

Projeto de desenvolvimento em Back-End

Grupo 10

João Pedro Dos santos Nascimento

Professor orientador: Charles Bastos

CG – Noite

Rio de janeiro, RJ

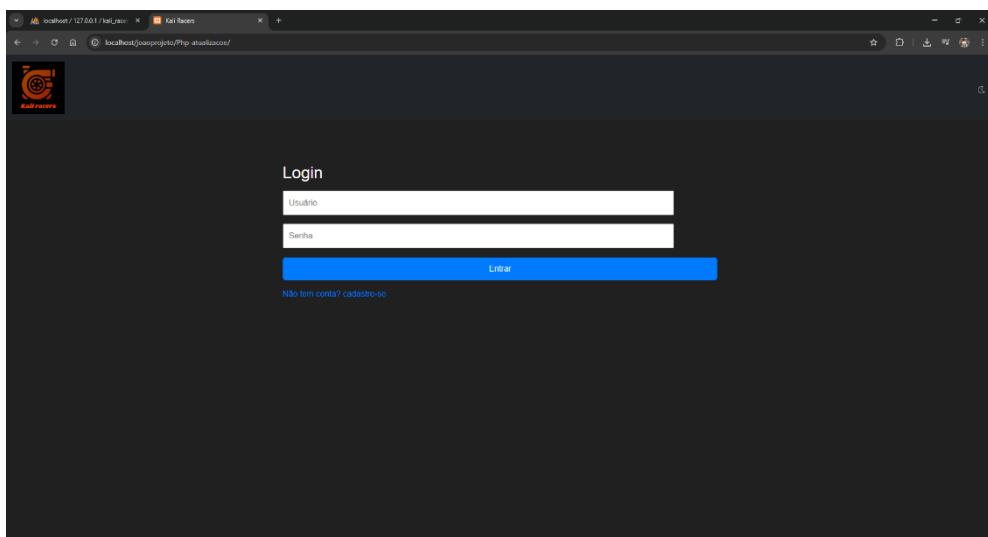
11/2025

Repositório GitHub: <https://github.com/Cz24-Type3/Projeto-back-end>



Kali Racers “Mais do que performance, é uma paixão”

1- Login/ Cadastro

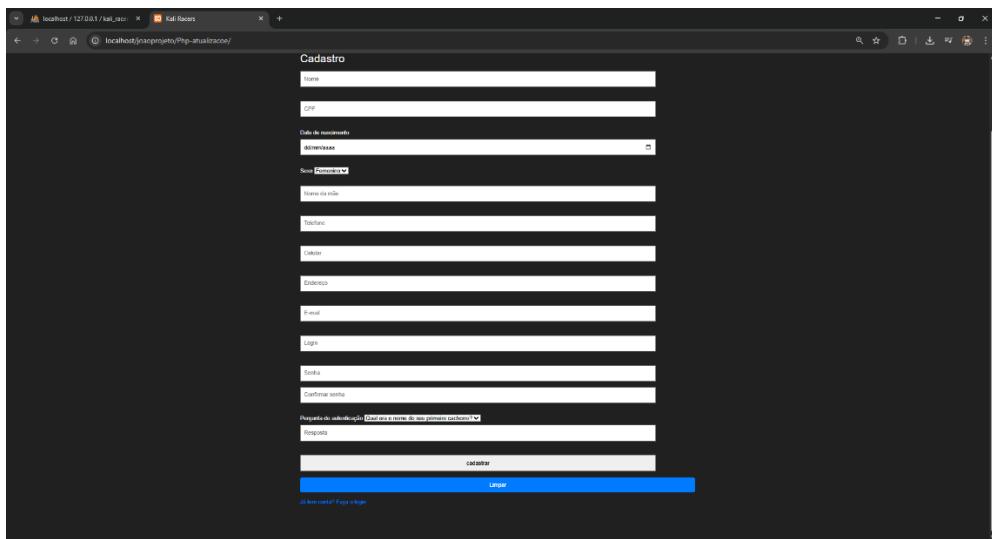


Para o login foi utilizado um método simples de formulário:

```
<!--FORM DE LOGIN-->

<div class="container">
    <form id="login-form" class="active" action="login.php" method="post">
        <h2>Login</h2>
        <input type="text" name="ulogin" placeholder="Usuário" required>
        <input type="password" name="senha" placeholder="Senha" required>
        <button type="submit">Entrar</button>
        <p class="toggle" onclick="mostrarCadastro()">Não tem conta? cadastre-se</p>
    </form>
```

Print: index.html



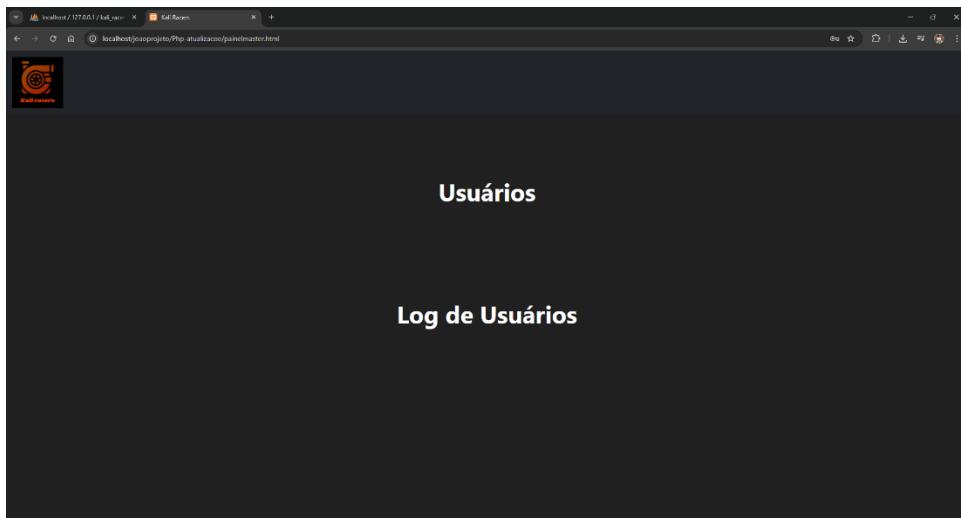
Print: index.html

1.1- Caso master

```
//Verifica se é master
if ($login == 'admin') {
    header("Location: painelmaster.html");
    exit;
```

Print: login.php

Caso o usuário seja um usuário master será redirecionado para uma pagina exclusiva que lhe dará poder de gerenciamento de usuários.



Print: painelmaster.html

Usuários

1	Silva Pedro	150270000700	2001-06-11	masculino	Ana Lucia	81080277	2110112233886	Rua jornalista Argus Alho	palo@gmail.com	spao	
2	Jorge	12350467500	2001-06-25	outro	Ale		2150457050	Rua 50, casa 01	jorge@gmail.com	jorge	
3	kali	01110164000	1998-07-14	feminino	Ane		2191408740801	Rua 2, casa 09	kali@gmail.com	kali	
4	renan santos	12312642777	2001-06-21	masculino	Renata	21977645392	21977645392	Rua visconde de araguá	renanpc01@gmail.com	renannik	
5	admin	0000000000	2001-01-01	outro	0	0002500000	a		admine@cdm.com	admin	
6	kg	1510156795	2004-08-07	masculino	Ingrid nemet		21977163118	becejabas	usu.renan.guilherme@1	kg71	

[Voltar](#)

Print: usermaster.html

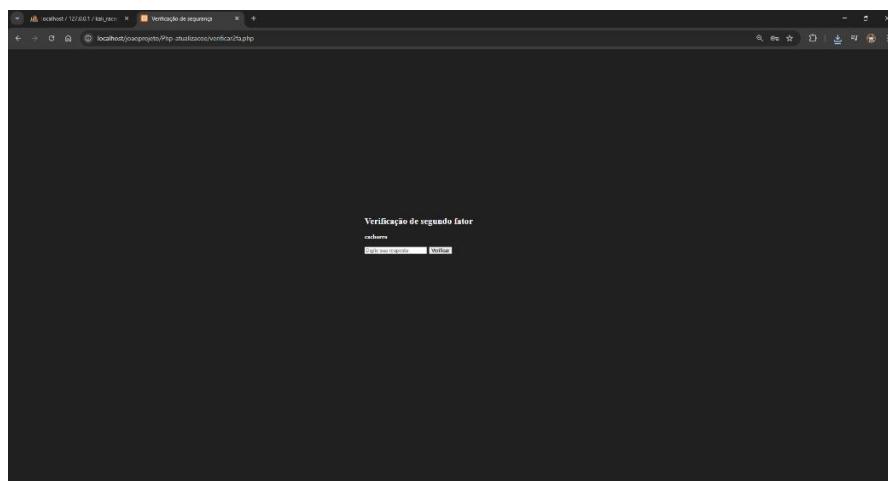
Logs do Sistema

ID	ID do usuário	Ação	Detalhes	Data/Hora
1	3	login	-	2025-11-22 22:47:44
2	3	logout	-	2025-11-22 22:55:39
3	3	login	-	2025-11-22 22:55:58
4	3	logout	-	2025-11-22 22:58:41
5	3	login	-	2025-11-22 23:27:29
6	3	logout	-	2025-11-22 23:27:54

[Voltar](#) [Exportar PDF](#)

Print: log.html

2- Verificação de 2 fatores



Print: autentica2fa.php

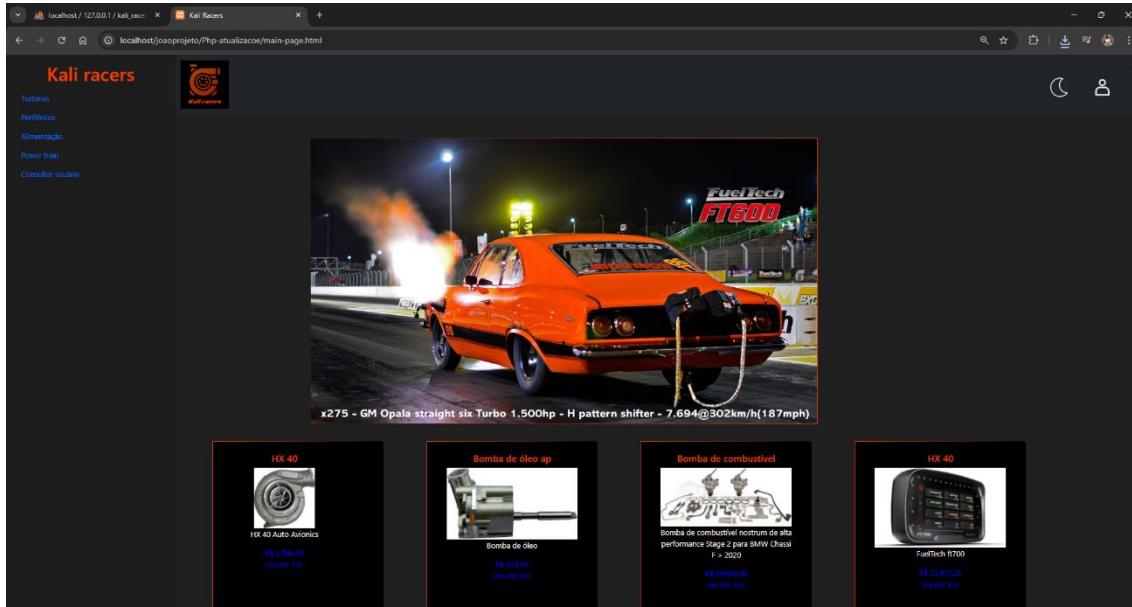
Nele é utilizado o php para verificar se as informações salvas no sql são iguais.

```
$sql = "SELECT resposta FROM usuarios WHERE id = '$id'";
$result = mysqli_query($conexao, $sql);
$usuario = mysqli_fetch_assoc($result);

if (password_verify($fa2r, $usuario['resposta'])) {
    //Autenticação completa!
    $_SESSION['id'] = $id;
    unset($_SESSION['temp_user_id']);
    header("Location: main-page.html");
    exit;
} else {
    echo "Resposta incorreta. Tente novamente.";
}
?>
```

Print: autentica2fa.php

3- Pagina inicial



print: main-page.html

```
<!--Menu lateral desktop-->


<div>
    <h1>Kali racers</h1>
    <ul>
        <li><a href="Turbinas.html">Turbinas</a></li>
        <li><a href="Perifericos.html">Periféricos</a></li>
        <li><a href="Alimentacao.html">Alimentação</a></li>
        <li><a href="#">Power train</a></li>
        <li><a href="#">Consultar usuário</a></li>
    </ul>
</div>


```

<!--Menu principal-->

```
<div>
    <nav>
        <div>
            <!--Botão do menu lateral (para o mobile)-->
            <button>...</button>
            <!--Logo do site com o link para voltar para o inicio-->
            <a href="main-page.html"></a>
            <div>...</div>
            <!--Carinho-->
            <!--Dark Mode-->
            <button id="darkmode" type="button" class="btn" data-bs-toggle="button"></button>
            <!--Login e cadastro-->
            <button id="usuario" type="button" class="btn" data-bs-toggle="button"></button>
        </div>
    </nav>

```

<!--Imagem foco-->

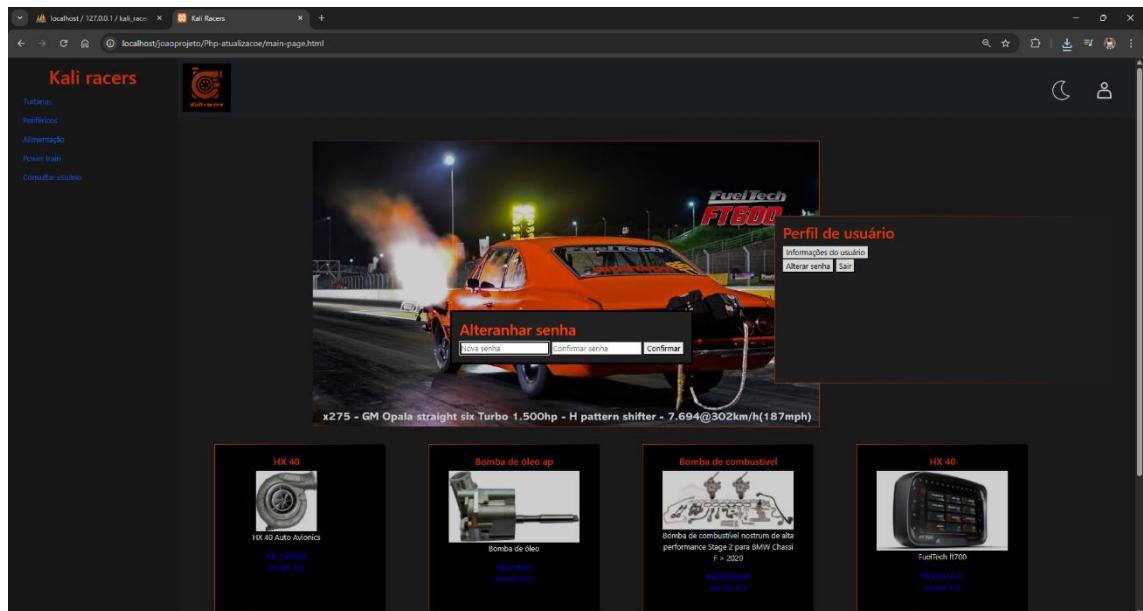
```
<div>
    
</div>
```

<!--Conteúdo principal-->

```
<div>
    <!--Row = Linhas de conteúdos-->
    <div class="row">
        <div>...</div>
    </div>
</div>
```

Print: main-page.html

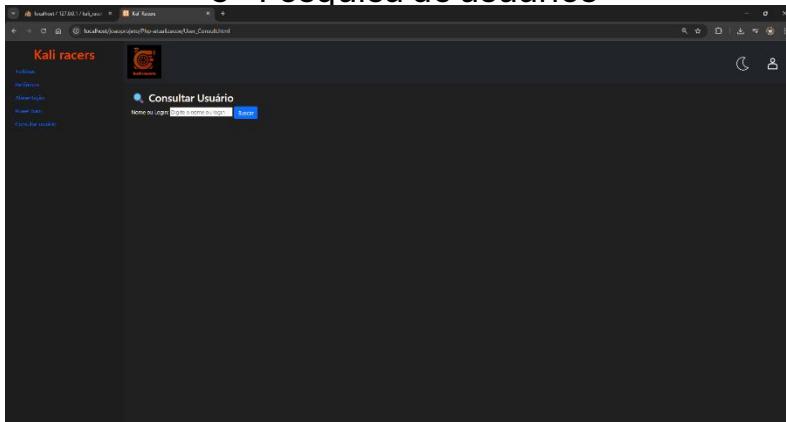
4- Alterar senha



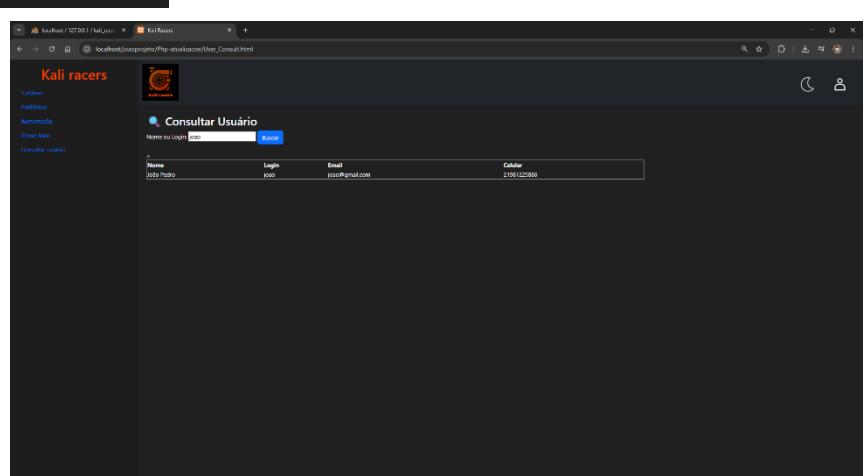
Print: main-page.html (modal do usuário)

A alteração de senha ser feita apenas pela parte do usuário foi uma escolha de segurança, para evitar vazamentos e possíveis adversidades envolvendo master x usuário.

5- Pesquisa de usuários



print: User_Consult.html



6- Banco de dados

Para a criação do banco de dados foi utilizada a ferramenta xampp, com ela conseguimos por meio do sql e o phpmyadmin gerenciar o banco de dados, com ele temos um perfeito gerenciamento de usuários com as seguintes principais tabelas: usuarios e log_usuarios

6.1- Tabela de usuários

A tabela de usuários foi pensada de forma que tenhamos informações suficientes para entregar total suporte e agilidade no gerenciamento de infraestrutura e pessoal dos usuários

Código SQL da criação da tabela usuarios:

```
CREATE TABLE usuarios (
    id INT AUTO_INCREMENT PRIMARY KEY,
    nome VARCHAR(150) NOT NULL,
    cpf CHAR(11) NOT NULL UNIQUE,
    nascimento DATE NOT NULL,
    sexo ENUM('masculino', 'feminino', 'outro') NOT NULL,
    mae VARCHAR(150) NOT NULL,
    telefone VARCHAR(15),
    celular VARCHAR(15) NOT NULL,
    endereco VARCHAR(255) NOT NULL,
    email VARCHAR(150) NOT NULL UNIQUE,
    ulogin VARCHAR(50) NOT NULL UNIQUE,
    senha VARCHAR(255) NOT NULL, -- vai armazenar o hash (password_hash
gera até 255 caracteres)
    pergunta ENUM('cachorro', 'cor', 'musica') NOT NULL,
    resposta VARCHAR(255) NOT NULL,
    criado_em TIMESTAMP DEFAULT CURRENT_TIMESTAMP
);
```

Um detalhe interessante desta tabela foi o armazenamento da senha, utilizando o método hash que antes não era conhecido pela equipe.

```
$senha = $_POST['senha'];
$hash = password_hash($senha, PASSWORD_BCRYPT); //transforma a senha em um hash
$senha_conf = $_POST['senha_conf'];
//Se $senha != $senha_conf // Pergunta de autenticação da segunda senha
Print: cadastrp.php (área de cadastro da senha)
```

Utilizando este método de salvar o hash da senha do usuário, que seria uma forma criptografada da senha do usuário) temos mais segurança com os nossos usuários.

6.2- Tabela de Log

Esta tabela ficou responsável por armazenar informações sobre entrada e saída de usuários, como a hora de login e logout. Foi

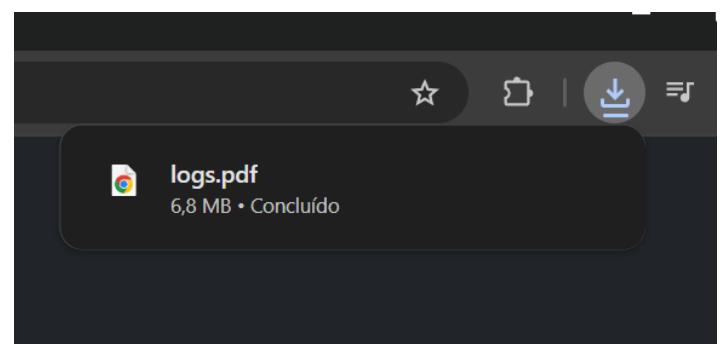
log_id	id	action	datafile	ip_address	created_at
1	3	login	NULL	NULL	2025-11-22 22:47:44
2	3	logout	NULL	NULL	2025-11-22 22:55:59
3	3	login	NULL	NULL	2025-11-22 22:55:58
4	3	logout	NULL	NULL	2025-11-22 22:58:41
5	3	login	NULL	NULL	2025-11-22 23:27:29
6	3	logout	NULL	NULL	2025-11-22 23:27:34
7	3	login	NULL	NULL	2025-11-23 01:55:52
8	3	logout	NULL	NULL	2025-11-23 01:58:29
9	3	login	NULL	NULL	2025-11-23 01:58:49
10	3	logout	NULL	NULL	2025-11-23 01:59:01

Print: Tabela de log

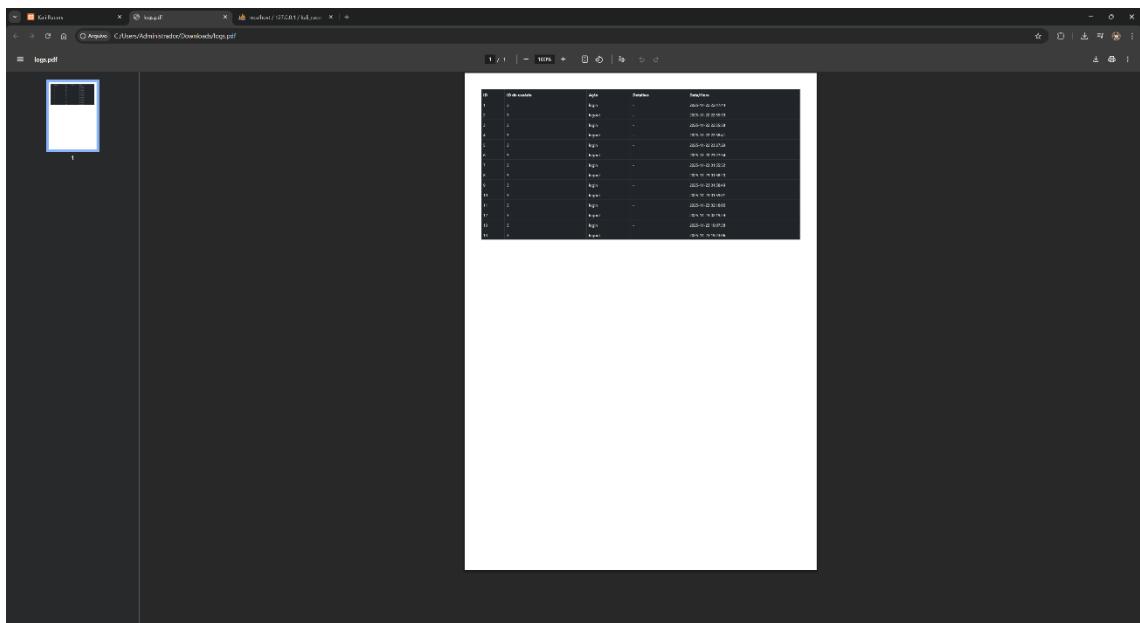
Outra aplicação utilizando o log é a de gerar um PDF com o registro de logs

10	3
11	3
12	3
13	3
14	3

Print: Log.html (botão Exportar PDF)



print: ao clickar no Exportar PDF



Print: PDF gerado ao clickar no botão

Códigos de criação do banco de dados e tabelas

Tabela usuários:

```
CREATE TABLE usuarios (
    id INT AUTO_INCREMENT PRIMARY KEY,
    nome VARCHAR(150) NOT NULL,
    cpf CHAR(11) NOT NULL UNIQUE,
    nascimento DATE NOT NULL,
    sexo ENUM('masculino', 'feminino', 'outro') NOT NULL,
    mae VARCHAR(150) NOT NULL,
    telefone VARCHAR(15),
    celular VARCHAR(15) NOT NULL,
    endereco VARCHAR(255) NOT NULL,
    email VARCHAR(150) NOT NULL UNIQUE,
    ulogin VARCHAR(50) NOT NULL UNIQUE,
    senha VARCHAR(255) NOT NULL, -- vai armazenar o hash
    (password_hash gera até 255 caracteres)
    pergunta ENUM('cachorro', 'cor', 'musica') NOT NULL,
    resposta VARCHAR(255) NOT NULL,
    criado_em TIMESTAMP DEFAULT CURRENT_TIMESTAMP
);
```

Tabela log_usuarios

```
CREATE TABLE log_usuarios (
    logid INT AUTO_INCREMENT PRIMARY KEY,
    id INT NOT NULL,
    action VARCHAR(50) NOT NULL,
    details TEXT NULL,
    ip_address VARCHAR(50),
    created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
    FOREIGN KEY (id) REFERENCES usuarios(id)
);
```