

```

document.addEventListener('DOMContentLoaded', () => {
  const userInput = document.getElementById('user-input');
  const sendButton = document.getElementById('send-button');
  const messagesDisplay = document.getElementById('messages-display');

  sendButton.addEventListener('click', sendMessage);
  userInput.addEventListener('keypress', (e) => {
    if (e.key === 'Enter' && !e.shiftKey) { // Envia ao apertar Enter, a menos que Shift+Enter
      e.preventDefault(); // Evita nova linha no textarea
      sendMessage();
    }
  });

  function sendMessage() {
    const messageText = userInput.value.trim();
    if (messageText === "") return;

    // Adiciona a mensagem do usuário
    addMessage(messageText, 'user');

    // Limpa o input
    userInput.value = "";
    userInput.style.height = 'auto'; // Reseta a altura do textarea

    // Simula uma resposta da IA após um pequeno atraso
    setTimeout(() => {
      const iaResponse = "Essa é uma resposta simulada do Mig para: " + messageText +
        ". ";
      addMessage(iaResponse, 'ia');
    }, 1000); // Responde após 1 segundo
  }

  function addMessage(text, type) {
    const messageDiv = document.createElement('div');
    messageDiv.classList.add('message', `${type}-message`);

    const avatarDiv = document.createElement('div');
    avatarDiv.classList.add('avatar');
    avatarDiv.innerHTML = type === 'user' ? '<i class="fas fa-user"></i>' : '<i class="fas fa-robot"></i>';

    const contentDiv = document.createElement('div');
    contentDiv.classList.add('message-content');

    if (type === 'ia') {
      const senderName = document.createElement('div');
      senderName.classList.add('sender-name');
      senderName.textContent = 'Mig'; // Ou o nome da sua IA
    }
  }

```

```

        contentDiv.appendChild(senderName);
    }

    contentDiv.innerHTML += text; // Adiciona o texto da mensagem (já pode conter
    Markdown simples)

    if (type === 'user') {
        messageDiv.appendChild(contentDiv);
        messageDiv.appendChild(avatarDiv);
    } else {
        messageDiv.appendChild(avatarDiv);
        messageDiv.appendChild(contentDiv);
    }

    messagesDisplay.appendChild(messageDiv);
    messagesDisplay.scrollTop = messagesDisplay.scrollHeight; // Rola para o final
}

// Ajusta a altura do textarea dinamicamente
userInput.addEventListener('input', () => {
    userInput.style.height = 'auto'; // Reseta a altura
    userInput.style.height = userInput.scrollHeight + 'px'; // Ajusta para o conteúdo
});
});

```