* **Dia 4: Lógica do Quiz (Conexão com DB e Pontuação) e Sessão Admin**
  + Refinar a lógica PHP para verificar respostas corretas, atualizar a pontuação no banco de dados.
  + Criar a interface e a lógica para o painel administrativo (adicionar/editar/excluir perguntas).
  + Implementar a validação de sessão para o administrador.

**Dia 4: Lógica do Quiz (Conexão com DB e Pontuação) e Sessão Admin**

Aqui está o que vamos implementar hoje:

1. **Refinar a lógica PHP para verificar respostas corretas, atualizar a pontuação no banco de dados.** Isso significa que a pontuação dos usuários não será mais apenas temporária na sessão, mas persistente.
2. **Criar a interface e a lógica para o painel administrativo (adicionar/editar/excluir perguntas).**
3. **Implementar a validação de sessão para o administrador.**

Vamos começar por ajustar a atualização da pontuação e, em seguida, mergulhamos no painel de administração.

**1. Atualizar a Lógica de Pontuação no quiz\_game.php**

Para que a pontuação do usuário seja guardada no banco de dados, precisamos modificar o quiz\_game.php.

**O que vamos mudar:**

* Quando uma resposta estiver correta, além de adicionar à pontuação da sessão, vamos **atualizar a coluna pontuacao\_total** do usuário na tabela usuarios.

Abre o ficheiro **quiz\_game.php** no teu VS Code e localiza a parte onde a pontuação é adicionada (dentro do if ($id\_resposta\_selecionada == $resposta\_correta\_db['id\_resposta'])).

**Substitui o bloco if ($resposta\_correta\_db) inteiro pelo seguinte código atualizado:**

PHP

if ($resposta\_correta\_db) {

if ($id\_resposta\_selecionada == $resposta\_correta\_db['id\_resposta']) {

$\_SESSION['pontuacao'] += 10; // Adiciona 10 pontos na sessão

// --- NOVO: Atualizar a pontuação no banco de dados ---

$id\_usuario\_atual = $\_SESSION['id\_usuario'];

$nova\_pontuacao\_total = $\_SESSION['pontuacao']; // Assumimos que pontuação na sessão já é a total acumulada

$stmt\_update\_score = $conn->prepare("UPDATE usuarios SET pontuacao\_total = ? WHERE id\_usuario = ?");

$stmt\_update\_score->bind\_param("ii", $nova\_pontuacao\_total, $id\_usuario\_atual);

$stmt\_update\_score->execute();

$stmt\_update\_score->close();

// --- FIM NOVO ---

$mensagem\_feedback = "<p class='feedback correta'>Correto! Pontuação: " . $\_SESSION['pontuacao'] . "</p>";

} else {

// Se incorreto, podemos também mostrar qual era a resposta certa

$stmt\_texto\_correto = $conn->prepare("SELECT texto\_resposta FROM respostas WHERE id\_resposta = ? LIMIT 1");

$stmt\_texto\_correto->bind\_param("i", $resposta\_correta\_db['id\_resposta']);

$stmt\_texto\_correto->execute();

$result\_texto\_correto = $stmt\_texto\_correto->get\_result();

$texto\_resposta\_correta = $result\_texto\_correto->fetch\_assoc()['texto\_resposta'];

$stmt\_texto\_correto->close();

$mensagem\_feedback = "<p class='feedback incorreta'>Incorreto. A resposta correta era: \*\*" . htmlspecialchars($texto\_resposta\_correta) . "\*\*.</p>";

$mensagem\_feedback .= "<p class='feedback incorreta'>Pontuação: " . $\_SESSION['pontuacao'] . "</p>";

}

} else {

$mensagem\_feedback = "<p class='feedback incorreta'>Erro ao verificar a resposta. Pontuação: " . $\_SESSION['pontuacao'] . "</p>";

}

$stmt->close();

**2. Criar o Painel Administrativo e Lógica Admin**

Para o painel administrativo, vamos precisar de:

* Uma página de login separada para administradores (ou um campo extra no login existente). Para simplificar e cumprir o prazo, vamos assumir um **administrador com id\_usuario = 1** por enquanto, e redirecionaremos para um admin\_dashboard.php se esse ID estiver logado.
* Uma nova página: **admin\_dashboard.php** (com listagem, adição, edição, exclusão de perguntas).
* Uma nova página: **gerir\_perguntas.php** (para o formulário de adicionar/editar).

**Primeiro, vamos ajustar o login.php para identificar um admin (o usuário com id\_usuario = 1):**

Abre login.php. Localiza a linha header("Location: dashboard.php"); (logo após a senha estar correta).

**Substitui-a pelo seguinte código:**

PHP

// Senha correta, iniciar sessão

$\_SESSION['id\_usuario'] = $id\_usuario;

$\_SESSION['nome\_usuario'] = $nome\_usuario;

// --- NOVO: Redirecionar para painel admin se for o utilizador com ID 1 ---

if ($id\_usuario == 1) { // Supondo que o usuário com ID 1 é o administrador

$\_SESSION['is\_admin'] = true;

header("Location: admin\_dashboard.php");

} else {

$\_SESSION['is\_admin'] = false; // Define como não-admin para usuários comuns

header("Location: dashboard.php");

}

exit();

// --- FIM NOVO ---

**Agora, vamos criar os ficheiros do painel de administração:**

**2.1. admin\_dashboard.php**

Cria um novo ficheiro chamado **admin\_dashboard.php** na tua pasta quiz\_angola e cola o seguinte código. Ele listará as perguntas existentes e terá links para adicionar/editar.

PHP

<?php

session\_start();

require\_once 'db\_connect.php';

// --- Proteção do Painel Administrativo ---

if (!isset($\_SESSION['id\_usuario']) || !isset($\_SESSION['is\_admin']) || !$\_SESSION['is\_admin']) {

header("Location: login.php"); // Redireciona se não estiver logado ou não for admin

exit();

}

// --- Fim Proteção ---

$mensagem = '';

if (isset($\_GET['msg'])) {

$mensagem = htmlspecialchars($\_GET['msg']);

}

// Lógica para excluir pergunta

if ($\_SERVER['REQUEST\_METHOD'] == 'POST' && isset($\_POST['action']) && $\_POST['action'] == 'delete') {

$id\_pergunta\_delete = (int)$\_POST['id\_pergunta'];

// Para garantir que as respostas também são excluídas devido à FOREIGN KEY com ON DELETE CASCADE

$stmt = $conn->prepare("DELETE FROM perguntas WHERE id\_pergunta = ?");

$stmt->bind\_param("i", $id\_pergunta\_delete);

if ($stmt->execute()) {

header("Location: admin\_dashboard.php?msg=" . urlencode("Pergunta excluída com sucesso!"));

exit();

} else {

$mensagem = "<p class='error'>Erro ao excluir pergunta: " . $stmt->error . "</p>";

}

$stmt->close();

}

// Buscar todas as perguntas para exibir

$sql = "SELECT id\_pergunta, texto\_pergunta, categoria, nivel\_dificuldade FROM perguntas ORDER BY categoria, id\_pergunta DESC";

$result = $conn->query($sql);

$perguntas = [];

if ($result && $result->num\_rows > 0) {

while($row = $result->fetch\_assoc()) {

$perguntas[] = $row;

}

}

$conn->close();

?>

<!DOCTYPE html>

<html lang="pt">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Painel Admin - Quiz Conheça Angola</title>

<style>

body {

font-family: 'Arial', sans-serif;

background: linear-gradient(to bottom, #EF0000 50%, #000000 50%);

color: #fff;

display: flex;

justify-content: center;

align-items: center;

min-height: 100vh;

margin: 0;

text-align: center;

}

.container {

background-color: rgba(255, 255, 255, 0.95);

padding: 30px;

border-radius: 10px;

box-shadow: 0 5px 20px rgba(0, 0, 0, 0.3);

width: 90%;

max-width: 900px;

color: #333;

text-align: left;

}

h1 {

font-size: 2.2em;

margin-bottom: 25px;

color: #EF0000;

text-align: center;

}

.message {

margin-bottom: 15px;

padding: 10px;

border-radius: 5px;

text-align: center;

font-weight: bold;

}

.success {

background-color: #d4edda;

color: #155724;

border: 1px solid #c3e6cb;

}

.error {

background-color: #f8d7da;

color: #721c24;

border: 1px solid #f5c6cb;

}

.admin-actions {

text-align: center;

margin-bottom: 30px;

}

.btn {

display: inline-block;

padding: 10px 20px;

border-radius: 5px;

text-decoration: none;

font-weight: bold;

font-size: 1em;

transition: background-color 0.3s ease;

box-shadow: 0 2px 5px rgba(0, 0, 0, 0.1);

cursor: pointer;

margin: 5px;

}

.btn-add {

background-color: #28a745; /\* Verde \*/

color: white;

border: 1px solid #218838;

}

.btn-add:hover {

background-color: #218838;

}

.btn-logout {

background-color: #000000; /\* Preto \*/

color: white;

border: 1px solid #333333;

}

.btn-logout:hover {

background-color: #333333;

}

table {

width: 100%;

border-collapse: collapse;

margin-top: 20px;

}

th, td {

border: 1px solid #ddd;

padding: 10px;

text-align: left;

color: #333;

}

th {

background-color: #f2f2f2;

color: #555;

font-weight: bold;

}

.table-actions {

display: flex;

gap: 5px;

justify-content: flex-start;

}

.btn-edit, .btn-delete {

padding: 8px 12px;

font-size: 0.9em;

text-align: center;

white-space: nowrap;

}

.btn-edit {

background-color: #007bff;

color: white;

border: 1px solid #0056b3;

}

.btn-edit:hover {

background-color: #0056b3;

}

.btn-delete {

background-color: #dc3545;

color: white;

border: 1px solid #c82333;

}

.btn-delete:hover {

background-color: #c82333;

}

form.delete-form {

display: inline; /\* Para alinhar o botão de exclusão \*/

}

</style>

</head>

<body>

<div class="container">

<h1>Painel Administrativo</h1>

<?php if ($mensagem): ?>

<p class="message <?php echo strpos($mensagem, 'sucesso') !== false ? 'success' : 'error'; ?>">

<?php echo $mensagem; ?>

</p>

<?php endif; ?>

<div class="admin-actions">

<a href="gerir\_perguntas.php" class="btn btn-add">Adicionar Nova Pergunta</a>

<a href="logout.php" class="btn btn-logout">Sair do Admin</a>

</div>

<?php if (!empty($perguntas)): ?>

<table>

<thead>

<tr>

<th>ID</th>

<th>Pergunta</th>

<th>Categoria</th>

<th>Dificuldade</th>

<th>Ações</th>

</tr>

</thead>

<tbody>

<?php foreach ($perguntas as $pergunta): ?>

<tr>

<td><?php echo $pergunta['id\_pergunta']; ?></td>

<td><?php echo htmlspecialchars($pergunta['texto\_pergunta']); ?></td>

<td><?php echo htmlspecialchars($pergunta['categoria']); ?></td>

<td><?php echo htmlspecialchars($pergunta['nivel\_dificuldade']); ?></td>

<td class="table-actions">

<a href="gerir\_perguntas.php?id=<?php echo $pergunta['id\_pergunta']; ?>" class="btn btn-edit">Editar</a>

<form action="admin\_dashboard.php" method="POST" class="delete-form" onsubmit="return confirm('Tem certeza que deseja excluir esta pergunta e suas respostas?');">

<input type="hidden" name="action" value="delete">

<input type="hidden" name="id\_pergunta" value="<?php echo $pergunta['id\_pergunta']; ?>">

<button type="submit" class="btn btn-delete">Excluir</button>

</form>

</td>

</tr>

<?php endforeach; ?>

</tbody>

</table>

<?php else: ?>

<p style="text-align: center; color: #555;">Nenhuma pergunta encontrada. Adicione a sua primeira pergunta!</p>

<?php endif; ?>

</div>

</body>

</html>

**2.2. gerir\_perguntas.php (Adicionar/Editar Perguntas)**

Cria um novo ficheiro chamado **gerir\_perguntas.php** na tua pasta quiz\_angola e cola o seguinte código. Esta página permitirá adicionar novas perguntas ou editar as existentes, incluindo as respostas.

PHP

<?php

session\_start();

require\_once 'db\_connect.php';

// --- Proteção do Painel Administrativo ---

if (!isset($\_SESSION['id\_usuario']) || !isset($\_SESSION['is\_admin']) || !$\_SESSION['is\_admin']) {

header("Location: login.php");

exit();

}

// --- Fim Proteção ---

$pergunta\_id = isset($\_GET['id']) ? (int)$\_GET['id'] : 0;

$pergunta\_data = ['texto\_pergunta' => '', 'categoria' => '', 'nivel\_dificuldade' => 'Facil'];

$respostas\_data = [['texto\_resposta' => '', 'correta' => 0], ['texto\_resposta' => '', 'correta' => 0], ['texto\_resposta' => '', 'correta' => 0], ['texto\_resposta' => '', 'correta' => 0]];

$page\_title = "Adicionar Nova Pergunta";

$mensagem = '';

// Carregar dados da pergunta e respostas se for uma edição

if ($pergunta\_id > 0) {

$page\_title = "Editar Pergunta";

$stmt\_p = $conn->prepare("SELECT texto\_pergunta, categoria, nivel\_dificuldade FROM perguntas WHERE id\_pergunta = ?");

$stmt\_p->bind\_param("i", $pergunta\_id);

$stmt\_p->execute();

$result\_p = $stmt\_p->get\_result();

if ($result\_p->num\_rows > 0) {

$pergunta\_data = $result\_p->fetch\_assoc();

} else {

$mensagem = "<p class='error'>Pergunta não encontrada.</p>";

$pergunta\_id = 0; // Reset para modo de adição

}

$stmt\_p->close();

$stmt\_r = $conn->prepare("SELECT id\_resposta, texto\_resposta, correta FROM respostas WHERE id\_pergunta = ? ORDER BY id\_resposta ASC");

$stmt\_r->bind\_param("i", $pergunta\_id);

$stmt\_r->execute();

$result\_r = $stmt\_r->get\_result();

$respostas\_carregadas = [];

while($row\_r = $result\_r->fetch\_assoc()) {

$respostas\_carregadas[] = $row\_r;

}

// Preenche as respostas carregadas, garantindo 4 campos de resposta

for($i = 0; $i < 4; $i++) {

if (isset($respostas\_carregadas[$i])) {

$respostas\_data[$i] = $respostas\_carregadas[$i];

} else {

$respostas\_data[$i] = ['texto\_resposta' => '', 'correta' => 0];

}

}

$stmt\_r->close();

}

// Processar formulário de submissão

if ($\_SERVER['REQUEST\_METHOD'] == 'POST') {

$texto\_pergunta = $\_POST['texto\_pergunta'];

$categoria = $\_POST['categoria'];

$nivel\_dificuldade = $\_POST['nivel\_dificuldade'];

$respostas\_submetidas = $\_POST['respostas']; // Array de textos de resposta

$resposta\_correta\_indice = (int)$\_POST['correta\_indice']; // Índice da resposta correta (0 a 3)

// Validação básica

if (empty($texto\_pergunta) || empty($categoria) || empty($nivel\_dificuldade) || empty($respostas\_submetidas)) {

$mensagem = "<p class='error'>Por favor, preencha todos os campos da pergunta e respostas.</p>";

} elseif ($resposta\_correta\_indice < 0 || $resposta\_correta\_indice > 3 || !isset($respostas\_submetidas[$resposta\_correta\_indice])) {

$mensagem = "<p class='error'>Selecione uma resposta correta válida.</p>";

} else {

$conn->begin\_transaction(); // Inicia transação para garantir integridade

try {

if ($pergunta\_id == 0) { // Adicionar nova pergunta

$stmt\_p = $conn->prepare("INSERT INTO perguntas (texto\_pergunta, categoria, nivel\_dificuldade) VALUES (?, ?, ?)");

$stmt\_p->bind\_param("sss", $texto\_pergunta, $categoria, $nivel\_dificuldade);

$stmt\_p->execute();

$nova\_pergunta\_id = $conn->insert\_id;

$stmt\_p->close();

} else { // Editar pergunta existente

$stmt\_p = $conn->prepare("UPDATE perguntas SET texto\_pergunta = ?, categoria = ?, nivel\_dificuldade = ? WHERE id\_pergunta = ?");

$stmt\_p->bind\_param("sssi", $texto\_pergunta, $categoria, $nivel\_dificuldade, $pergunta\_id);

$stmt\_p->execute();

$stmt\_p->close();

// Excluir respostas antigas para reinserir (simplifica a lógica de atualização)

$stmt\_delete\_r = $conn->prepare("DELETE FROM respostas WHERE id\_pergunta = ?");

$stmt\_delete\_r->bind\_param("i", $pergunta\_id);

$stmt\_delete\_r->execute();

$stmt\_delete\_r->close();

$nova\_pergunta\_id = $pergunta\_id;

}

// Inserir/Atualizar respostas

$stmt\_r = $conn->prepare("INSERT INTO respostas (id\_pergunta, texto\_resposta, correta) VALUES (?, ?, ?)");

for ($i = 0; $i < 4; $i++) {

$is\_correta = ($i == $resposta\_correta\_indice) ? 1 : 0;

$stmt\_r->bind\_param("isi", $nova\_pergunta\_id, $respostas\_submetidas[$i], $is\_correta);

$stmt\_r->execute();

}

$stmt\_r->close();

$conn->commit(); // Confirma a transação

header("Location: admin\_dashboard.php?msg=" . urlencode("Pergunta salva com sucesso!"));

exit();

} catch (mysqli\_sql\_exception $e) {

$conn->rollback(); // Reverte em caso de erro

$mensagem = "<p class='error'>Erro ao salvar pergunta: " . $e->getMessage() . "</p>";

}

}

}

$conn->close();

?>

<!DOCTYPE html>

<html lang="pt">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title><?php echo $page\_title; ?> - Quiz Conheça Angola</title>

<style>

body {

font-family: 'Arial', sans-serif;

background: linear-gradient(to bottom, #EF0000 50%, #000000 50%);

color: #fff;

display: flex;

justify-content: center;

align-items: center;

min-height: 100vh;

margin: 0;

text-align: center;

}

.container {

background-color: rgba(255, 255, 255, 0.95);

padding: 30px;

border-radius: 10px;

box-shadow: 0 5px 20px rgba(0, 0, 0, 0.3);

width: 90%;

max-width: 700px;

color: #333;

text-align: left;

}

h1 {

font-size: 2.2em;

margin-bottom: 25px;

color: #EF0000;

text-align: center;

}

.message {

margin-bottom: 15px;

padding: 10px;

border-radius: 5px;

text-align: center;

font-weight: bold;

}

.success {

background-color: #d4edda;

color: #155724;

border: 1px solid #c3e6cb;

}

.error {

background-color: #f8d7da;

color: #721c24;

border: 1px solid #f5c6cb;

}

.form-group {

margin-bottom: 18px;

}

.form-group label {

display: block;

margin-bottom: 8px;

font-weight: bold;

color: #555;

}

.form-group input[type="text"],

.form-group textarea,

.form-group select {

width: calc(100% - 20px);

padding: 12px;

border: 1px solid #ddd;

border-radius: 5px;

font-size: 1em;

}

textarea {

resize: vertical;

min-height: 80px;

}

.radio-group {

display: flex;

align-items: center;

gap: 10px;

margin-top: 5px;

}

.radio-group input[type="radio"] {

margin-right: 5px;

}

.btn {

display: inline-block;

padding: 12px 25px;

border-radius: 5px;

text-decoration: none;

font-weight: bold;

font-size: 1em;

transition: background-color 0.3s ease;

box-shadow: 0 2px 5px rgba(0, 0, 0, 0.1);

cursor: pointer;

margin-top: 20px;

}

.btn-submit {

background-color: #28a745;

color: white;

border: 1px solid #218838;

}

.btn-submit:hover {

background-color: #218838;

}

.btn-back {

background-color: #6c757d;

color: white;

border: 1px solid #5a6268;

margin-left: 10px;

}

.btn-back:hover {

background-color: #5a6268;

}

</style>

</head>

<body>

<div class="container">

<h1><?php echo $page\_title; ?></h1>

<?php if ($mensagem): ?>

<p class="message <?php echo strpos($mensagem, 'sucesso') !== false ? 'success' : 'error'; ?>">

<?php echo $mensagem; ?>

</p>

<?php endif; ?>

<form action="gerir\_perguntas.php<?php echo $pergunta\_id > 0 ? '?id=' . $pergunta\_id : ''; ?>" method="POST">

<div class="form-group">

<label for="texto\_pergunta">Texto da Pergunta:</label>

<textarea id="texto\_pergunta" name="texto\_pergunta" required><?php echo htmlspecialchars($pergunta\_data['texto\_pergunta']); ?></textarea>

</div>

<div class="form-group">

<label for="categoria">Categoria:</label>

<select id="categoria" name="categoria" required>

<option value="">Selecione uma categoria</option>

<option value="Historia" <?php echo ($pergunta\_data['categoria'] == 'Historia') ? 'selected' : ''; ?>>História</option>

<option value="Geografia" <?php echo ($pergunta\_data['categoria'] == 'Geografia') ? 'selected' : ''; ?>>Geografia</option>

<option value="Cultura" <?php echo ($pergunta\_data['categoria'] == 'Cultura') ? 'selected' : ''; ?>>Cultura</option>

<option value="Curiosidades" <?php echo ($pergunta\_data['categoria'] == 'Curiosidades') ? 'selected' : ''; ?>>Curiosidades</option>

</select>

</div>

<div class="form-group">

<label for="nivel\_dificuldade">Nível de Dificuldade:</label>

<select id="nivel\_dificuldade" name="nivel\_dificuldade" required>

<option value="Facil" <?php echo ($pergunta\_data['nivel\_dificuldade'] == 'Facil') ? 'selected' : ''; ?>>Fácil</option>

<option value="Medio" <?php echo ($pergunta\_data['nivel\_dificuldade'] == 'Medio') ? 'selected' : ''; ?>>Médio</option>

<option value="Dificil" <?php echo ($pergunta\_data['nivel\_dificuldade'] == 'Dificil') ? 'selected' : ''; ?>>Difícil</option>

</select>

</div>

<div class="form-group">

<label>Respostas (Marque a correta):</label>

<?php for($i = 0; $i < 4; $i++): ?>

<div class="radio-group">

<input type="radio" id="resposta\_<?php echo $i; ?>" name="correta\_indice" value="<?php echo $i; ?>" <?php echo ($respostas\_data[$i]['correta'] == 1) ? 'checked' : ''; ?> required>

<input type="text" name="respostas[]" placeholder="Texto da Resposta <?php echo $i + 1; ?>" value="<?php echo htmlspecialchars($respostas\_data[$i]['texto\_resposta']); ?>" required>

</div>

<?php endfor; ?>

</div>

<button type="submit" class="btn btn-submit">Salvar Pergunta</button>

<a href="admin\_dashboard.php" class="btn btn-back">Voltar</a>

</form>

</div>

</body>

</html>

**Passos para ti:**

1. **Atualiza o quiz\_game.php** com o novo bloco de código para a atualização da pontuação no DB (Seção 1).
2. **Atualiza o login.php** com a lógica para redirecionar o admin para admin\_dashboard.php (Seção 2).
3. **Cria o ficheiro admin\_dashboard.php** e cola o código fornecido (Seção 2.1).
4. **Cria o ficheiro gerir\_perguntas.php** e cola o código fornecido (Seção 2.2).
5. **Guarda todos os ficheiros.**

**Testes a realizar:**

* **Teste a pontuação:** Faz login como um usuário normal, joga o quiz e verifica no phpMyAdmin se a pontuacao\_total do teu usuário é atualizada quando acertas perguntas.
* **Teste o admin:** Se ainda não tens um, **registra um novo usuário no cadastro.php e, manualmente no phpMyAdmin, altera o id\_usuario desse usuário para 1**. (Isto é uma solução temporária para o objetivo do dia 4).
  + Faz login com as credenciais deste usuário "admin". Deverás ser redirecionado para admin\_dashboard.php.
  + No painel admin, tenta adicionar uma nova pergunta. Preenche todos os campos, incluindo as 4 respostas e a correta.
  + Verifica se a pergunta aparece na lista e se as respostas foram salvas no DB.
  + Tenta editar uma pergunta existente.
  + Tenta excluir uma pergunta.

Este é um dia com muitos passos, mas com eles terás o essencial para gerir o conteúdo do quiz e garantir a persistência das pontuações!

**Diz-me quando tiveres implementado e testado estes pontos. Se tiveres alguma dúvida ou encontrares um erro, estou aqui para ajudar.**