

6 – APLICAÇÃO WEB – GERENCIAMENTO USUÁRIOS

Neste capítulo foi elaborado e desenvolvido uma as aplicações web para gerenciamento de usuário, onde realiza-se o CRUD para usuário. A aplicação foi desenvolvida utilizando-se *HTML*, *CSS*, *PHP* e Banco de Dados *MySQL* e uma biblioteca externa para estilização chamada *Bootstrap*, através de várias técnicas e estudos para seja totalmente adequado aos parâmetros deste assunto.

6.1 Desenvolvimento

Em consideração a este estudo, a disciplina do curso (Tópicos Especiais em Informática), solicitou e orientou em criar a parte de gerenciamento de usuário da aplicação web de produtos e usuário em HTML, CSS e PHP e Banco de Dados SQL onde utilizou-se a biblioteca Bootstrap para estilização CSS.

Figura 6.1 Tela de listagem de usuário

Listagem de Usuario

Login	Nome	Senha	Email	Permissão	
disterTeste	Jonathan Teste	123456	jonhatanTeste@email.com	ADMIN	Editar Excluir
fireTeste	MarcusTeste	2233441122	marcusTeste@email.com	NORMAL	Editar
JuniorTeste	Juniro Faria	123456	junior@email.com	NORMAL	Editar

Fontes: Elaborado pelo autor, 2024

Nesta tela do usuário consta a lista de usuários cadastrado e suas informações, cada usuário tem permissões diferente. A permissão Admin tem poder de efetuar edição e exclusão de outros usuários e a permissão Normal tem apenas o poder de edição.

Figura 6.2 Tela do Formulário do Usuário

Cadastro de Usuário

Login:

Nome:

Senha:

Email:

Permissão:
Selecione uma Permissão ▼

Fontes: Elaborado pelo autor, 2024

Nesta tela do usuário consta o formulário para criação de um novo usuário, cada cadastro permite inserir, login, nome, senha, email e qual permissão ele tem, tendo 3 opções disponíveis, admin, normal ou leitura. Ao clicar em cadastrar exibe-se uma mensagem de cadastro realizado com sucesso.

Figura 6.3 Conexão com o banco de dados

```
// Conexão com o banco de dados MySQL
$host = 'localhost:3306'; // Host do MySQL
$usuario = 'root'; // Usuário do MySQL
$senha = ''; // Senha do MySQL
$banco = 'crudproduto'; // Nome do banco de dados

// Conectando ao banco de dados
$conexao = new mysqli($host, $usuario, $senha, $banco);

// Verificando a conexão
if ($conexao->connect_error) {
    die("Erro na conexão: " . $conexao->connect_error);
}
```

Fontes: Elaborado pelo autor, 2024

No bloco de código da Figura 6.3 é responsável pela declaração das informações do banco de dados e sua conexão. No início declara-se variável contendo as informações de link de acesso, usuário, senha e nome do produto. Em seguida cria-se um comando de conexão e nela coloca-se as variáveis declarada. Por fim realizamos a conexão e caso de problema é informado para o usuário.

Figura 6.4 Coleta de dados e verificação

```

// Recebendo os dados do formulário
$login = $_POST['login'];
$nome = $_POST['nome'];
$senha = $_POST['senha'];
$email = $_POST['email'];
$permissao = $_POST['permissao'];

// Verifica se a permissão é válida
$permissoes_validas = array('ADMIN', 'NORMAL', 'LEITURA');
if (!in_array($permissao, $permissoes_validas)) {
    echo "Permissão inválida!";
    exit(); // Termina a execução do script se a permissão for inválida
}

```

Fontes: Elaborado pelo autor, 2024

No bloco de código da Figura 6.4 realiza-se a coleta de informações do formulário e inserindo-as em variáveis momentâneas para futura utilizações, posteriormente é feita uma verificação se a permissão for diferente de uma das três declarada, é emitido uma mensagem de permissão invalida.

Figura 6.5 – Código de inserção no banco de dados

```

// Inserindo os dados na tabela do banco de dados
$sql = "INSERT INTO usuario (login, nome, senha, email, permissao) VALUES ('$login', '$nome', '$senha', '$email', '$permissao')";

if ($conexao->query($sql) === TRUE) {
    echo "Usuário cadastrado com sucesso!";
} else {
    echo "Erro ao cadastrar usuário: " . $conexao->error;
}

// Fechando a conexão com o banco de dados
$conexao->close();

```

Fontes: Elaborado pelo autor, 2024

Neste bloco de código é responsável para criar o comando do banco de dados para inserção e posteriormente realizar a mesma e emitindo a mensagem de sucesso ou fracasso.

Figura 6.6 – Código de Busca do usuário para edição.

```

// Obtém o ID do usuário da URL
$login = $_GET['login'];

// Consulta SQL para obter os dados do usuário com o Login fornecido
$sql = "SELECT * FROM usuario WHERE login='$login'";
$resultado = $conexao->query($sql);

if ($resultado->num_rows > 0) {
    $usuarios = $resultado->fetch_assoc();
} else {
    echo "Usuário não encontrado";
}

```

Fontes: Elaborado pelo autor, 2024

Neste bloco de código é responsável pela coleta do login que deseja alterar, posteriormente realiza-se a criação do comando SQL para busca no banco e a executa, caso encontrado o usuário é salvo na variável ou emitido a mensagem de usuário não encontrado.

Figura 6.7 – Verificação de formulário de usuário.

```

29 // Verifica se o formulário foi submetido
30 if ($_SERVER["REQUEST_METHOD"] == "POST") {
31     $login = $_POST['login'];
32     $nome = $_POST['nome'];
33     $senha = $_POST['senha'];
34     $email = $_POST['email'];
35     $permissao = $_POST['permissao'];
36
37     // Atualiza os dados do usuário no banco de dados
38     $sql = "UPDATE usuario SET login='$login', nome='$nome', senha='$senha', email='$email', permissao='$permissao' WHERE login='$login'";
39     if ($conexao->query($sql) === TRUE) {
40         echo "Usuário atualizado com sucesso!";
41     } else {
42         echo "Erro ao atualizar usuário: " . $conexao->error;
43     }
44 }
45 header("Location: listar_usuarios.php");
46 exit();

```

Fontes: Elaborado pelo autor, 2024

Neste bloco de código tem por sua finalidade pegar os dados alterados no formulário, criar o comando SQL para atualização e executá-la, trazendo uma notificação de sucesso ou fracasso, por fim aplicação retornará a página de listagem de usuário.

Figura 6.8 Código de exclusão de usuário.

```

16 // Obtém o ID do usuário da URL
17 $login = $_GET['login'];
18
19 // Exclui o usuário do banco de dados
20 $sql = "DELETE FROM usuario WHERE login='$login'";
21 if ($conexao->query($sql) === TRUE) {
22     echo "Usuário excluído com sucesso!";
23 } else {
24     echo "Erro ao excluir usuário: " . $conexao->error;
25 }

```

Fontes: Elaborado pelo autor, 2024

No bloco de código da Figura 6.8 é responsável pela exclusão do usuário, através do login passado na variável inicial, cria-se o comando SQL para deleção e executa-o, por fim é emitido a mensagem de sucesso ou fracasso.

Figura 6.9 – Código Listagem de Usuário Parte 1

```

16 <?php
17 //Credenciais para conexão com o Banco
18 $dbhost = 'localhost:3306';
19 $dbuser = 'root';
20 $dbpass = '';
21 $dbname = 'crudproduto';
22
23 //Abre conexão com o MySQL
24 $conn = mysqli_connect($dbhost, $dbuser, $dbpass,$dbname);
25
26 if(!$conn ){
27     die('Falha ao conectar com o MySQL: ' . mysqli_connect_error());
28 }
29 //echo '<br>Conexão ao Banco realizada com Sucesso.';
30 $sql = 'SELECT * FROM Usuario';
31 $result = mysqli_query($conn, $sql); //A query seleciona as linhas da Tabela
32

```

Fontes: Elaborado pelo autor, 2024

No bloco de código da figura 6.9 é responsável declarar as variáveis correspondente ao banco de dados, no qual consta, o local do banco, usuário e senha de acesso e qual banco queremos selecionar, cria-se o comando SQL para busca e por fim executa a chamada e o retorno dela é inserido em uma variável.

Figura 6.10 – Código Listagem de Usuário Parte 2

```

34     if (mysqli_num_rows($result) > 0) {
35         echo '<div class="table-responsive">';
36         echo '<table class="table table-bordered table-hover table-sm">';
37         echo '<thead >';
38         echo '<tr style="background-color: #bee5eb;">';
39         echo '<th class="info">Login</th>';
40         echo '<th class="info">Nome</th>';
41         echo '<th class="info">Senha</th>';
42         echo '<th class="info">Email</th>';
43         echo '<th class="info">Permissão</th>';
44         echo '<th class="info"></th>';
45         echo '</tr>';
46         echo '</thead>';
47         echo '<tbody>';
48         while ($row = mysqli_fetch_assoc($result)){
49             echo '<tr>';
50             echo '<td>'.$row['login'].'</td>';
51             echo '<td>'.$row['nome'].'</td>';
52             echo '<td>'.$row['senha'].'</td>';
53             echo '<td>'.$row['email'].'</td>';
54             echo '<td>'.$row['permissao'].'</td>';
55             echo '<td><a href="telaEditarUsuario.php?login='.$row['login'].'" class="btn btn-info btn-sm">Editar</a>';
56             if ($row["permissao"] == "ADMIN" || $row["permissao"] == "NORMAL") {
57                 echo " | <a href='telaEditarUsuario.php?login=".$row["login"]."'>Editar</a>";
58             }
59             if ($row["permissao"] == "ADMIN") {
60                 echo " | <a href='excluirUsuario.php?login=".$row["login"]."'>Excluir</a>";
61             }
62             echo '</td>';
63             echo '</tr>';
64         }
65     }
66     echo '</tbody>';
67     echo '</table>';
68     echo '</div>';
69 }else {
70     echo "Nenhum Usuario Encontrado.";
71 }

```

Fontes: Elaborado pelo autor, 2024

A Figura 6.10 é continuação do código da Figura 6.9, após realizar todo procedimento de busca no banco de dados, cria-se a estrutura da tabela com suas respectivas colunas login, nome, senha, email e permissão, posteriormente é realizado a população da tabela com os dados coletados. Os botões de editar e Exclusão irá aparecer apenas para usuários admin e o apenas o editar aparecerá para o normal. Por fim caso não encontre usuário emite uma mensagem de aviso.

6.2 Conclusão

O tema abordado no qual uma aplicação web para gerenciamento de Usuários, foi uma experiencia enriquecedora. Por meio deste projeto pude aplicar os conhecimentos adquiridos na respectiva matéria. Enfrentando um desafio real e desenvolver soluções eficientes.

Esta aplicação web não é apenas agenda de anotações de viagens, mas também representa um marco em minha jornada de conhecimento. Consolidando todo conhecimento, técnicas em um lugar só.