

Universidade Federal de Viçosa
Campus Florestal

Trabalho Prático 03

ADRIELE DUTRA SOUZA - 1788

CLÁUDIO GUSTAVO SANTOS CAPANEMA - 1316

JOSE WESLEY DE SOUZA MAGALHÃES - 2267

LEONARDO JÚNIO ALVES DOS SANTOS - 2258

Florestal
Outubro de 2017

Sumário

1	Introdução	1
2	Desenvolvimento	2
2.1	Banco de Dados	3
2.1.1	Ator	3
2.1.2	Dependente	3
2.1.3	Emprestimo	4
2.1.4	Estúdio	4
2.1.5	Filme	5
2.1.6	Atores dos filmes	5
2.1.7	Estúdio dos filmes	6
2.1.8	Gêneros dos filmes	6
2.1.9	Fita	7
2.1.10	Funcionário	7
2.1.11	Gênero	7
2.1.12	Tipo de Usuário	8
2.1.13	Usuário	8
2.2	Aplicação	10
2.3	Exemplos de Execução	12
2.3.1	Consulta Real x Consulta Visualizada	16
3	Conclusão	17

1 Introdução

A segurança dos dados é de vital importância em ambientes de organizações em geral. Certas informações devem possuir níveis de segurança mais elevados que outras por serem mais confidenciais e não podem ser vistas por qualquer usuário. O controle de acesso as informações são o foco deste trabalho. Para isso, uma aplicação foi desenvolvida para realizar o devido tratamento das autorizações dos usuários.

O trabalho deu enfoque na propriedade de segurança simples, a qual diz que não é autorizado a um sujeito S o acesso de leitura a um objeto O , a não ser que:

$$\text{classe } (S) \geq \text{classe } (O).$$

Por isso a parte de segurança de acesso de leitura está mais fortemente presente e relacionada com o conceito de controle de acesso obrigatório visto na disciplina. A pouca representação da propriedade estrela será explicada em seções posteriores.

O contexto escolhido para o trabalho foi o de uma vídeo locadora, a VideoTOP. Nessa vídeo locadora trabalham um gerente, vários funcionários e possui vários clientes com seus dependentes. A escolha desse contexto se deve ao fato de que uma vídeo locadora possui diferentes tipos de usuários com distintas permissões. Um dependente menor de idade, por exemplo, não pode acessar vídeos que não são da sua faixa etária. Outro exemplo é que um atendente não pode ver informações de outros funcionários. Essa escolha se tornou um ambiente mais simples de simular o sistema de controle de acesso obrigatório.

Com esse sistema de segurança baseado em classes, os usuários da aplicação estão sendo "iludidos" de que estão vendo todas as tuplas e conteúdo do banco, quando na verdade estão vendo apenas aquelas permitidas pela sua classe de segurança.

2 Desenvolvimento

A vídeo locadora possui diferentes tipos de usuários e cada um tem permissões diferentes. Assim, as seguintes classes de seguranças foram consideradas para cada tipo de usuário:

Sigla	Classe	Tipo de Usuário
AS	Altamente Secretas	Gerente
S	Secretas	Atendente
C	Confidencial	Títular e Dependente Maior 18
U	Não Classificadas	Dependente Menor 18

Onde a classe U possui a menor permissão e a AS possui a maior permissão. Dessa forma, $U \leq C \leq S \leq AS$.

Os tipos de usuarios e suas permissões existentes no contexto abordado são:

- GERENTE: é permitido ao gerente visualizar os cadastros de todos os funcionários, bem como os cadastros de clientes e filmes.
- ATENDENTE: ao atendente é permitido cadastrar e listar filmes e clientes/dependentes. Não é permitido a um atendente visualizar informações de outros funcionários.
- CLIENTE: um cliente só pode ver informações dos filmes cadastrados e realizar empréstimos.
- DEPENDENTE: análogo ao cliente.

Caso a classe do usuário seja U, ou seja, a classe mais baixa, isso indica que o usuário é menor de idade. Sendo assim a aplicação faz esse tratamento e não exibe filmes que tenham uma classificação proibida para menores de 18 anos. Nesse caso, a classe de segurança desses filmes seria C.

2.1 Banco de Dados

O banco de dados criado, *videotopdb*, possui o uso de classe implementada em todas as tabelas. O SGBD utilizado foi o MySQLWorkbench em conjunto com o MySQLServer.

Os atributos que representam a classe possuem o mesmo nome do atributo original seguido do caracter 'C', a classe do atributo *nome* é representada pelo atributo *nomeC*, por exemplo.

Algumas tabelas possuem um atributo com mesmo nome da tabela concatenado com a sigla TC (*Tuple Classification*), a tabela *filme* possui o atributo *filmeTC*, por exemplo. O atributo TC representa a classe de cada tupla de uma tabela e por ele uma tupla é ou não mostrada em uma consulta.

Os valores inseridos nas tabelas, na maioria dos casos, representam a realidade. Pesquisas foram realizadas para gerar os comandos SQL de inserção. No caso, um único arquivo de inserções foi criado contendo dados para preencher as tabelas: filme, genero, ator, estudio, filme_tem_genero, filme_tem_ator, filme_tem_estudio e fita. Os dados das tabelas restantes são inseridos pela aplicação.

Todas as tabelas serão descritas a seguir. Importante resaltar que nas representações das tabelas a sigla AI significa Auto Increment e FK Foreign Key.

2.1.1 Ator

Alguns dos atores dos filmes são armazenados em uma única tabela, ator.

ator	
Atributo	Significado
idator	Identificador do ator (AI)
idatorC	Classe do idator
nome	Nome artístico do ator
nomeC	Classe do nome
dataNascimento	Data de Nascimento do ator
dataNascimentoC	Classe do dataNascimento
atorTC	Classe da tupla

Na tabela ator todos os atributos Classe possuem pré-definido o valor mais baixo de classe (*Default* = 'U'), já que a maioria dos atores fazem filmes para menores de idade (18 anos). Porém, quando o ator faz somente filmes para maiores ele possui todos os seu atributos Classe igual a 'C'.

2.1.2 Dependente

Os clientes titulares de uma conta podem possuir dependentes. Existem dois tipos de dependentes: maior de idade (acima de 18 anos) e menor de idade (abaixo de 18),

a diferença entre eles é a classe de permissão. Os maiores de idade possuem classe 'C' assim, podem alugar filmes com classificação para maiores de idade. Enquanto isso, os menores de idade possuem classe 'U' para terem uma permissão de acesso mais restrita aos filmes.

dependente	
Atributo	Significado
iddependente	Identificador do dependente (AI)
iddependenteC	Classe do dependente
cpf	CPF do titular da conta
cpfC	Classe do cpf
nome	Nome do dependente
nomeC	Classe do nome
dataNascimento	Data de Nascimento do dependente
dataNascimentoC	Classe do dataNascimento
iddependenteTC	Classe da tupla

Na tabela dependente todos os atributos Classe possuem pré-definido o valor mais baixo de classe (*Default* = 'U').

2.1.3 Empréstimo

A tabela empréstimo guarda as informações dos alugueis das fitas.

empréstimo	
Atributo	Significado
idfita	Identificador da fita (FK)
idfitaC	Classe do idfitaC (FK)
cpf	CPF do titular da conta (FK)
cpfC	Classe do cpf (FK)
dataEmprestimo	Data do empréstimo da fita
dataEmprestimoC	Classe do dataEmprestimo
dataDevolucao	Data de devolução da fita
dataDevolucaoC	Classe do dataDevolucao
valor	Valor do empréstimo
valorC	Classe do valor
emprestimoTC	Classe da tupla

Na tabela dependente todos os atributos Classe possuem pré-definido o valor mais baixo de classe (*Default* = 'U').

2.1.4 Estúdio

A tabela estúdio armazena os nomes e sedes dos estúdios dos filmes.

estudio	
Atributo	Significado
idestudio	Identificador da fita (FK)
idestudioC	Classe do idfitaC (FK)
cpf	CPF do titular da conta
cpfC	Classe do cpf
dataEmprestimo	Data do empréstimo da fita
dataEmprestimoC	Classe do dataEmprestimo
dataDevolucao	Data de devolução da fita
dataDevolucaoC	Classe do dataDevolucao
valor	Valor do empréstimo
valorC	Classe do valor
emprestimoTC	Classe da tupla

Todos os atributos Classe possuem pré-definido o valor mais baixo de classe (*Default*='U').

2.1.5 Filme

A tabela filme representa os atributos básicos de um filme. Para reduzir redundância foram criadas outras tabelas, e são elas: genero, estudio, ator. Estas tabelas possuem um relacionamento muitos para muitos com filme e por este motivo outras três tabelas foram criadas para representatividade.

filme	
Atributo	Significado
idfime	Identificador do filme (AI)
ifilmeC	Classe do idfime
titulo	Título do filme
tituloC	Classe do titulo
ano	Ano de lançamento do filme
anoC	Classe do ano
filmeTC	Classe da tupla

Os atributos Classe da tabela filme possuem pré-definido o valor mais baixo de classe (*Default* = 'U'). Mas, todos os filmes para maiores de idade possuem todos os atributos como 'C' já que eles não podem ser vistos por quem possui classe 'U', menores de idade.

2.1.6 Atores dos filmes

A tabela filme_tem_ator armazena a relação muitos para muitos entre filme e ator. Isso porque, um ator pode participar de mais de um filme e um filme pode ter mais de um ator.

filme_tem_ator	
Atributo	Significado
idfilme	Identificador do filme (FK)
idfilmeC	Classe do idfilme (FK)
idator	Identifiacdor do ator (FK)
idatorC	Classe do ator (FK)
nomePersonagem	Nome do personagem do ator no filme
nomePersonagemC	Classe do nomePersonagem
filme_tem_atorTC	Classe da tupla

Como a maioria dos atributos dessa relação possui classe 'U', os atributos Classe possuem pré-definido o valor mais baixo de classe (*Default* = 'U').

2.1.7 Estúdio dos filmes

A tabela filme_tem_estudio representa a relação muitos para muitos entre filme e estúdio. Dessa maneira, um estúdio pode produzir mais de um filme e um filme pode ser produzido por mais de um filme.

filme_tem_estudio	
Atributo	Significado
idfilme	Identificador do filme (FK)
idfilmeC	Classe do idfilme (FK)
idestudio	Identifiacdor do ator (FK)
idestudioC	Classe do ator (FK)

Como a maioria dos atributos dessa relação possui classe 'U', os atributos Classe possuem pré-definido o valor mais baixo de classe (*Default* = 'U').

2.1.8 Gêneros dos filmes

A tabela filme_tem_genero representa a relação muitos para muitos entre filme e genero. Já que, um filme pode ter mais de um gênero e um gênero possivelmente tem mais de um filme.

filme_tem_genero	
Atributo	Significado
idfilme	Identificador do filme (FK)
idfilmeC	Classe do idfilme (FK)
idestudio	Identifiacdor do ator (FK)
idestudioC	Classe do ator (FK)

Como a maioria dos atributos dessa relação possui classe 'U', os atributos Classe possuem pré-definido o valor mais baixo de classe (*Default* = 'U').

2.1.9 Fita

Os filmes geralmente possuem mais de uma fita para melhor atender os clientes. Por este motivo, a tabela fita foi criada para armazenar cada fita de um filme.

fita	
Atributo	Significado
idfita	Identificador da fita (AI)
idfitaC	Classe do idfita
idfilme	Identificador do filme (FK)
idfilmeC	Classe do idfilme (FK)

Como a maioria dos atributos dessa relação possui classe 'U', os atributos Classe possuem pré-definido o valor mais baixo de classe (*Default* = 'U').

2.1.10 Funcionário

A classe funcionário herda da classe usuário. Assim, a tabela funcionario liga com a tabela usuario através da chave composta cpf - cpfC.

funcionario	
Atributo	Significado
cpf	CPF do funcionário (FK)
cpfC	Classe do cpf (FK)
salario	Salário do funcionário
salarioC	Classe do salario
funcionarioTC	Classe da tupla

Como a maioria dos atributos dessa relação possui classe 'U', os atributos Classe possuem pré-definido o valor mais baixo de classe (*Default* = 'U').

2.1.11 Gênero

A tabela genero armazena os gêneros dos filmes com uma breve descrição.

genero	
Atributo	Significado
idgenero	Identificador do gênero (AI)
idgeneroC	Classe do idgenero
nome	Nome do gênero
nomeC	Classe do nome
descricao	Descrição do gênero
descricaoC	Classe da descricao
generoTC	Classe da tupla

Como a maioria dos atributos dessa relação possui classe 'U', os atributos Classe possuem pré-definido o valor mais baixo de classe (*Default* = 'U').

2.1.12 Tipo de Usuário

Como já descrito anteriormente, o sistema possui diferentes tipos de usuário com classes distintas.

tipousuario	
Atributo	Significado
idtipousuario	Identificador do Tipo do Usuário (AI)
idtipousuarioC	Classe do idtipousuario
nome	Nome do Tipo do Usuário
nomeC	Classe do nome
classe	Classe do Tipo do Usuário
classeC	Classe da classe
tipousuarioTC	Classe da tupla

Como a maioria dos atributos dessa relação possui classe 'U', os atributos Classe possuem pré-definido o valor mais baixo de classe (*Default* = 'U').

2.1.13 Usuário

Como os clientes titulares e os funcionários possuem uma grande quantidade de atributos em comum a tabela usuario é a superclasse. Porém, somente os funcionários possuem atributos distintos. Dessa forma, não foi necessário a criação de uma tabela para o cliente que herdasse da tabela usuario.

usuario	
Atributo	Significado
cpf	CPF do usuário
cpfC	Classe do cpf
senha	Senha de acesso ao sistema
senhaC	Classe da senha
nome	Nome do usuário
nomeC	Classe da nome
dataNascimento	Data de Nascimento do usuário
dataNascimentoC	Classe da dataNascimento
logradouro	Logradouro (parte do endereço)
logradouroC	Classe do logradouro
numero	Número do logradouro (parte do endereço)
numeroC	Classe do numero
bairro	Bairro do logradouro (parte do endereço)
bairroC	Classe do bairro
cep	CEP do logradouro (parte do endereço)
cepC	Classe da cep
telefone	Telefone de contato do usuário
telefoneC	Classe do telefone
idtipousuario	Classe do Tipo do Usuário (FK)
idtipousuarioC	Classe do idtipousuario (FK)
tipousuarioTC	Classe da tupla

Como a maioria dos atributos dessa relação possui classe 'U', os atributos Classe possuem pré-definido o valor mais baixo de classe (*Default* = 'U').

2.2 Aplicação

A aplicação para este trabalho foi desenvolvida na linguagem JAVA na IDE NetBeans versão 8.2. A escolha da linguagem Java se deu devido ao fato de facilitar o trabalho com interfaces gráficas com o usuário. A arquitetura do software utiliza o padrão MVC. Na camada de Modelo, estão as classes de entidade que representam a maioria das entidades presentes no banco, são elas Ator, Genero, Estudio, Filme, Usuario, Funcionario, Empréstimo, e TipoUsuario. Lembrando que mesmo com o enum TipoUsuario, a classe Usuario armazena a classe do mesmo, porque tanto dependente como cliente podem ser da classe U ou C. Na camada de controle temos as classes controladorFilmes, controladorUsuarios e controladorEmpréstimos, que fazem toda a manipulação das classes de Modelo e fazem a transição entre a Visão e o banco de dados. A camada mais superficial, a camada da Visão, faz o contato direto com o Usuario e contém as classes visuais que interagem com o usuário, as telas da aplicação. São elas TelaLogin, TelaPrincipal, CadastrarUsuarios, CadastrarFilmes, TodosUsuarios (tela que mostra a lista de usuários cadastrados, é a mesma para clientes/dependentes e funcionarios tendo alterações quando é chamada), TodosFilmes (análoga a classe TodosUsuarios) e RealizarEmpréstimo.

Foi criado o pacote Persistência apenas para guardar a classe ConexaoBanco (cujo nome é bem intuitivo). Isso não quer dizer que uma persistência em memória secundária foi simulada. Todos os dados manipulados pela aplicação estão armazenados no banco de dados. A conexão com o banco foi feita através da biblioteca *mysql-connector-java* e os recursos de interação com SQL oferecidos por JAVA, como `sql.Connection` para fazer conexão no banco, `sql.ResultSet` para armazenar resultados de consultas, por exemplo.

A aplicação desenvolvida permite cadastrar, excluir, editar e visualizar quaisquer usuarios e permite inserir, visualizar e excluir filmes. Para cada uma dessas operações a classe controladora correspondente irá acionar a classe ConexaoBanco para fazer as operações no banco de dados. Para inserir, editar, excluir e visualizar na aplicação; são passadas para o banco os comandos INSERT, UPDATE, DELETE E SELECT respectivamente (com as devidas cláusulas e condições devidamente atribuídas). Dessa forma, qualquer operação que um usuário faz, ele está implicitamente fazendo consultas no banco de dados.

Inicialmente, a aplicação abre a tela de login para que o usuário acesse sua conta. Após fazer o login, a aplicação irá analisar a classe de segurança do usuário para verificar quais permissões e funcionalidades da aplicação podem ser acessadas pelo usuário; exibindo telas diferentes para usuários diferentes. O tratamento é feito com base nas classes de segurança do usuario.

Como o primeiro contato com o usuário é a tela de login e a locadore possui apenas um gerente, este mesmo gerente deve estar pré-cadastrado no banco antes da aplicação ser executada pela primeira vez.

Para executar esse mecanismo de restrições e permissões, a aplicação opta por não exibir as informações e dar acessos limitados através das interfaces gráficas. Por exemplo, no caso de um atendente fazer login, a tela que irá aparecer é a mesma que aparece quando um gerente faz login no sistema; porém ao verificar que é um atendente, a aplicação bloqueia e não exibe os botões que dão acesso às telas de cadastro e visualização de funcionários. Esse botões só são ativados caso a aplicação verifique que um gerente está usando o sistema no momento. No caso de um cliente ou dependente entrar no sistema, a aplicação irá abrir uma interface diferente onde o usuário apenas poderá ver filmes e realizar empréstimos. Para diferenciar essas telas são passados valores booleanos no construtor dessas classes.

As telas para cadastros são a mesma classe, e utilizam da mesma estratégia descrita anteriormente, exibindo e bloqueando componentes na tela de acordo com o cadastro que se deseja fazer. Especificamente no caso de cadastrar um dependente, há uma caixa de seleção que quando marcada ativa um campo para ser inserido o CPF do titular ao qual o dependente é vinculado.

Outro exemplo é o tratamento para usuários classe U. Como U é a classe mais baixa, a aplicação irá filtrar os filmes do que possuem classe de segurança $l=U$. Para isso quando a opção de listar os filmes for acionada, a aplicação confere que a classe do usuário é menor que a dos filmes +18 e não adicionará na tabela que aparece para o usuário tais filmes.

Um problema que a aplicação possui está relacionado com a parte de escrita no banco. No contexto criado, apenas gerentes e atendentes podem inserir dados no banco (usuarios em geral e filmes). Portanto a propriedade estrela não está muito presente, pois essas inserções no banco não causariam polinstantiação visto que as classes de gerente e atendente são as mais alta e por isso conseguem escrever livremente no banco. Nos casos que a propriedade estrela seria aplicada, a própria aplicação faz o controle do acesso e segurança, evitando que seja necessária tratar polinstantiação.

2.3 Exemplos de Execução

Nesta seção serão apresentados exemplos de execução, mostrando como a aplicação se comporta para classes de segurança diferentes.

A image shows a login window with a dark gray title bar containing standard window control buttons (close, maximize, minimize). The main area is light gray. It features two input fields: the first is labeled 'CPF:' and contains the text '123.456.789-00'; the second is labeled 'Senha:' and contains five dots. Below these fields is a blue button with the text 'Entrar'.

Figura 1: Tela de Login da Aplicação

Como dito anteriormente, depois de feito o login no sistema, a aplicação irá direcionar o usuário para alguma outra tela dependendo de sua classe de segurança. Abaixo temos um exemplo na diferença entre as telas de um funcionario "comum" (atendente), classe S e de um gerente, classe AS.

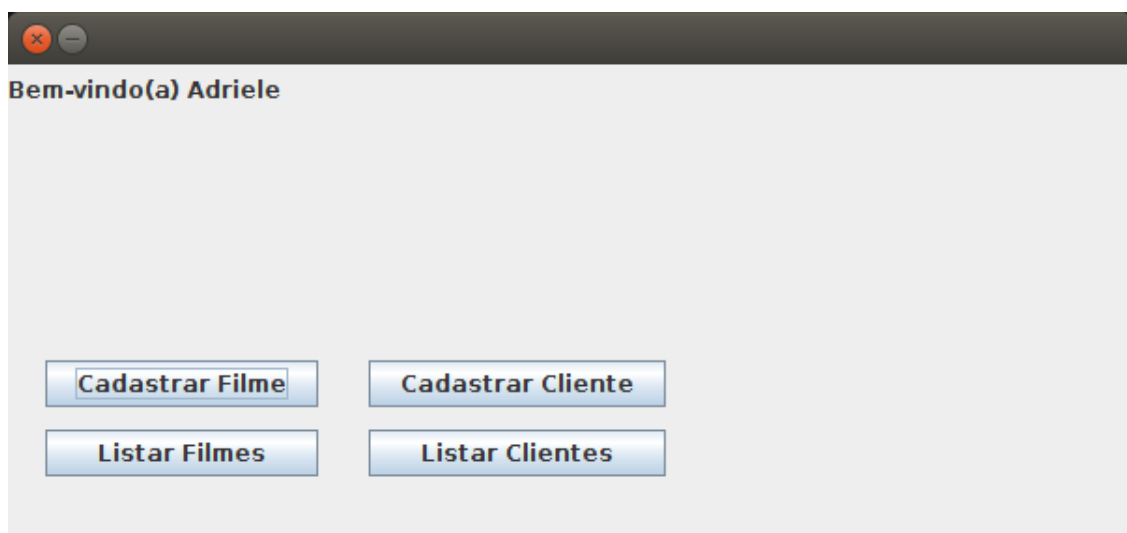
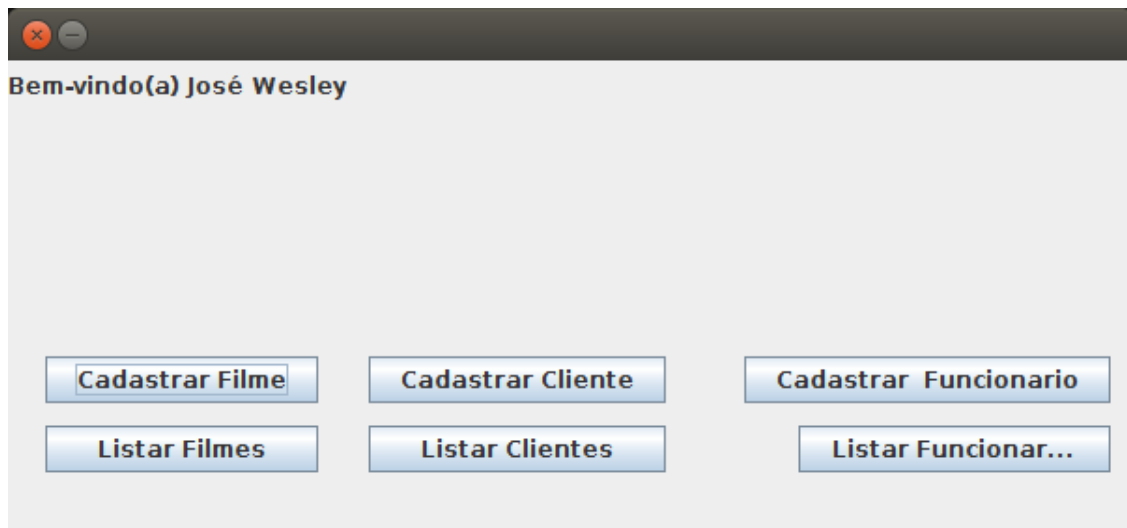


Figura 2: Diferença nas telas de Gerente e Atendente

Note que para um Gerente ("José Wesley"), estão disponíveis as opções tanto de cadastrar como de listar funcionários, o que não acontece para a atendente ("Adriele"). Os botões não ficam visíveis. Entretanto ambos podem cadastrar e listar clientes da locadora.

A telas para cadastro de funcionários e clientes, embora sejam na verdade a mesma classe, também se comportam de maneira diferente dependendo da classe de segurança do usuário.

A screenshot of a web application window titled "CADASTRAR FUNCIONARIO". The form contains the following fields: "Nome:" (a single-line text box), "CPF:" (a text box with pre-defined dots and dashes), "Data de Nascimento:" (a date picker with slashes), "Telefone:" (a text box), "Senha:" (a text box), "Logradouro:" (a text box), "Número:" (a text box), "Bairro:" (a text box), "CEP:" (a text box), and "Salário:" (a text box). At the bottom right, there are two buttons: "Cadastrar" and "Cancelar".

Figura 3: Tela de Cadastro de Funcionários

A screenshot of a web application window titled "CADASTRAR CLIENTE". The form contains the following fields: "Nome:" (a single-line text box), "CPF:" (a text box with pre-defined dots and dashes), "Data de Nascimento:" (a date picker with slashes), "Telefone:" (a text box), "Senha:" (a text box), "Logradouro:" (a text box), "Número:" (a text box), "Bairro:" (a text box), "CEP:" (a text box), and "CPF Titular:" (a text box with pre-defined dots and dashes). There is also a checkbox labeled "Dependente?". At the bottom right, there are two buttons: "Cadastrar" and "Cancelar".

Figura 4: Tela de Cadastro de Clientes

Caso esteja sendo realizado um cadastro de funcionário, será exibido o campo Salário e oculto a caixa de seleção Dependente, bem como o campo CPF Titular.

Finalmente, demonstrando outra restrição relacionada à classe, mas dessa vez em relação a maioria do usuário, abaixo encontram-se duas telas de um cliente e de um dependente que fizeram login, sendo um maior de idade (classe=C) e um dependente menor de idade (classe=U).

idFilme	Titulo	Ano
1	Harry Potter e a Pedra Filosofal	2001
2	A mulher de Preto	2012
3	A bela e a Fera	2017
4	Duro de matar	1988
5	Looper: Assassinos do Futuro	2012
6	Batman: O Cavaleiro das Trevas Ressurge	2012
7	Trapaça	2013
8	Jogos Vorazes	2012
9	X-Men: Primeira Classe	2011
10	Mad Max: Estrada da Fúria	2015
11	Transformers: Dark of the Moon	2011
12	Transformers	2007
13	Carros	2006
14	Marley & Eu	2008
15	Toy Story	1995
16	Náufrago	2000
17	Forrest Gump: O Contador de Histórias	1994
18	O Expresso Polar	2004
19	A Saga Crepúsculo: Amanhecer - Parte 2	2012
20	Harry Potter e o Cálice de Fogo	2005
21	Sr. & Sra. Smith	2005
22	Branquinha	2016
23	Evil Dead	2013
24	O Jardim de Cimento	1993
25	Bastardos Inglórios	2009

Figura 5: Lista de Filmes para um usuário classe C

idFilme	Titulo	Ano
1	Harry Potter e a Pedra Filosofal	2001
2	A mulher de Preto	2012
3	A bela e a Fera	2017
4	Duro de matar	1988
5	Looper: Assassinos do Futuro	2012
6	Batman: O Cavaleiro das Trevas Ressurge	2012
7	Trapaça	2013
8	Jogos Vorazes	2012
9	X-Men: Primeira Classe	2011
10	Mad Max: Estrada da Fúria	2015
11	Transformers: Dark of the Moon	2011
12	Transformers	2007
13	Carros	2006
14	Marley & Eu	2008
15	Toy Story	1995
16	Náufrago	2000
17	Forrest Gump: O Contador de Histórias	1994
18	O Expresso Polar	2004
19	A Saga Crepúsculo: Amanhecer - Parte 2	2012
20	Harry Potter e o Cálice de Fogo	2005
21	Sr. & Sra. Smith	2005

Figura 6: Lista de Filmes para um usuário classe U

Podemos notar que para o usuário classe U (Dependente menor de idade) não foram listados os filmes com id de 22 a 25, por serem filmes proibidos para menores de idade.

2.3.1 Consulta Real x Consulta Visualizada

Tendo em vista os níveis de segurança apresentados e suas respectivas restrições, ao realizar alguma operação no banco de dados, seja remover, inserir, alterar ou simplesmente visualizar, são dispostas duas formas de consulta, que chamaremos de real e visualizada.

A consulta real consiste na forma tradicional realizada a partir de comandos. Considerando um ou mais parametros de busca é possível realizar a consulta e posteriormente visualizar e até alterar todo o conteúdo que está disponível no banco de dados. Porém, devido a atribuição de níveis específicos de segurança, algumas informações são omitidas de acordo com a restrição pre-estabelecida pelo DBA. Neste caso, temos o conceito de consulta visualizada, no qual, considerando parametros de busca é possível realizar a consulta como no caso anterior, porém com alguns campos omitidos.

No contexto do trabalho prático em si, a consulta visualizada é predominante, uma vez a aplicação foi definida de forma que cada usuário e filme tem suas respectivas restrições.

3 Conclusão

Com a realização desse trabalho foi possível fixar e compreender melhor os conceitos de controle de acesso obrigatório em bancos de dados, como funcionam as classes de segurança e a sua importância para manter a integridade de uma base de dados.

Ainda falando em questões de aprendizado, foi de alta relevância para o grupo a etapa de criar a conexão de uma aplicação de software com um banco de dados e ver isso realmente funcionando na prática nos deu uma maior dimensão de como os softwares no mercado real operam, principalmente em relação a sistemas que devem manipular dados e informações em grande quantidade e de alta relevância.