

13. Trabalho de PHP – Parte VII e VIII

Aproveitando o exercício da atividade PHP com Banco de Dados MySQL, onde o sistema trabalhava com uma única tabela de acessos do usuários, incremente as seguintes alterações:

- Além da tabela acesso, crie uma tabela usuário com os atributos: login (pk¹), nome, data de nascimento, sexo, endereço completo (Logradouro, número, complemento, bairro, cidade, UF e CEP), conforme mostrado no diagrama de classe da figura 11:

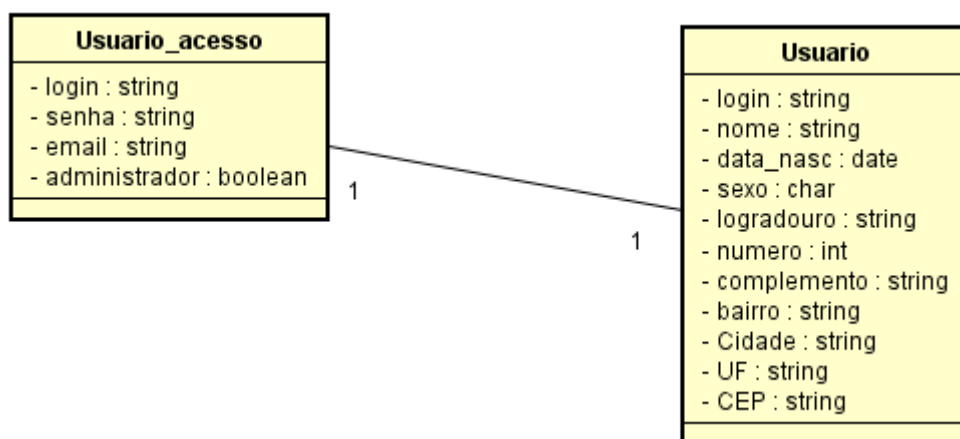


Figura 11 - Diagrama de Classe – Usuário

- Agora o cadastro deve também conter essas informações e conseqüentemente todas as opções de CRUD (Create, Read, Update, Delete) que quer dizer basicamente as 4 (quatro) principais operações com um banco de dados (inserir, ler, atualizar, excluir).
- A opção “create”, se dá quando criamos e/ou cadastramos um usuário novo. A opção “read” é acionada quando busca uma ou mais informações armazenadas no banco de dados. Já a opção “update” é ativada quando se pretende alterar alguma(s) informação(ões), por exemplo, mudança de endereço e finalmente a opção “delete” é colocada em funcionamento quando se deseja excluir um usuário no banco de dados.
- Nessa primeira versão, só será permitido alterar informações referentes ao endereço.

¹ Pk – primary key – Campo chave

- Nessa versão do trabalho, o acionamento da opção “read” será no instante que o usuário do sistema sair do foco do login, por ser um campo chave primária, recomenda-se disparar o evento on blur². Então nesse momento será dado um select no banco de dados e buscado as informações (se houver) referentes ao login. Esse select deve buscar informações das duas tabelas (ver o join no exemplo a seguir)

```
SELECT * FROM USUARIOS_ACESSO AS UA  
JOIN USUARIOS AS U ON UA.LOGIN = U.LOGIN
```

Caso o login esteja cadastrado no banco de dados, as informações desse login deve ser preenchidas automaticamente no formulário e apresentados na tela. O usuário do sistema então poderá alterar as informações referente ao endereço, já que o usuário já existe no sistema. Nessa situação o botão gravar do formulário irá executar o código referente ao update no banco de dados.

Caso contrário, isto é, o login não está cadastrado no banco de dados, o select não retornará nenhum valor, então o formulário vai estar com os campos vazios para serem preenchidos. Nessa situação o botão gravar do formulário irá executar o código referente ao insert no banco de dados.

Dicas:

- Ao gravar um novo usuário, as informações devem ser inseridas nas duas tabelas, resumindo, você terá um único formulário com todas as informações, então ao clicar no botão gravar do formulário, deve ser dados dois inserts: um na tabela usuario_acesso e outro na tabela usuario.
- Como somente pode alterar informações sobre o endereço, basta então um comando update para a tabela usuário.
- Ao remover um usuário, os dados devem ser apagados das duas tabelas. Você pode usar dois deletes (um para cada tabela) **ou** usar delete on cascade (visto em banco de dados)

² On blur - o evento onBlur ocorre quando um objeto perde o foco, então uma ação é disparada

Observações

- Todos os campos são obrigatórios com a exceção do complemento.
- O campo referente a informação do sexo, deve ser apresentado em forma de radio no formulário
- O campo referente ao estado deve ser uma combobox.
- Para remover um usuário, deve existir um botão com essa finalidade. Lembre-se que para remover, o login do usuário deve ser digitado e encontrado no banco de dados (claro ele tem que existir). Então recomendo só disponibilizar o botão remover, isto é, **enable=true**, se for encontrado o login no banco de dados, mas como ter essa informação? No evento onblur do login.
- Use a criatividade para a interface.

Sequência lógica das telas do sistema

O cadastro proposto servirá como base para a tela inicial de login do sistema, não confunda cadastro de usuário com acesso ao sistema. O diagrama de sequência da figura 12 mostra a sequência lógica do usuário no sistema.

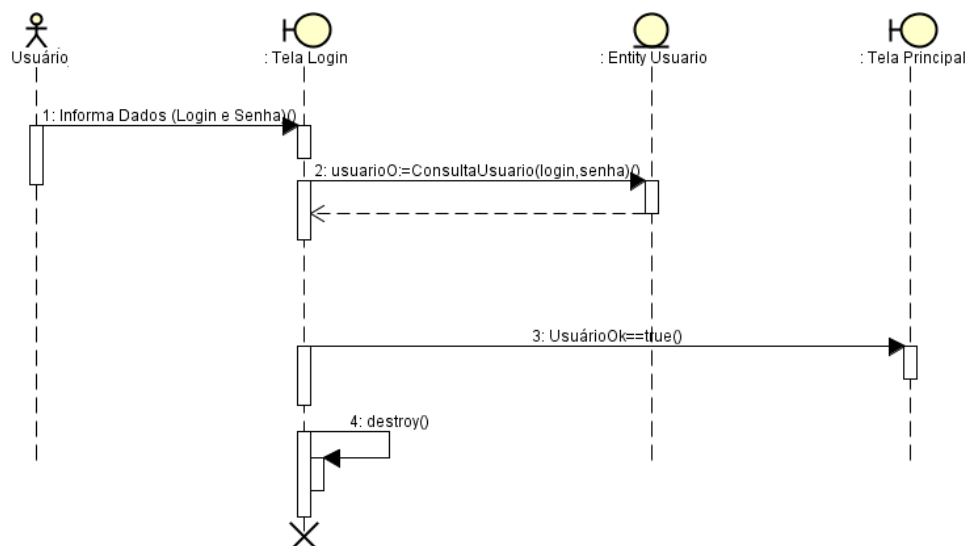


Figura 12 - Diagrama de sequência – Login

Repare da necessidade da tela de login para acessar o sistema, **somente com acesso autorizado e sendo administrador que o usuário poderá cadastrar um novo usuário**. Mas como fica a primeira vez de uso do sistema se não tem nenhum usuário cadastrado? Para essa situação será criado direto no banco de dados um usuário

chamado “*master*” com privilégio de administrador. Então após criar as tabelas (ver o diagrama de classe da figura 11), insira no banco o seguinte registro:

```
Insert into usuário_acesso values  
(‘master’,‘master@123’,‘master@gmail.com’,true);
```

Uma vez inserido, novos usuário poderão ser cadastrados. Lembre-se, somente usuário com perfil de administrador poderá inserir usuários no sistema. As figuras 13 e 14 ilustram a lógica da sequência de telas de um sistema utilizado como exemplo .



Figura 13 - Sequência 1 (Usuário **com** acesso de administrador)

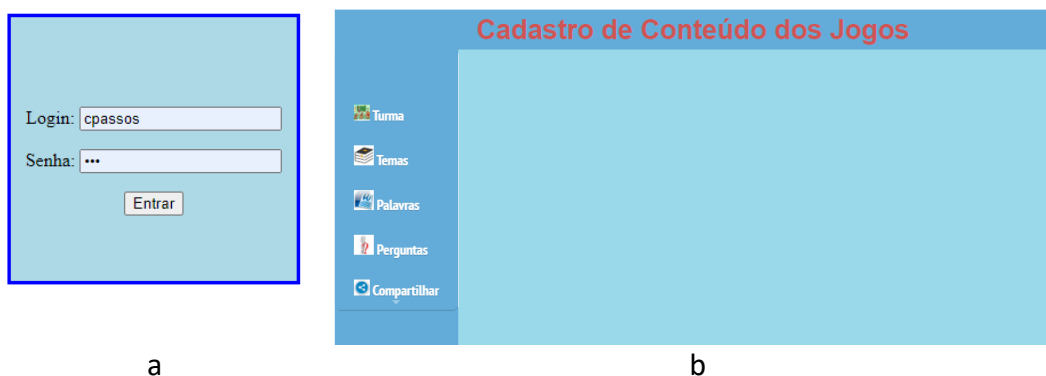


Figura 14 - Sequência 2 (Usuário **sem** acesso de administrador)

Repare o menu à esquerda da figura 14b não aparece o item usuários. Essas telas só foram utilizadas como exemplo, por enquanto, o sistema proposto nesse trabalho, só tem o cadastro de usuários por enquanto.