



**INSTITUTO
FEDERAL**

Sudeste de
Minas Gerais

Campus
Manhuaçu

Desenvolvimento Web III

Rotinas para edição

Prof. Leonardo C. R. Soares - leonardo.soares@ifsudestemg.edu.br

Instituto Federal do Sudeste de Minas Gerais

17 de janeiro de 2025





Atualização de dados

A rotina para edição de dados, de forma geral, possui duas etapas bem simples:





Atualização de dados

A rotina para edição de dados, de forma geral, possui duas etapas bem simples:

- Exibir os dados atualmente cadastrados para que o usuário faça as alterações;





Atualização de dados

A rotina para edição de dados, de forma geral, possui duas etapas bem simples:

- ▶ Exibir os dados atualmente cadastrados para que o usuário faça as alterações;
- ▶ Salvar as alterações.





Alterações na classe Usuário

- Crie um método que retorne os dados de um usuário específico a partir do e-mail do mesmo.





Alterações na classe Usuario

```
public static function pegaUser(string $email):?Usuario {  
    try{  
        $con = Conexao::conectar();  
    } catch(PDOException $p){  
        throw new PDOException($p->getMessage());  
    }  
    try{  
        $SQL = "select * from usuarios where email=?";  
        $pst = $con->prepare($SQL);  
        $pst->bindParam(1,$email);  
        $pst->execute();  
        $dado = $pst->fetch(PDO::FETCH_OBJ);  
        if ($pst->rowCount()>0){  
            $usuario = new Usuario($dado->nome,$dado->email,"",$dado->nivel);  
            return $usuario;  
        }else  
            return null;  
    }catch(PDOException $p){  
        throw new PDOException("Ocorreu um erro ao tentar acessar a base de dados. <br>Verifique: <i>" .  
            $p->getMessage()."</i>");  
    }  
}
```





Alterações na classe Usuário

- Crie um método que atualize os campos da tabela usuários. O campo e-mail é a chave primária e por isso não será atualizado. Caso o usuário **não** deseje alterar a senha, deverá deixá-la em branco.





Alterações na classe Usuário

```
public static function atualizar(string $nome, string $email, int $nivel, string $senha = "" ): void {  
    try{  
        $con = Conexao::conectar();  
    } catch(PDOException $p){  
        throw new PDOException($p->getMessage());  
    }  
    try {  
        if ($senha == "")  
            $SQL = "update usuarios set nome=:nome, nivel=:nivel where email=:email";  
        else  
            $SQL = "update usuarios set nome=:nome, nivel=:nivel, senha=md5(:senha) where email=:email";  
        $st = $con->prepare($SQL);  
        $st->bindParam(':nome',$nome);  
        $st->bindParam(':nivel',$nivel);  
        $st->bindParam(':email',$email);  
        if($senha!="")  
            $st->bindParam(':senha',$senha);  
        $st->execute();  
    }catch(PDOException $p){  
        throw new PDOException("Ocorreu um erro ao tentar atualizar um usuário. <br>Verifique: <i>" .  
            $p->getMessage()."</i>");  
    }  
}
```





Alterações na página verTodos.inc.php

- Na página que apresenta a listagem dos usuários, vamos acrescentar um link para editar um determinado usuário.





Alterações na página verTodos.inc.php

```
<table class="table">
  <thead>
    <tr>
      <th scope="col">Nome</th>
      <th scope="col">Email</th>
      <th scope="col">Nível</th>
    </tr>
  </thead>
  <tbody>
    <?php foreach($usuarios as $linha) { ?>
      <tr>
        <th scope="row"><?=$linha->getNome();?></th>
        <td><?=$linha->getEmail();?></td>
        <td><?=$linha->getNivel()==1 ? "Administrador" : "Usuário";?></td>
        <td><a class="nav-link" href="principal.php?pg=<?=base64_encode("altUser");?>&x=<?=base64_encode($linha->getEmail());?>" title="Editar <?=$linha->getNome();?>">Editar</a></td>
        <td><a class="nav-link" href="principal.php?c=<?=base64_encode("delUser");?>&x=<?=base64_encode($linha->getEmail());?>" title="Excluir <?=$linha->getNome();?>">Excluir</a></td>
      </tr>
    <?php } ?>
  </tbody>
</table>
```





Atualização dos dados

O link criado para a edição segue o padrão definido para a inclusão de arquivos, assim, a página principal irá fazer a inclusão da página *altUser.inc.php*.





Atualização dos dados

O link criado para a edição segue o padrão definido para a inclusão de arquivos, assim, a página principal irá fazer a inclusão da página *altUser.inc.php*.

A página *altUser.inc.php* será responsável exibir os dados atualmente cadastrados para que o usuário possa fazer as alterações desejadas.





altUser.inc.php

```
1 <?php
2 $us = Usuario::pegarUser(base64_decode($_REQUEST['x']));
3 ?>
4 <h1>Atualização de usuário</h1>
5 <form action="principal.php?c=<?=base64_encode('updateUser');?>" method="post">
6     <div class="form-group">
7         <label for="nome">Nome</label>
8         <input type="text" class="form-control" id="nome" name="nome" value="<?=$us->getNome();?>">
9     </div>
10    <div class="form-group">
11        <label for="email">E-mail</label>
12        <input type="text" class="form-control" id="email" name="email" value="<?=$us->getEmail();?>" readonly
13        aria-describedby="emailHelp" >
14        <small id="emailHelp" class="form-text text-muted">O e-mail não pode ser alterado.</small>
15    </div>
```





altUser.inc.php

```
15 <div class="form-group">
16   <label for="nivel">Nível</label>
17   <select class="form-control" id="nivel" name="nivel">
18     <option value="0" <?=$us->getNivel()==0 ? " selected " : "" ;?> >0 - Usuário</option>
19     <option value="1" <?=$us->getNivel()==1 ? " selected " : "" ;?> >1 - Administrador</option>
20   </select>
21 </div>
22 <div class="form-group">
23   <label for="senha">Senha</label>
24   <input type="password" class="form-control" id="senha" name="senha" aria-describedby="senhaHelp">
25   <small id="senhaHelp" class="form-text text-muted">Caso não deseje alterar a senha, deixe em branco.</small>
26 </div>
27 <div class="form-group">
28   <label for="csenha">Confirme a nova senha</label>
29   <input type="password" class="form-control" id="csenha" name="csenha" >
30 </div>
31 <button type="submit" class="btn btn-primary">Salvar alterações</button>
32 </form>
```





Atualização dos dados

A página principal terá que ser atualizada para **executar a ação referente à atualização dos dados** (veja o *action* do formulário *altUser.inc.php*).





principal.php

```
1 <?php
2 require_once "session.php";
3 $usuario = unserialize($_SESSION['usuario']);
4 $nome = $usuario->getNome();
5 ?>
6 <!doctype html>
7 <html lang="pt-br">
8     <head>
9         <meta charset="utf-8">
10         <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.min.css" rel="stylesheet"
            integrity="sha384-1BmE4kWBq78iYhFldvKuhfTAU6auU8tT94WrHftjDbrCEXSU1oBoqyl2QvZ6jIW3"
            crossorigin="anonymous">
11         <title>Menu principal</title>
12     </head>
13 <body>
```





principal.php

```
14 <div class="container">
15   <header class="d-flex flex-wrap justify-content-center py-3 mb-4 border-bottom">
16     <a href="/" class="d-flex align-items-center mb-3 mb-md-0 me-md-auto text-dark text-decoration-none">
17       <span class="fs-4"><?=$nome;?></span>
18     </a>
19     <ul class="nav nav-pills">
20       <li class="nav-item"><a href="principal.php" class="nav-link">Home</a></li>
21       <?php
22         // Link visível apenas para usuários nível 1 (administradores)
23         if ($usuario->getNivel()==1)
24           echo '<li class="nav-item"><a href="principal.php?pg=' . base64_encode("verTodos") . '" title="Ver
25             usuários cadastrados" class="nav-link">Usuários</a></li>'
26           ?>
27       <li class="nav-item"><a href="logout.php" class="nav-link">Logout</a></li>
28     </ul>
29   </header>
30   <?php
31   // Se existir o parâmetro PG, recebemos uma página que será incluída.
32   if(isset($_GET['pg'])){
33     include base64_decode($_GET['pg']).".inc.php";
34   }
```





```
34 // Se existir o parâmetro C, recebemos um comando a ser executado.
35 if (isset($_GET['c'])){
36     // Apenas usuários de nível 1 podem executar essas operações
37     if ($_usuario->getNivel()==1) {
38         if (base64_decode($_GET['c'])=="delUser"){
39             try{
40                 Usuario::delUser(base64_decode($_GET['x']));
41                 echo '<div class="alert alert-success" role="alert">O usuário '. base64_decode($_GET['x']) .' foi
42                 excluído com sucesso. </div>';
43             }catch(PDOException $p){
44                 echo '<div class="alert alert-danger" role="alert">Atenção: '. $p->getMessage() .'</div>';
45             }
46         }
47         if (base64_decode($_GET['c'])=="updateUser"){
48             try{
49                 Usuario::atualizar($_POST['nome'],$_POST['email'],$_POST['nivel'],$_POST['senha']);
50                 echo '<div class="alert alert-success" role="alert">O usuário ' .$_POST['nome'] .' foi atualizado com
51                 sucesso. </div>';
52             }catch(PDOException $p){
53                 echo '<div class="alert alert-danger" role="alert">Atenção: '. $p->getMessage() .'</div>';
54             }
55         }
56     } else {
57         echo '<div class="alert alert-warning" role="alert">Operação não permitida. Consulte um administrador do
58         sistema. </div>';
59     }
60 }
61 ?>
62 </div>
63 </body>
64 </html>
```





Dúvidas?





Exercício

- ▶ Implemente o exemplo. Acrescente ao código uma opção para que o usuário logado possa alterar a própria senha (de forma isolada).
- ▶ Altere o sistema da LibSoft (CEP) para que os administradores do sistema possam alterar os endereços já cadastrados.



