

index.html

```
<!DOCTYPE html>
<html lang="pt-br">
<head>
  <meta charset="UTF-8">
  <title>Analisador Léxico</title>
  <link rel="stylesheet" href="css/style.css">
</head>
<body>
  <div class="container">
    <h1>Analisador Léxico</h1>

    <div class="secao">
      <h2>Adicionar palavras</h2>
      <input type="text" id="entrada-palavra" placeholder="Digite uma palavra...">
      <button onclick="adicionarPalavra()">+ ADICIONAR</button>
      <ul id="lista-palavras"></ul>
      <button class="limpar-btn" onclick="limparPalavras()">[limpar] LIMPAR TODAS AS
PALAVRAS</button>
    </div>

    <div class="secao">
      <h2>Validar palavras</h2>
      <input type="text" id="campo-validar" placeholder="Digite para validar...">
      <div id="resultado-validacao" class="resultado-validacao"></div>
      <p><strong>Reconhecidas:</strong></p>
      <ul id="validadas"></ul>
      <p><strong>Rejeitadas:</strong></p>
      <ul id="rejeitadas"></ul>
      <p><strong>Caminho:</strong> <span id="caminho-resultado"></span></p>
    </div>

    <div class="secao">
      <h2>Tabela de estados</h2>
      <div style="overflow-x:auto;">
        <table id="tabela-estados"></table>
      </div>
    </div>
  </div>

  <script src="js/afd.js"></script>
</body>
</html>
```

style.css

```
:root {
  --verde: #148a0a;
  --preto: #222;
  --cinza-claro: #f5f5f5;
  --vermelho: #d50000;
  --azul: #d0e6ff;
}

body {
  font-family: Arial, sans-serif;
  margin: 0;
  background-color: white;
}

.container {
  max-width: 1080px;
  margin: 0 auto;
  padding: 40px 20px;
}

h1 {
  background-color: var(--verde);
  color: white;
  padding: 18px 30px;
  font-size: 26px;
  margin-bottom: 30px;
  border-radius: 8px;
}

.secao {
  background-color: var(--cinza-claro);
  padding: 25px;
  margin-bottom: 25px;
  border-radius: 8px;
}

input[type="text"] {
  width: 100%;
  padding: 12px;
  font-size: 16px;
  margin-bottom: 15px;
  border: 1px solid #ccc;
  border-radius: 4px;
}

button {
  background-color: var(--verde);
  color: white;
  font-weight: bold;
  border: none;
```

```
padding: 10px 20px;
font-size: 15px;
border-radius: 4px;
cursor: pointer;
margin: 5px 5px 5px 0;
display: inline-flex;
align-items: center;
gap: 8px;
}

.limpar-btn {
  background-color: var(--preto);
}

button:hover {
  opacity: 0.85;
}

table {
  width: 100%;
  border-collapse: collapse;
  margin-top: 10px;
}

th, td {
  border: 1px solid #ccc;
  padding: 6px;
  text-align: center;
  font-size: 14px;
}

th {
  background-color: #e9e9e9;
}

.cell-final-ok {
  background-color: #c0ffc0 !important;
}

.cell-final-erro {
  background-color: #ffc0c0 !important;
}

ul {
  padding-left: 20px;
}

li {
  margin-bottom: 5px;
  display: flex;
  justify-content: space-between;
  align-items: center;
  max-width: 300px;
  padding: 4px 8px;
```

```
    border-radius: 5px;
}

li.reconhecida {
    background-color: #d8ffd8;
}

li.rejeitada {
    background-color: #ffd8d8;
}

li button {
    background-color: transparent;
    color: var(--vermelho);
    border: none;
    font-size: 16px;
    cursor: pointer;
    transition: color 0.2s ease;
}

li button:hover {
    color: #a00000;
}

.resultado-validacao {
    font-weight: bold;
    font-size: 16px;
    margin-top: 10px;
}

.resultado-validacao.ok {
    color: green;
}

.resultado-validacao.erro {
    color: red;
}

li.adicionada {
    background-color: #e0e0e0;
}
```

afd.js

```
let palavras = [];
let afd = {};
let finais = new Set();
const alfabeto = 'abcdefghijklmnopqrstuvwxyz'.split('');

function adicionarPalavraManual(palavra) {
  if (!palavra || palavras.includes(palavra)) return;
  palavras.push(palavra);
  atualizarAFD();
}

function adicionarPalavra() {
  const entrada = document.getElementById('entrada-palavra');
  const palavra = entrada.value.trim().toLowerCase();
  entrada.value = '';
  adicionarPalavraManual(palavra);
}

document.getElementById('entrada-palavra').addEventListener('keydown', function(e) {
  if (e.key === ' ' || e.key === 'Enter') {
    e.preventDefault();
    adicionarPalavra();
  }
});

function atualizarAFD() {
  afd = {};
  finais.clear();
  let contadorEstado = 1;
  for (let palavra of palavras) {
    let estado = 'q0';
    if (!afd[estado]) afd[estado] = {};
    for (let i = 0; i < palavra.length; i++) {
      const letra = palavra[i];
      const proximo = afd[estado][letra] || `q${contadorEstado++}`;
      afd[estado][letra] = proximo;
      estado = proximo;
      if (!afd[estado]) afd[estado] = {};
    }
    finais.add(estado);
  }
  atualizarListaPalavras();
  desenharTabela();
}

function atualizarListaPalavras() {
  const ul = document.getElementById('lista-palavras');
  ul.innerHTML = '';
  for (let palavra of palavras) {
    const li = document.createElement('li');
```

```

li.classList.add('adicionada');
li.appendChild(document.createTextNode(palavra));

const botao = document.createElement('button');
botao.textContent = 'x';
botao.onclick = function () {
  palavras = palavras.filter(p => p !== palavra);
  atualizarAFD();
};
li.appendChild(botao);
ul.appendChild(li);
}
}

function desenharTabela() {
  const tabela = document.getElementById('tabela-estados');
  tabela.innerHTML = '';

  const thRow = document.createElement('tr');
  thRow.innerHTML = '<th>Estado</th>' + alfabeto.map(s => `<th>${s}</th>`).join('');
  tabela.appendChild(thRow);

  for (let estado in afd) {
    const tr = document.createElement('tr');
    const ehFinal = finais.has(estado);
    tr.innerHTML = `<td${ehFinal ? ' style="font-weight:bold;color:green;" : ''}>${estado}</td>`
+
    alfabeto.map(s => {
      const destino = afd[estado][s];
      return `<td data-estado="${estado}" data-letra="${s}">${destino ? destino : ''}</td>`;
    }).join('');
    tabela.appendChild(tr);
  }
}

document.getElementById('campo-validar').addEventListener('input', function () {
  const entrada = this.value.trim().toLowerCase();
  const caminhoSpan = document.getElementById('caminho-resultado');
  caminhoSpan.textContent = '';

  document.querySelectorAll('td').forEach(td => {
    td.classList.remove('cell-final-ok', 'cell-final-erro');
  });

  if (!entrada || Object.keys(afd).length === 0) return;

  let estado = 'q0';
  let caminho = [{ nome: estado, erro: false }];
  let reconhecida = true;

  for (let i = 0; i < entrada.length; i++) {
    const letra = entrada[i];
    const prox = afd[estado]?.[letra];
    const celula = document.querySelector(`td[data-estado="${estado}"][data-letra="${letra}"]`);

```

```

if (prox) {
  if (celula) celula.classList.add('cell-final-ok');
  estado = prox;
  caminho.push({ nome: estado, erro: false });
} else {
  if (celula) celula.classList.add('cell-final-erro');
  caminho.push({ nome: '?', erro: true });
  reconhecida = false;
  break;
}
}

const caminhoHtml = caminho.map(item =>
  `<span style="color:${item.erro ? 'red' : 'inherit'}">${item.nome}</span>`
).join(' -> ');
document.getElementById('caminho-resultado').innerHTML = caminhoHtml;
});

document.getElementById('campo-validar').addEventListener('keydown', function (e) {
  if (e.key === ' ' || e.key === 'Enter') {
    e.preventDefault();
    const entrada = this.value.trim().toLowerCase();
    if (!entrada) return;

    const caminho = [];
    let estado = 'q0';
    let reconhecida = true;
    caminho.push(estado);
    for (let letra of entrada) {
      if (afd[estado] && afd[estado][letra]) {
        estado = afd[estado][letra];
        caminho.push(estado);
      } else {
        caminho.push('?');
        reconhecida = false;
        break;
      }
    }
    if (!finais.has(estado)) reconhecida = false;

    const resultado = document.getElementById('resultado-validacao');
    resultado.textContent = reconhecida ? '[ok] Palavra reconhecida' : 'x Palavra rejeitada';
    resultado.className = 'resultado-validacao ' + (reconhecida ? 'ok' : 'erro');

    const ul = document.getElementById(reconhecida ? 'validadas' : 'rejeitadas');
    const li = document.createElement('li');
    li.textContent = entrada;
    li.classList.add(reconhecida ? 'reconhecida' : 'rejeitada');

    const botao = document.createElement('button');
    botao.textContent = 'x';
    botao.onclick = () => li.remove();
    li.appendChild(botao);
    ul.appendChild(li);
  }
});

```

```
        this.value = '';
    }
});

function limparPalavras() {
    palavras = [];
    afd = {};
    finais.clear();
    document.getElementById('lista-palavras').innerHTML = '';
    document.getElementById('validadas').innerHTML = '';
    document.getElementById('rejeitadas').innerHTML = '';
    document.getElementById('tabela-estados').innerHTML = '';
    document.getElementById('caminho-resultado').textContent = '';
    document.getElementById('resultado-validacao').textContent = '';
}
```