

```
</form>
</main>

<footer>
    <p>&copy; 2025 BitVagas - Todos os direitos
reservados.</p>
</footer>

<script>
    const remuneradoCheckbox =
document.getElementById('remunerado');
    const salarioContainer =
document.getElementById('salario-container');

    remuneradoCheckbox.addEventListener('change',
function() {
    if (this.checked) {
        salarioContainer.style.display = 'block';
    } else {
        salarioContainer.style.display = 'none';
        salarioContainer.querySelector('input').value = "";
    }
});
</script>
</body>
</html>
```

Apêndice 14

editar.php:

```
<?php
session_start();
require_once "./config/conexao.php";

if (!isset($_SESSION["empresa_nome"])) {
    header("Location: login.php");
    exit;
}

$empresa_nome = $_SESSION["empresa_nome"];
$id_empresa = $_SESSION["empresa_id"] ?? null;

$id_vaga = $_GET['id'] ?? null;

if (!$id_vaga) {
```

```
header("Location: painel_empresa.php");
exit;
}

$erro = "";
$sucesso = "";

$stmt = $conn->prepare("SELECT * FROM
vagas_completa WHERE id = ? AND empresa = ? LIMIT
1");
$stmt->bind_param("is", $id_vaga, $empresa_nome);
$stmt->execute();
$resultado = $stmt->get_result();

if ($resultado->num_rows !== 1) {
    header("Location: painel_empresa.php");
    exit;
}

$vaga = $resultado->fetch_assoc();

if ($_SERVER["REQUEST_METHOD"] == "POST") {
    $titulo_vaga = trim($_POST['titulo_vaga'] ?? "");
    $descricao_vaga = trim($_POST['descricao_vaga'] ?? "");
    $linguagem = trim($_POST['linguagem'] ?? "");
    $requisitos = trim($_POST['requisitos'] ?? "");
```

```

$beneficios = trim($_POST['beneficios'] ?? "");
$tipo_contrato = $_POST['tipo_contrato'] ?? 'Estágio';
$carga_horaria = trim($_POST['carga_horaria'] ?? "");
$localidade = trim($_POST['localidade'] ?? "");
$modalidade = $_POST['modalidade'] ?? 'Presencial';
$remunerado = isset($_POST['remunerado']) ? 1 : 0;
$salario = $remunerado ? floatval($_POST['salario']) :
0;
$email_contato = trim($_POST['email_contato'] ?? "");
$site_empresa = trim($_POST['site_empresa'] ?? "");
$telefone = trim($_POST['telefone'] ?? "");

if (empty($titulo_vaga) || empty($descricao_vaga) ||
empty($linguagem) || empty($localidade)) {
    $erro = "Preencha todos os campos obrigatórios!";
} else {
    $sql = "UPDATE vagas_completa SET
            titulo_vaga = ?, descricao_vaga = ?,
            linguagem = ?, requisitos = ?, beneficios = ?,
            tipo_contrato = ?, carga_horaria = ?,
            localidade = ?, modalidade = ?,
            remunerado = ?, salario = ?, email_contato =
            ?, site_empresa = ?, telefone = ?
            WHERE id = ? AND empresa = ?";
    $stmt = $conn->prepare($sql);

    if ($stmt === false) {

```

```
$erro = "Erro ao preparar a query. Causa: " .  
$conn->error;  
} else {  
    $stmt->bind_param(  
        "ssssssssidsssis",  
        $titulo_vaga,  
        $descricao_vaga,  
        $linguagem,  
        $requisitos,  
        $beneficios,  
        $tipo_contrato,  
        $carga_horaria,  
        $localidade,  
        $modalidade,  
        $remunerado,  
        $salario,  
        $email_contato,  
        $site_empresa,  
        $telefone,  
        $id_vaga,  
        $empresa_nome  
    );  
  
    if ($stmt->execute()) {  
        $sucesso = "Vaga atualizada com sucesso!";  
        $vaga = array_merge($vaga, $_POST);  
    } else {
```

```
$erro = "Erro ao atualizar a vaga. Tente
novamente.";

    }
}
}
}
?>
<!DOCTYPE html>
<html lang="pt-BR">
<head>
    <meta charset="UTF-8">
    <title>Editar Vaga - BitVagas</title>
    <link rel="stylesheet" href="css/controle_vaga.css">
</head>
<body>
    <header>
        <nav>
            <h3>BitVagas</h3>
            <ul class="menu">
                <li><a
href="painel_empresa.php">Painel</a></li>
            </ul>
            <p>Olá, <?php echo
htmlspecialchars($empresa_nome); ?>!</p>
        </nav>
    </header>
```

```
<main>
    <h1>Editar Vaga</h1>

    <?php if ($erro) echo "<p class='erro'>$erro</p>"; ?>
    <?php if ($sucesso) echo "<p
class='sucesso'>$sucesso</p>"; ?>

    <form method="POST" class="vaga-form">
        <label for="titulo_vaga">Título da Vaga*</label>
        <input type="text" id="titulo_vaga"
name="titulo_vaga" required value="<?=
htmlspecialchars($vaga['titulo_vaga']) ?>">

        <label for="descricao_vaga">Descrição*</label>
        <textarea id="descricao_vaga"
name="descricao_vaga" required><?=
htmlspecialchars($vaga['descricao_vaga']) ?></textarea>

        <label for="linguagem">Linguagem*</label>
        <input type="text" id="linguagem"
name="linguagem" required value="<?=
htmlspecialchars($vaga['linguagem']) ?>">

        <label for="requisitos">Requisitos:</label>
        <textarea id="requisitos" name="requisitos"><?=
htmlspecialchars($vaga['requisitos']) ?></textarea>
```

```
<label for="beneficios">Benefícios:</label>
<textarea id="beneficios" name="beneficios"><?=
htmlspecialchars($vaga['beneficios']) ?></textarea>

<label for="tipo_contrato">Tipo de
Contrato:</label>
<select id="tipo_contrato" name="tipo_contrato">
<?php
$tipos = ['Estágio', 'CLT', 'PJ', 'Temporário'];
foreach ($tipos as $tipo) {
    $selected = ($vaga['tipo_contrato'] === $tipo)
? "selected" : "";
    echo "<option value='$tipo'
$selected>$tipo</option>";
}
?>
</select>

<label for="carga_horaria">Carga Horária:</label>
<input type="text" id="carga_horaria"
name="carga_horaria" value="<?=
htmlspecialchars($vaga['carga_horaria']) ?>">

<label for="localidade">Localidade*:</label>
<input type="text" id="localidade"
name="localidade" required value="<?=
htmlspecialchars($vaga['localidade']) ?>">
```

```
<label for="modalidade">Modalidade:</label>
<select id="modalidade" name="modalidade">
    <?php
        $modalidades = ['Presencial', 'Híbrido',
'Remoto'];
        foreach ($modalidades as $mod) {
            $selected = ($vaga['modalidade'] === $mod)
? "selected" : "";
            echo "<option value='$mod'
$selected>$mod</option>";
        }
    ?>
</select>

<label>
    <input type="checkbox" name="remunerado"
value="1" <?= $vaga['remunerado'] ? "checked" : "" ?>>
    Remunerado
</label>

<label for="salario">Salário:</label>
<input type="number" step="0.01" id="salario"
name="salario" value="<?
htmlspecialchars($vaga['salario']) ?>">
```

```
<label for="email_contato">Email de  
Contato:</label>  
    <input type="email" id="email_contato"  
name="email_contato" value="<?=  
htmlspecialchars($vaga['email_contato']) ?>">  
  
    <label for="site_empresa">Site da  
Empresa:</label>  
    <input type="url" id="site_empresa"  
name="site_empresa" value="<?=  
htmlspecialchars($vaga['site_empresa']) ?>">  
  
    <label for="telefone">Telefone:</label>  
    <input type="text" id="telefone" name="telefone"  
value="<?= htmlspecialchars($vaga['telefone']) ?>">  
  
    <button type="submit"  
class="btn-destaque">Atualizar Vaga</button>  
    <a href="painel_empresa.php"  
class="btn-limpar">Voltar</a>  
    </form>  
</main>  
  
<footer>  
    <p>&copy; 2025 BitVagas - Todos os direitos  
reservados.</p>  
</footer>
```

```
</body>
</html>
```

Apêndice 15

desativar.php:

```
<?php
session_start();
require_once "./config/conexao.php";

if (!isset($_SESSION["empresa_nome"])) {
    header("Location: login.php");
    exit;
}

if ($_SERVER["REQUEST_METHOD"] === "POST") {
    $id_vaga = intval($_POST["id_vaga"]);
    $senha = $_POST["senha"];
    $empresa_nome = $_SESSION["empresa_nome"];

    $sql_verificar = "SELECT senha FROM empresa
WHERE nome = ?";
    $stmt = $conn->prepare($sql_verificar);
    $stmt->bind_param("s", $empresa_nome);
```

```
$stmt->execute();
$resultado = $stmt->get_result();

if ($resultado->num_rows === 1) {
    $dados = $resultado->fetch_assoc();
    if (password_verify($senha, $dados["senha"])) {
        $sql_update = "UPDATE vagas_estagio SET ativa
= 0 WHERE id = ? AND empresa = ?";
        $stmt2 = $conn->prepare($sql_update);
        $stmt2->bind_param("is", $id_vaga,
$empresa_nome);
        if ($stmt2->execute()) {
            header("Location:
painelempresa.php?msg=Vaga desativada com
sucesso");
            exit;
        } else {
            echo "Erro ao desativar a vaga.";
        }
    } else {
        echo "Senha incorreta.";
    }
} else {
    echo "Empresa não encontrada.";
}
?>
```

Apêndice 16

Status.php:

```
<?php
session_start();
require_once "./config/conexao.php";

if (!isset($_SESSION["empresa_nome"])) {
    header("Location: login.php");
    exit;
}

if ($_SERVER["REQUEST_METHOD"] === "POST") {
    $id_vaga = intval($_POST["id_vaga"]);
    $senha = $_POST["senha"];
    $empresa_nome = $_SESSION["empresa_nome"];

    $sql_verificar = "SELECT senha FROM empresa
WHERE nome = ?";
    $stmt = $conn->prepare($sql_verificar);
    $stmt->bind_param("s", $empresa_nome);
```

```
$stmt->execute();
$resultado = $stmt->get_result();

if ($resultado->num_rows === 1) {
    $dados = $resultado->fetch_assoc();

    if (password_verify($senha, $dados["senha"])) {
        $sql_status = "SELECT ativa FROM
vagas_completa WHERE id = ? AND empresa = ?";
        $stmt_status = $conn->prepare($sql_status);
        $stmt_status->bind_param("is", $id_vaga,
$empresa_nome);
        $stmt_status->execute();
        $resultado_status = $stmt_status->get_result();

        if ($resultado_status->num_rows === 1) {
            $vaga = $resultado_status->fetch_assoc();
            $nova_atividade = $vaga["ativa"] ? 0 : 1;

            $sql_update = "UPDATE vagas_completa SET
ativa = ? WHERE id = ? AND empresa = ?";
            $stmt2 = $conn->prepare($sql_update);
            $stmt2->bind_param("iis", $nova_atividade,
$id_vaga, $empresa_nome);

            if ($stmt2->execute()) {
```

```
$msg = $nova_atividade ? "Vaga ativada com  
sucesso" : "Vaga desativada com sucesso";  
header("Location:  
painelempresa.php?msg=".urlencode($msg));  
exit;  
} else {  
    echo "Erro ao atualizar o status da vaga.";  
}  
} else {  
    echo "Vaga não encontrada.";  
}  
} else {  
    echo "Senha incorreta.";  
}  
} else {  
    echo "Empresa não encontrada.";  
}  
}  
?>
```

Apêndice 17

Enviar_mensagem.php:

```
<?php
session_start();
require_once "./config/conexao.php";
header("Content-Type: application/json");

if ($_SERVER["REQUEST_METHOD"] === "POST") {

    if (!isset($_SESSION["empresa_id"])) {
        echo json_encode(["status" => "erro", "mensagem"
=> "Acesso negado."]);
        exit;
    }

    $id_empresa = $_SESSION["empresa_id"];
    $id_candidatura = isset($_POST["id"]) ?
intval($_POST["id"]) : 0;
```

```
$mensagem = trim($_POST["mensagem"] ?? "");  
$tipo = $_POST["tipo"] ?? "info";  
  
  
$sqlCandidatura = "SELECT id_usuario FROM  
candidaturas WHERE id = ?";  
$stmtCandidatura = $conn->prepare($sqlCandidatura);  
$stmtCandidatura->bind_param("i", $id_candidatura);  
$stmtCandidatura->execute();  
$resultado = $stmtCandidatura->get_result();  
  
  
if ($resultado->num_rows === 0) {  
    echo json_encode(["status" => "erro", "mensagem"  
=> "Candidatura não encontrada."]);  
    exit;  
}  
  
  
$row = $resultado->fetch_assoc();  
$id_usuario = $row["id_usuario"];  
  
  
$sql = "INSERT INTO mensagens (id_candidatura,  
id_usuario, id_empresa, tipo, mensagem) VALUES (?, ?,  
?, ?, ?);  
$stmt = $conn->prepare($sql);  
$stmt->bind_param("iiiss", $id_candidatura,  
$id_usuario, $id_empresa, $tipo, $mensagem);
```

```
if ($stmt->execute()) {  
    echo json_encode(["status" => "sucesso",  
"mensagem" => "Mensagem enviada com sucesso!");  
} else {  
    echo json_encode(["status" => "erro", "mensagem"  
=> "Erro ao enviar mensagem."]);  
}  
}  
?>
```

Apêndice 18

Ver_mensagens.php:

```
<?php
session_start();
require_once "./config/conexao.php";

header("Content-Type: application/json");

if (!isset($_SESSION["usuario_id"])) {
    echo json_encode([]);
    exit;
}

$id_usuario = $_SESSION["usuario_id"];

$sql = "SELECT id, id_candidatura, id_empresa,
mensagem, tipo, data_envio
```

```
FROM mensagens
WHERE id_usuario = ?
ORDER BY data_envio DESC";

$stmt = $conn->prepare($sql);
$stmt->bind_param("i", $id_usuario);
$stmt->execute();
$result = $stmt->get_result();

$mensagens = [];
while ($msg = $result->fetch_assoc()) {
    $mensagens[] = $msg;
}

error_log("Debug ID Usuário: $id_usuario");
error_log(print_r($mensagens, true));

echo json_encode($mensagens);

?>
```

Estilização - Apêndice 1

css/HomeStyle.css:

```
* {  
    margin: 0;  
    padding: 0;  
    box-sizing: border-box;  
}  
  
body {  
    font-family: Arial, Helvetica, sans-serif;  
    background: radial-gradient(circle at 30% 20%, #1e5be0,  
    #0d1b4c 60%, #0a0f2d 100%);  
    background-attachment: fixed;  
    color: #fff;  
    line-height: 1.6;  
    min-height: 100vh;  
    display: flex;
```

```
flex-direction: column;
}

header {
  background: #0a1a3c;
  padding: 1rem;
  position: sticky;
  top: 0;
  z-index: 10;
}

header nav {
  display: flex;
  justify-content: space-between;
  align-items: center;
}

header h3 {
  font-size: 1.8rem;
  color: blue;
  letter-spacing: 1px;
  text-align: center;
}

header button {
  background: rgba(255, 255, 255, 0.15);
  border: none;
```

```
color: #fff;
padding: 0.6rem 1rem;
border-radius: 6px;
cursor: pointer;
transition: 0.3s;
}

header button:hover {
    background: rgba(255, 255, 255, 0.3);
}

.intro {
    text-align: center;
    padding: 3rem 1rem 2rem;
}

.intro h1 {
    font-size: 2.2rem;
    font-weight: bold;
    color: #ffffff;
    text-shadow: 0 2px 8px rgba(0,0,0,0.4);
}

main {
    flex: 1;
    padding: 2rem;
}
```

```
section {  
    margin: 3rem 0;  
    display: flex;  
    justify-content: center;  
}
```

```
article {  
    position: relative;  
    width: 80%;  
    max-width: 800px;  
    padding: 2rem;  
    border-radius: 16px;  
    color: #fff;  
    z-index: 1;  
    background: rgba(255, 255, 255, 0.08);  
    backdrop-filter: blur(20px) saturate(180%);  
    -webkit-backdrop-filter: blur(20px) saturate(180%);  
    border: 1px solid rgba(255, 255, 255, 0.2);  
    box-shadow: 0 10px 25px rgba(0,0,0,0.5);  
    transition: transform 0.3s;  
}
```

```
article::before {  
    content: "";  
    position: absolute;  
    top: 14px;
```

```
left: 14px;  
width: 100%;  
height: 100%;  
border-radius: 16px;  
background: linear-gradient(135deg, #0d47a1, #1976d2,  
#42a5f5);  
z-index: -1;  
opacity: 0.5;  
box-shadow: 0 8px 18px rgba(0,0,0,0.6);  
}  
  
article:hover {  
    transform: translateY(-5px);  
}
```

```
h1, h2 {  
    margin-bottom: 1rem;  
    color: #04bcfa;  
}
```

```
.btn {  
    display: flex;  
    margin-top: 1rem;  
    padding: 0.7rem 1.5rem;  
    background: rgba(66, 165, 245, 0.3);  
    color: #fff;  
    border-radius: 8px;
```