

INSTITUTO DE INTELIGÊNCIA ARTIFICIAL APLICADA (I2A2)

REDES GENERATIVAS

JOÃO PAULO DA SILVA CARDOSO

RELATÓRIO:

TEMA 1: CRIAÇÃO DE SITE UTILIZANDO REDES GENERATIVAS

12/06/2023

## 1. INTRODUÇÃO

Este relatório tem como objetivo apresentar os passos realizados para solucionar o desafio 1 do curso de redes generativas do instituto I2A2. A tarefa escolhida foi o Tema:1 que consiste em criar um site atendendo a alguns requisitos que serão explorados a seguir. Como resultado a página de vendas fictícias de bonés chamada **“Next Gen Caps”** foi criada, contendo 10 modelos nomeados, com ids e valores de cada produto.

## 2. DESENVOLVIMENTO

Ao entrar no chagpt e fazer o login, o primeiro prompt que digitei foi:

***” You are a senior frontend engineer, with solid ux design skills”***

Após obter uma confirmação do entendimento, segui pedindo para que os requisitos fossem atendidos com os prompts que serão vistos a seguir.

### 2.1 Página de Login

Os requisitos para criação da Página de Login são os seguintes:

- Campo para inserir nome de usuário.
- Campo para inserir senha.
- Botão "Entrar" para efetuar o login.
- Link "Esqueci minha senha" para acessar a página de recuperação de senha.
- Validação para garantir que os campos estejam preenchidos antes de permitir o login.
- Redirecionamento para a página de Lista de Produtos após o login bem-sucedido.

O seguinte prompt foi pedido para sanar esta demanda, e de forma básica o chat atendeu.

***“Create a website with the following pages:***

***• Page 1: Login***

***Screen with fields for username and password, the "login" button, and the "forgot my password" link.***

***The fields must be filled in when clicking the login button.***

***The login button redirects to the product page.”***

Após fazer algumas alterações com a saída do chatgpt decidi por separar os arquivos css para cada página, sendo três no total com seus respectivos arquivos css, abaixo o código final da página de login e seu arquivo css, com as funcionalidades pedidas, devidamente atendidas.

```
<!DOCTYPE html>
<html>
<head>
  <title>Next Gen Caps</title>
  <link rel="stylesheet" href="loginstyles.css">
</head>
<body>
  <div>
    <h1>Next Gen Caps</h1>
    
  </div>
  <form>
    <label for="username">Username:</label>
    <input type="text" id="username" required>

    <label for="password">Password:</label>
    <input type="password" id="password" required>

    <button type="submit">Login</button>
  </form>
  <a href="forgot-password.html">Forgot my password</a>

  <script>
    // Add JavaScript logic to handle form submission and redirection
    document.querySelector('form').addEventListener('submit', function(e) {
      e.preventDefault();
      const username = document.getElementById('username').value;
      const password = document.getElementById('password').value;

      // Perform login authentication, e.g., through an API call

      // Redirect to the product page after successful login
      window.location.href = 'product-list.html';
    });
  </script>
</body>
</html>
```

```
/* AI-inspired theme */
```

```
/* Global styles */
```

```
body {
  font-family: "Arial", sans-serif;
  margin: 0;
```

```
padding: 0;
background-color: #f0f0f0;
display: flex;
justify-content: center;
align-items: center;
min-height: 100vh;
}
```

```
.container {
width: 400px;
padding: 20px;
text-align: center;
background-color: #222;
color: #fff;
margin-top: 40px;
}
```

```
h1 {
font-size: 32px;
margin-bottom: 20px;
text-align: center;
border-radius: 4px;
border: 1px solid #555;
background-color: #333;
color: #fff;
}
```

```
form {
display: flex;
flex-direction: column;
align-items: center;
}
```

```
label {
margin-bottom: 15px;
font-size: 16px;
color: #fff;
}
```

```
input[type="text"],
input[type="password"] {
width: 100%;
padding: 10px;
font-size: 14px;
border-radius: 4px;
border: 1px solid #555;
background-color: #333;
color: #fff;
}
```

```
button[type="submit"] {
  width: 100%;
  padding: 10px;
  font-size: 16px;
  font-weight: bold;
  text-transform: uppercase;
  background-color: #555;
  color: #fff;
  border: none;
  border-radius: 40px;
  cursor: pointer;
}

button[type="submit"]:hover {
  background-color: #111;
}

a {
  display: block;
  margin-top: 250px;
  margin-left: -150px;
  text-align: center;
  font-size: 14px;
  color: #fff;
}

a:hover {
  text-decoration: underline;
}

/* Edgy brand styles */
body {
  background-color: #000;
}

.container {
  background-color: #222;
  color: #fff;
}

h1 {
  font-size: 32px;
}

label,
input[type="text"],
input[type="password"] {
  color: #fff;
  background-color: #333;
```

```
border-color: #555;
}

button[type="submit"] {
  background-color: #555;
}
```

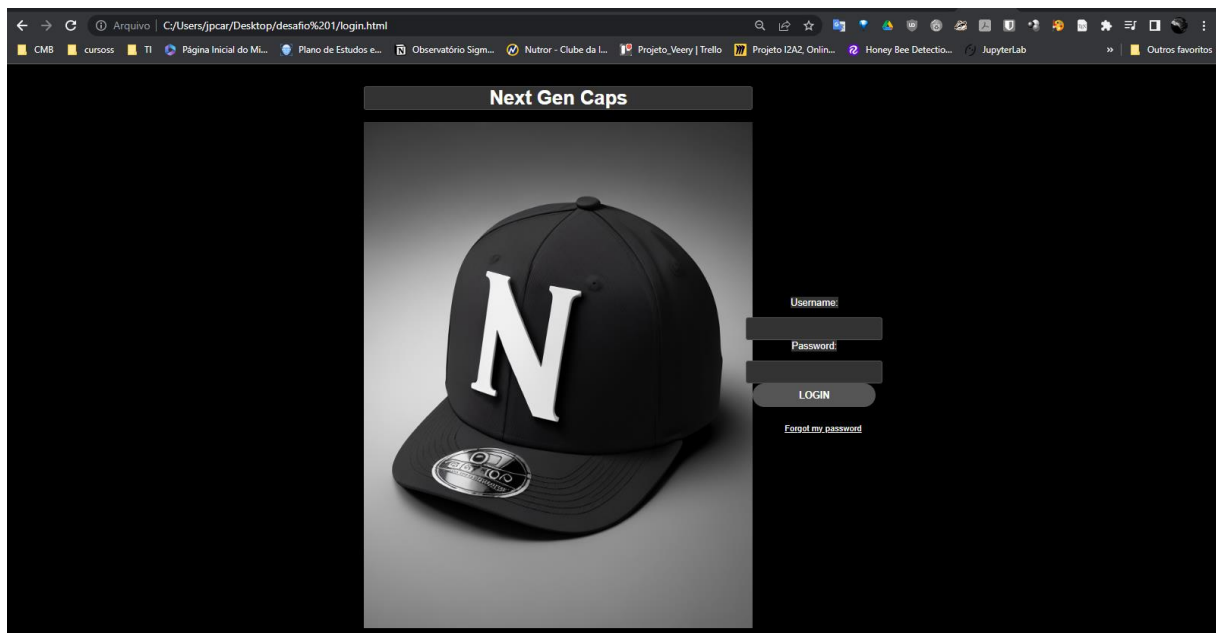


Figura 1 Página de login

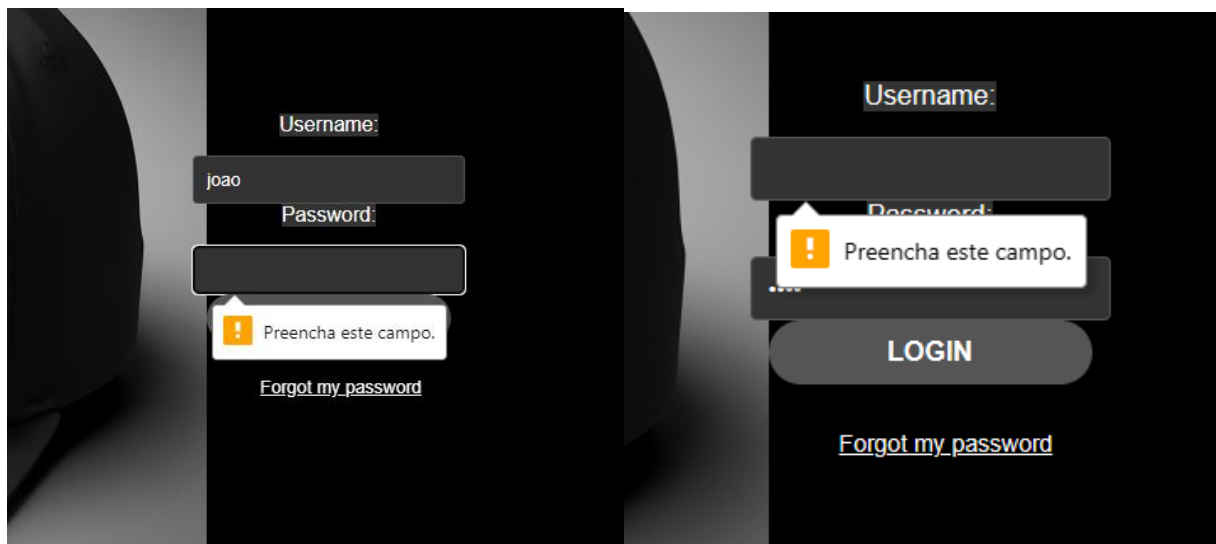


Figura 2 Imagem pedindo para validar login.

## 2.2 Página de Lista de Produtos

Os requisitos para criação da Página de Lista de Produtos são os seguintes:

- Apresentação de uma lista de 10 produtos.

- Cada produto deve exibir: ID do produto, nome, preço unitário e foto.
- Utilizar um logo hipotético gerado por uma rede generativa de imagens.
- Implementação em JavaScript.

O seguinte prompt foi pedido para sanar esta demanda, e de forma básica o chat respondeu.

**• Page 2: Product List**

*Present a page with a list of 10 products (of your choice). The list should include the product ID, product name, unit price, and product photo.*

Após fazer algumas alterações com a saída do chat, os códigos finais da página de recuperação de senha e seu css encontram-se abaixo

```
<!DOCTYPE html>
<html>
<head>
  <title>Products List</title>
  <link rel="stylesheet" href="styles.css">
</head>
<body>
  <h1>Product List</h1>
  <ul>
    <li>
      <div>
        
      </div>
      <div>
        <span>Product ID: 1</span>
        <span>Product Name: Cibertech</span>
        <span>Unit Price: $10.00</span>
      </div>
    </li>
    <li>
      <div>
        
      </div>
      <div>
        <span>Product ID: 2</span>
        <span>Product Name: NeoWave</span>
        <span>Unit Price: $10.00</span>
      </div>
    </li>
    <li>
      <div>
        
```

```

</div>
<div>
  <span>Product ID: 3</span>
  <span>Product Name: TechGlow</span>
  <span>Unit Price: $10.00</span>
</div>
</li>
<li>
  <div>
    
  </div>
  <div>
    <span>Product ID: 4</span>
    <span>Product Name: QuantumFit</span>
    <span>Unit Price: $10.00</span>
  </div>
</li>
<li>
  <div>
    
  </div>
  <div>
    <span>Product ID: 5</span>
    <span>Product Name: HoloFlex</span>
    <span>Unit Price: $10.00</span>
  </div>
</li>
<li>
  <div>
    
  </div>
  <div>
    <span>Product ID: 6</span>
    <span>Product Name: AeroTech</span>
    <span>Unit Price: $10.00</span>
  </div>
</li>
<li>
  <div>
    
  </div>
  <div>
    <span>Product ID: 7</span>
    <span>Product Name: InfinitiCap</span>
    <span>Unit Price: $10.00</span>
  </div>
</li>
<li>
  <div>
    

```



```

    </div>
    <div>
      <span>Product ID: 8</span>
      <span>Product Name: SynapseVisor</span>
      <span>Unit Price: $10.00</span>
    </div>
  </li>
  <li>
    <div>
      
    </div>
    <div>
      <span>Product ID: 9</span>
      <span>Product Name: HyperWave</span>
      <span>Unit Price: $10.00</span>
    </div>
  </li>
  <li>
    <div>
      
    </div>
    <div>
      <span>Product ID: 10</span>
      <span>Product Name: UltraGrip</span>
      <span>Unit Price: $10.00</span>
    </div>
  </li>

  <!-- Repeat the above list item structure for the other products -->
</ul>
</body>
</html>

```

```

/* Global styles */
body {
  font-family: "Arial", sans-serif;
  margin: 0;
  padding: 0;
  background-color: #f0f0f0;
  font-size: 16px;
}

.container {
  max-width: 800px;
  margin: 0 auto;
  padding: 20px;
  background-color: #222;
  color: #fff;
}

```

```
h1 {
  color: #fff;
  font-size: 40px;
  font-weight: bold;
  text-transform: uppercase;
  letter-spacing: 4px;
  margin-top: 60px;
  margin-bottom: 40px;
  text-align: center;
  border-radius: 4px;
  border: 1px solid #555;
  background-color: #333;
  color: #fff;
}

ul {
  list-style: none;
  padding: 0;
  margin: 0;
  display: flex;
  flex-wrap: wrap;
}

li {
  flex-basis: 100%;
  background-color: #111;
  padding: 20px;
  margin-bottom: 20px;
  border-radius: 5px;
  box-shadow: 0 4px 8px rgba(0, 0, 0, 0.2);
  box-sizing: border-box;
}

@media screen and (min-width: 600px) {
  li {
    flex-basis: calc(50% - 20px);
    margin-right: 20px;
  }
}

@media screen and (min-width: 900px) {
  li {
    flex-basis: calc(33.33% - 20px);
    margin-right: 20px;
  }
}

li:last-child {
  margin-right: 0;
}
```

```

}

li div:first-child {
  margin-bottom: 10px;
}

li img {
  width: 100%;
  height: auto;
  border-radius: 5px;
}

li div:last-child {
  flex: 1;
}

span {
  display: block;
  margin-bottom: 10px;
  font-size: 18px;
  color: #fff;
}

```

As imagens dos produtos foram criadas artificialmente a partir da ferramenta Leonardo.ai com o seguinte prompt

***“A fashion cap for a modern, edgy brand, featuring a bold, all-caps font with sharp edges and a striking, asymmetrical graphic.”***

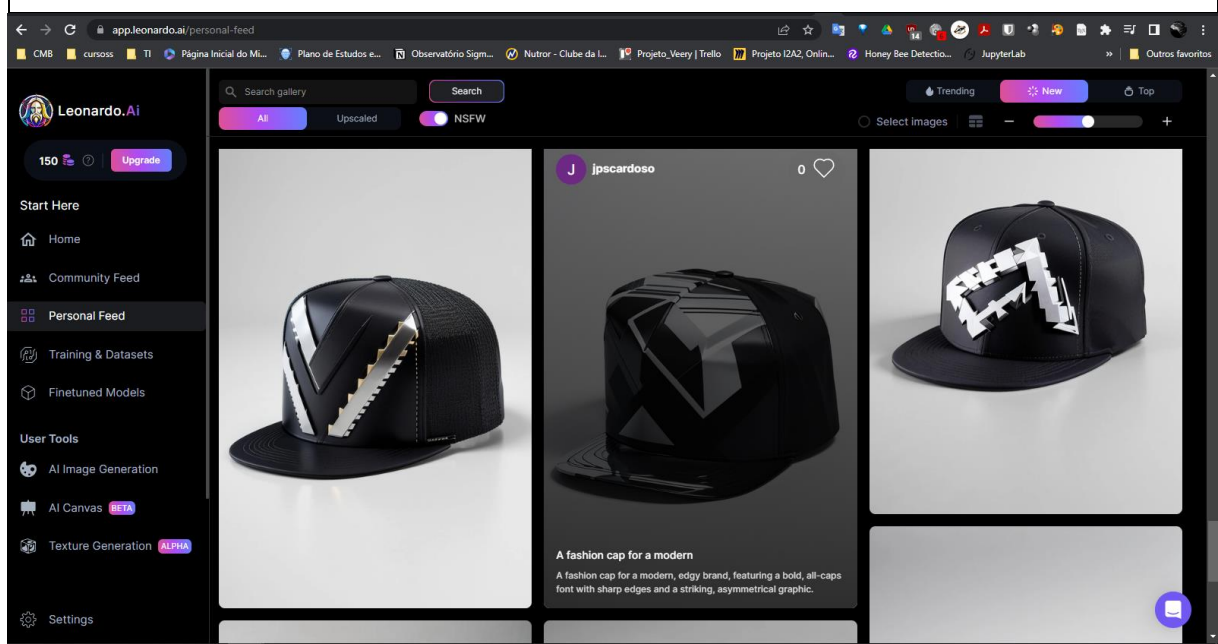


Figura 3 Imagem da página Leonardo exibindo as saídas do prompt.

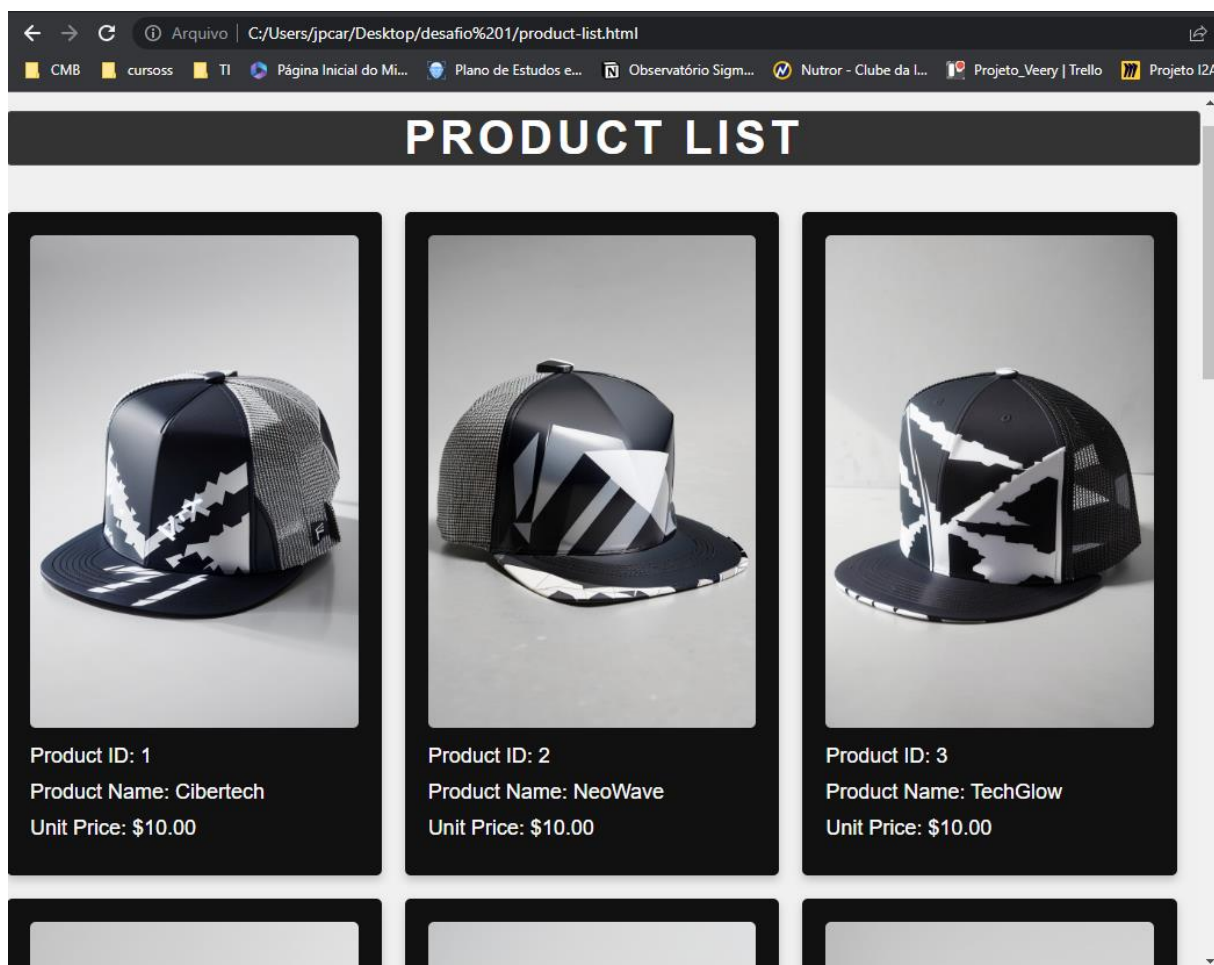


Figura 4 Amostra de alguns produtos listados na página.

### 2.3 Página de Recuperação de Senha

Os requisitos para criação da Página de Recuperação de senha são os seguintes:

- Campo para inserir o e-mail cadastrado.
- Botão "Redefinir Senha" para iniciar o processo de recuperação.
- Validação para garantir que o campo de e-mail esteja preenchido antes de prosseguir.
- Exibição de um pop-up informando que o link para redefinição foi enviado para o e-mail fornecido.
- Redirecionamento para a página de Login após o envio do link.

O seguinte prompt foi pedido para sanar esta demanda, e de forma básica o chat respondeu.

• **Page 3: Forgot My Password**

*Present a page with a field for the person to enter their registered email and a "reset password" button. Upon clicking the "reset password" button, validate if the email field is filled. Display a pop-up message stating that the reset link has been sent to the provided address and redirect to the login page."*

Após fazer algumas alterações com a saída do chat, os códigos finais da página de recuperação de senha e seu css encontram-se abaixo

```
<!DOCTYPE html>
<html>
<head>
<title>Forgot My Password</title>
<link rel="stylesheet" href="forgotstyles.css">
</head>
<body>
<form>
<label for="email">Email:</label>
<input type="email" id="email" required>

<button type="submit">Reset Password</button>
</form>

<script>
// Add JavaScript logic to handle form submission and redirection
document.querySelector('form').addEventListener('submit', function(e) {
  e.preventDefault();
  const email = document.getElementById('email').value;

  // Perform email validation and send reset password link

  // Display a pop-up message confirming the email has been sent
  alert('Reset link has been sent to ' + email);

  // Redirect to the login page
  window.location.href = 'login.html';
});
</script>
</body>
</html>
```

```
/* AI-inspired theme */
```

```
/* Global styles */
```

```
body {
  font-family: "Arial", sans-serif;
  margin: 0;
```

```
padding: 0;
background-color: #f9f9f9;
display: flex;
justify-content: center;
align-items: center;
min-height: 100vh;
}

.container {
width: 400px;
padding: 20px;
background-color: #ffffff;
border-radius: 8px;
box-shadow: 0 2px 8px rgba(0, 0, 0, 0.1);
}

h1 {
color: #333333;
font-size: 24px;
font-weight: bold;
text-transform: uppercase;
letter-spacing: 2px;
margin-bottom: 20px;
text-align: center;
}

form {
display: flex;
flex-direction: column;
align-items: center;
}

label {
margin-bottom: 10px;
font-size: 16px;
color: #333333;
}

input[type="email"] {
width: 100%;
padding: 10px;
font-size: 14px;
border-radius: 4px;
border: 1px solid #dddddd;
}

button[type="submit"] {
width: 100%;
padding: 10px;
font-size: 16px;
```

```
font-weight: bold;
text-transform: uppercase;
background-color: #333333;
color: #ffffff;
border: none;
border-radius: 4px;
cursor: pointer;
}

button[type="submit"]:hover {
  background-color: #111111;
}

/* Edgy brand styles */
body {
  background-color: #000000;
}

.container {
  background-color: #222222;
  color: #ffffff;
}

h1 {
  font-size: 32px;
}

label,
input[type="email"] {
  color: #ffffff;
  background-color: #333333;
  border-color: #555555;
}

button[type="submit"] {
  background-color: #555555;
}
```

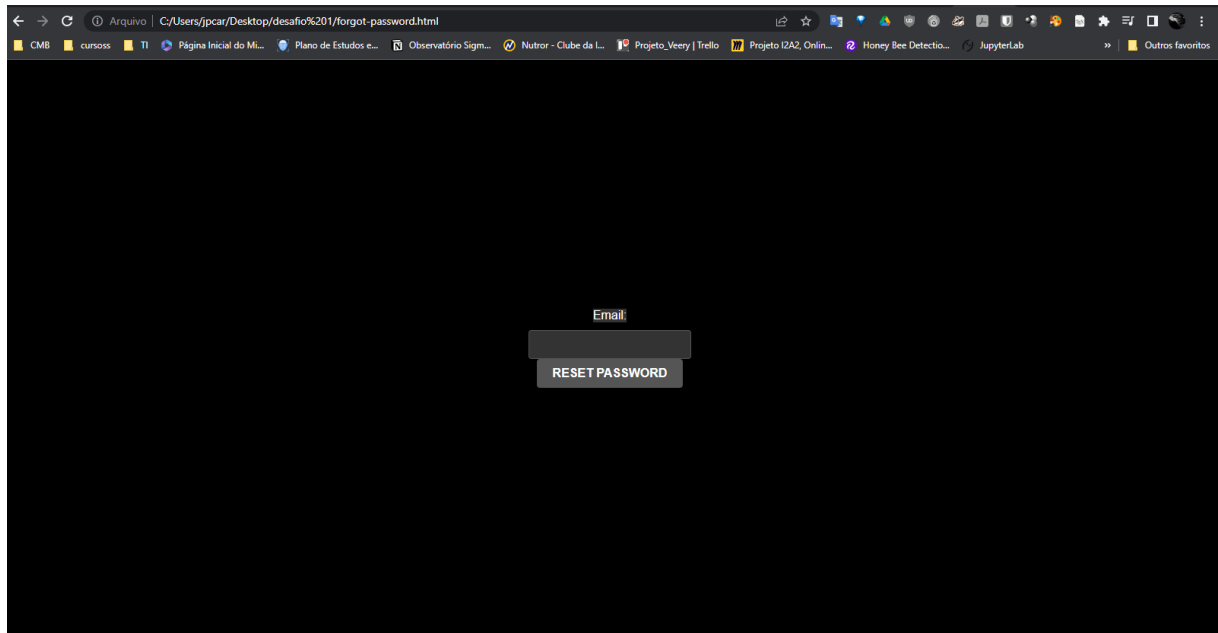


Figura 5 Imagem da página de recuperação de senha.

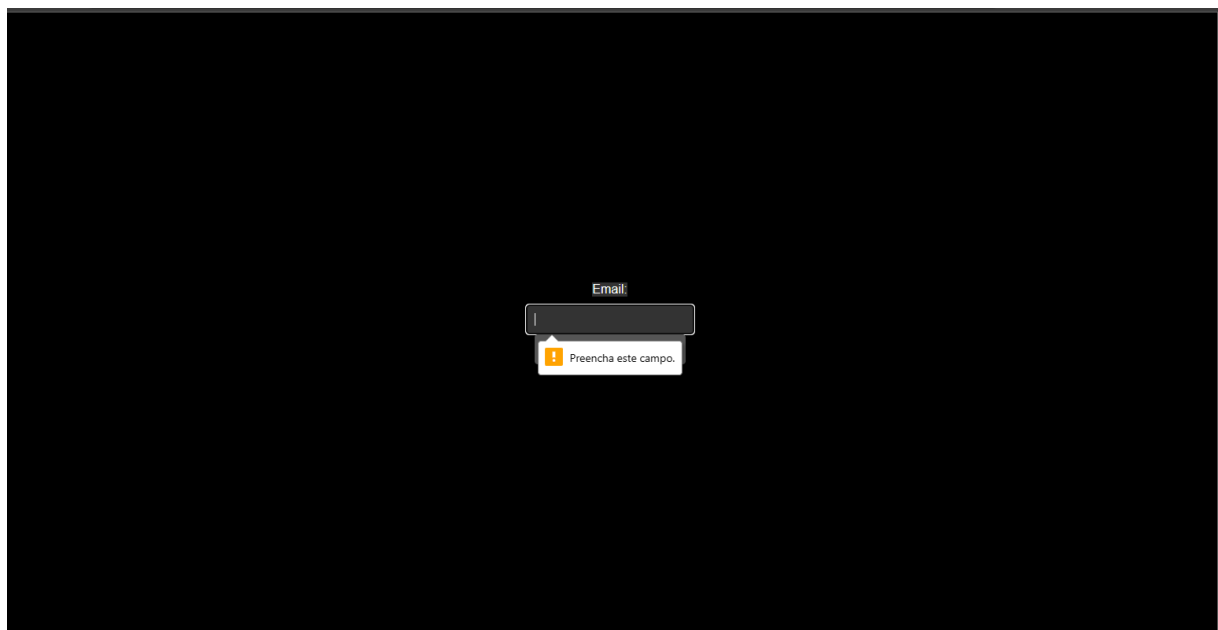


Figura 6 Imagem da página de recuperação pedindo para preencher email.



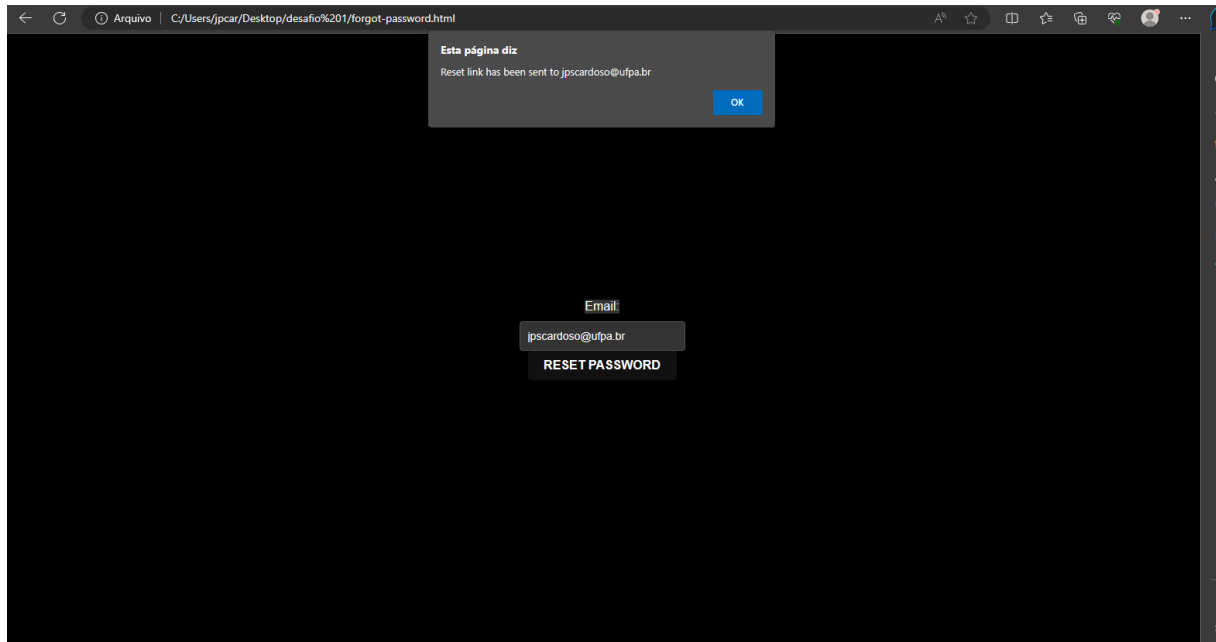


Figura 7 Confirmação de email enviado para recuperação.

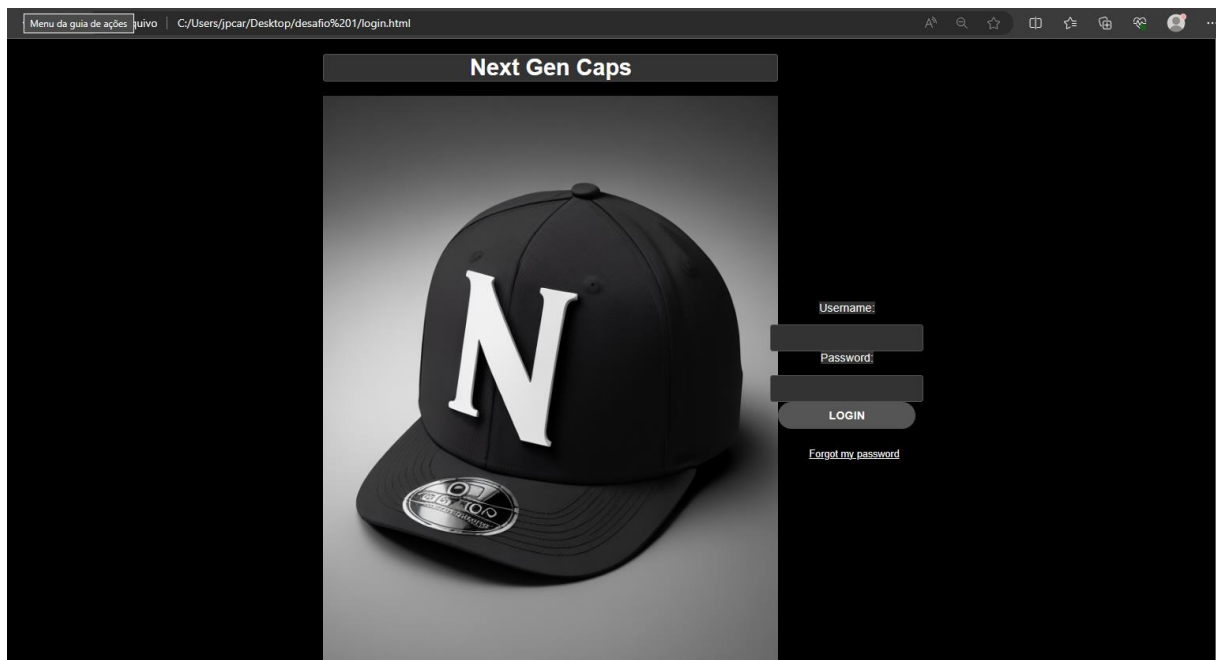


Figura 8 Retorno a página inicial após recuperação de senha.

### 3. CONSIDERAÇÕES FINAIS

As funcionalidades essenciais foram atendidas de forma prática e com certa facilidade, no entanto, alguns ajustes foram necessários para configurar a página, ligar uma a outra na sequencia adequada e com os respectivos arquivos de estilização, embora possa concentrar tudo

em apenas um. A criação das imagens foi relativamente tranquila, sendo que a dificuldade maior foi quanto a geração da logomarca, não foi tão efetivo criar uma logo com apenas as iniciais do nome, por exemplo. De modo geral, o desafio foi interessante e enriquecedor para mostrar a praticidade em se criar aplicações reais e com ganho de produtividade incrível, e com pouco conhecimento sobre as ferramentas de redes generativas.

## **REFERÊNCIAS**

Arquivos no Github

<https://github.com/jpscald/I2A2-Redes-Generativas/tree/main/Desafio%201/Tema%201>

Leonardo

<https://app.leonardo.ai/>