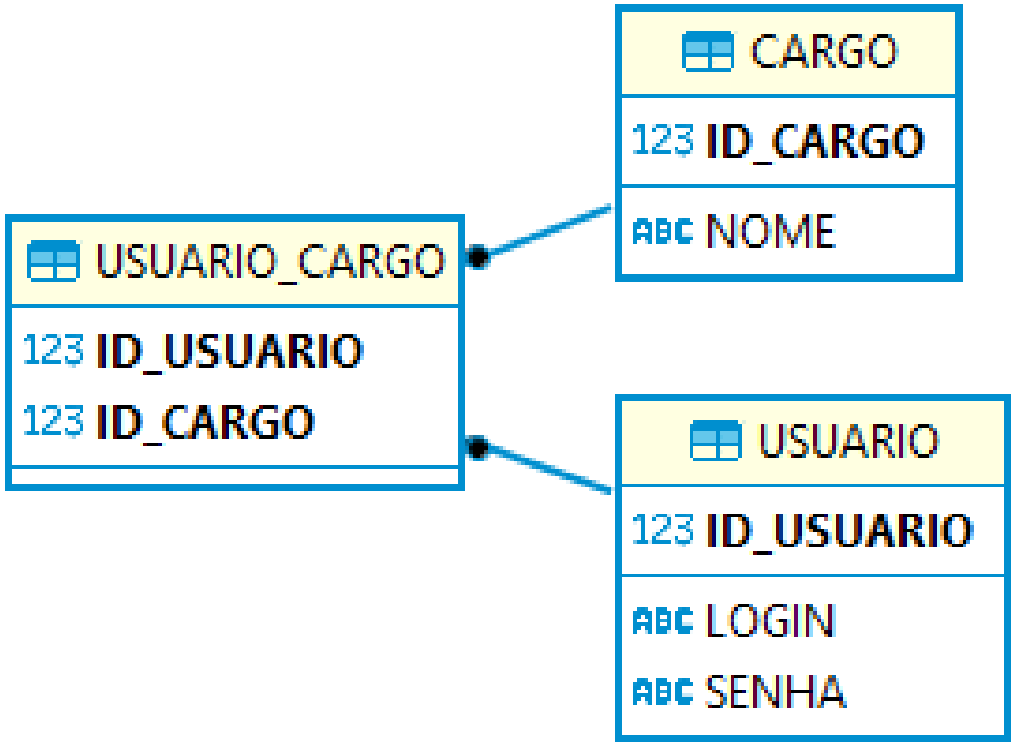
```
public UsuarioEntity getLoggedUser() throws BusinessException {
    return findById(getIdLoggedUser());
}

public UsuarioEntity findById(Integer idUsuario) throws BusinessException {
    return usuarioRepository.findById(idUsuario)
            .orElseThrow(() ->
                    new BusinessException("Usuário não encontrado!"));
}
```

# Exercício #1

- Recuperar usuário do contexto do security e criar um endpoint para exibir o usuário logado.

# Tabelas de Para Permissões Específicas



# Regras e Permissões

- No security, quando declaramos `.anyRequest().authenticated()`
  - Todo endpoint será protegido por uma autenticação

- Caso precisamos liberar alguma API sem autenticação:
  `.antMatchers("/auth").permitAll()`

# Permissões Específicas

- Um endpoint para uma regra:

```
.antMatchers("/pessoa").hasRole("ADMIN")
```

- Um endpoint para mais de uma regra:

```
.antMatchers("/contato/**").hasAnyRole("ADMIN", "MARKETING")
```

- Um endpoint e um método específico

```
.antMatchers(HttpMethod.GET, "/pessoa").hasRole("USUARIO")
```

- Vários endpoints para uma regra:

```
.antMatchers("/pessoa", "/endereco").hasRole("ADMIN")
```

DBC
DIGITAL BUSINESS COMPANY®

show me your code;

VEM SER
DBC

# Exercício Final / Homework

- Criar estrutura de permissões na base de dados (script_aula_3.sql)
  - Adaptar script caso haja mais de um usuário na sua base
- Mapear as entidades
- Inserir e recuperar as regras no token JWT
- Alterar usuários para inserir com cargos (lista de cargos)
  - Criar um endpoint do tipo post que recebe um usuário, senha e regras, cadastrar esse usuário na base de dados com a senha criptografada
- Criar a seguinte estrutura de permissões no SecurityConfiguration:
  - **ROLE_ADMIN**: pode acessar todos os endpoints
  - **ROLE_MARKETING**: pode acessar somente os gets de Pessoa, Contato e Endereço e Pet
  - **ROLE_USUARIO**: pode fazer qualquer operação em pessoa, contato e endereço
- Regra geral: somente o **ROLE_ADMIN** pode cadastrar usuários