

INSTITUTO FEDERAL

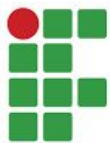
Catarinense

Campus Camboriú

Desenvolvimento Web I

Professor: Euclides Paim

euclides.paim@ifc.edu.br



INSTITUTO FEDERAL

Catarinense

Campus Camboriú

Fundamentos de HTML

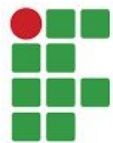
Professor: Euclides Paim

euclides.paim@ifc.edu.br

- **Formulários HTML**
 - Introdução;
 - Projetando o formulário;
 - O elemento form;
 - O elemento input;
 - Campos de texto;
 - O elemento label;
 - Radio buttons;
 - Caixas de seleção;
 - O atributo name;
 - O botão enviar;
 - O atributo action;
 - O atributo method
 - O atributo autocomplete;
 - O atributo target;
 - Tipos de entrada em HTML.



<html>



INSTITUTO FEDERAL
Catarinense
Campus Camboriú

Desenvolvimento Web I

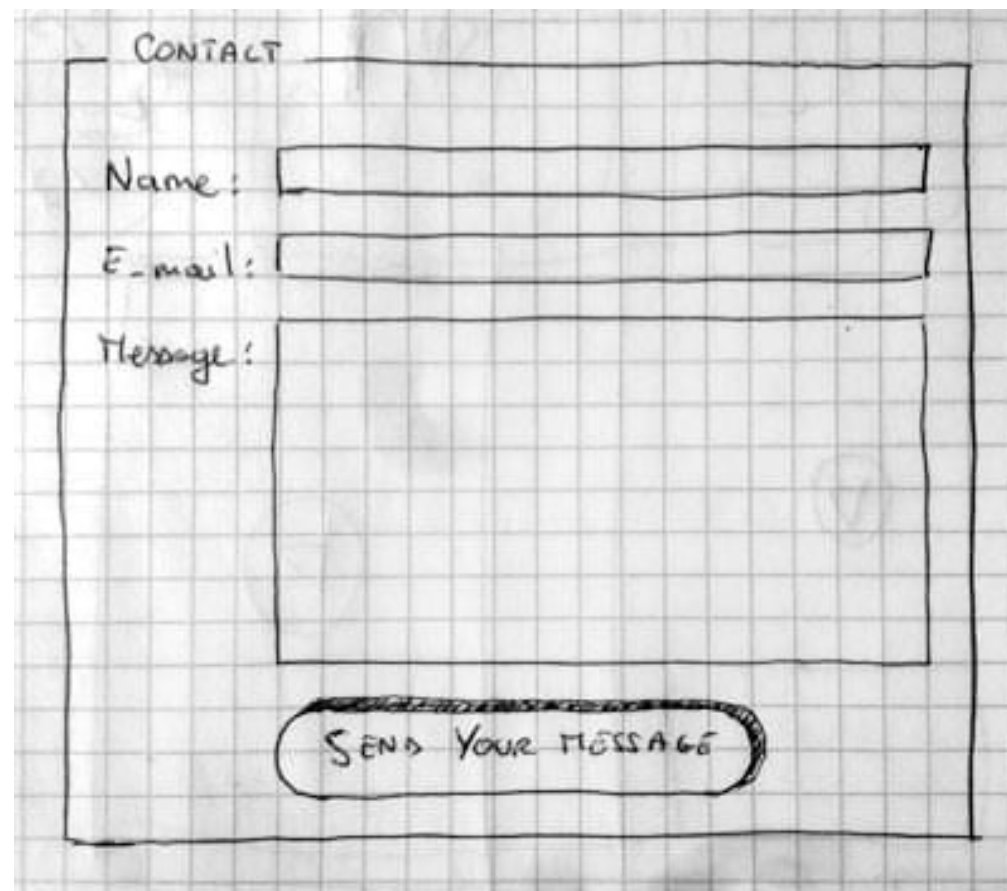


- **Introdução**

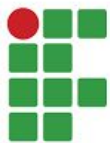
- Os formulários da web são um dos principais pontos de **interação** entre um usuário e um site ou aplicativo da web.
- Os formulários permitem que os usuários insiram dados, que geralmente são enviados a um **servidor web** ou usados no lado do cliente.
- O HTML de um formulário da web é composto de um ou mais **controles de formulário**. Os controles podem ser:
 - Campos de texto,
 - Caixas suspensas,
 - Botões,
 - Caixas de seleção ou botões de opção.
- São geralmente criados usando o elemento **<input>**, embora haja alguns outros elementos para aprender sobre também.
- Os controles de formulário também podem ser programados para **impor formatos** ou valores específicos a serem inseridos (**validação de formulário**).

- **Projetando o formulário**

- Antes de começar a codificar, é sempre melhor dar um passo atrás e **pensar** sobre o seu formulário.
- Projetar uma **maquete** rápida ajudará a definir o conjunto certo de dados que desejamos **solicitar** ao usuário.
- Do ponto de vista da experiência do usuário (UX), é importante lembrar que quanto **maior** for o seu formulário, mais você corre o risco de **frustrar** as pessoas e perder usuários.
- Mantenha a simplicidade e o foco: peça apenas os dados de que você realmente precisa.



A hand-drawn sketch of a contact form on graph paper. The form is titled "CONTACT" at the top. It contains three input fields: "Name:", "E-mail:", and "Message:". The "Message:" field is a larger text area. At the bottom of the form is a button labeled "SEND YOUR MESSAGE".



- O elemento `<form>`
 - O elemento HTML `<form>` é usado para criar um formulário HTML para entrada do usuário:
 - Ele é um contêiner para diferentes tipos de elementos de entrada, como: campos de texto, caixas de seleção, botões de opção, botões de envio, etc.

```
<form>  
  .  
  form elements  
  .  
</form>
```

- **O elemento `<input>`**
 - O elemento HTML `<input>` é o elemento de formulário mais usado. Um elemento `<input>` pode ser exibido de várias maneiras, dependendo do atributo **type**. Aqui estão alguns exemplos.

Tipo	Descrição
<code><input type="text"></code>	Exibe um campo de entrada de texto de linha única.
<code><input type="radio"></code>	Exibe um botão de opção (para selecionar uma das muitas opções).
<code><input type="checkbox"></code>	Exibe uma caixa de seleção (para selecionar zero ou mais de muitas opções).
<code><input type="submit"></code>	Exibe um botão de envio (para enviar o formulário).
<code><input type="button"></code>	Exibe um botão clicável.

- **Campos de texto**

- O **<input type = "text">** define um campo de entrada de linha única para entrada de texto. O valor **"text"** para este atributo também é o valor substituto se o valor que você especificar para o atributo **type** for desconhecido pelo navegador.

```
<form>
  <label for="fname">First name:</label><br>
  <input type="text" id="fname" name="fname"><br>
  <label for="lname">Last name:</label><br>
  <input type="text" id="lname" name="lname">
</form>
```

Obs.: O **form** em si não é visível. Observe também que a largura padrão de um campo de entrada é de **20** caracteres.

- O elemento **<label>**

- A tag **<label>** define um rótulo para muitos elementos de formulários. O elemento **<label>** é útil para usuários de leitores de tela, porque o leitor de tela lerá em voz alta o rótulo quando o usuário focar no elemento de entrada.
- Ele também ajuda os usuários que têm dificuldade em clicar em regiões muito pequenas (como botões de rádio ou caixas de seleção) - porque quando o usuário clica no **texto** dentro do elemento **<label>**, ele seleciona a caixa em questão.
- O atributo **for** da tag **<label>** deve ser igual ao atributo **id** do elemento **<input>** para vinculá-los.

```
<form>
  <label for="fname">First name:</label><br>
  <input type="text" id="fname" name="fname"><br>
  <label for="lname">Last name:</label><br>
  <input type="text" id="lname" name="lname">
</form>
```

- **Radio Buttons**

- O `<input type = "radio">` define um botão de opção (**radio button**). Os botões de opção permitem que o usuário selecione **UM** de um número limitado de opções..

```
<form>
  <input type="radio" id="male" name="gender" value="male">
  <label for="male">Male</label><br>
  <input type="radio" id="female" name="gender" value="female">
  <label for="female">Female</label><br>
  <input type="radio" id="other" name="gender" value="other">
  <label for="other">Other</label>
</form>
```

- **Caixas de seleção**
 - O **<input type = "checkbox">** define uma caixa de seleção. As caixas de seleção permitem que o usuário selecione ZERO ou MAIS opções de um número limitado de escolhas.

```
<form>
  <input type="checkbox" id="vehicle1" name="vehicle1" value="Bike">
  <label for="vehicle1"> I have a bike</label><br>
  <input type="checkbox" id="vehicle2" name="vehicle2" value="Car">
  <label for="vehicle2"> I have a car</label><br>
  <input type="checkbox" id="vehicle3" name="vehicle3" value="Boat">
  <label for="vehicle3"> I have a boat</label>
</form>
```

- **O atributo de name**
 - Observe que cada campo de entrada deve ter um atributo de nome para ser enviado. Se o atributo **name** for omitido, o valor do campo de entrada não será enviado.

- **O botão Enviar**

- O `<input type = "submit">` define um botão para enviar os dados do formulário a um **manipulador de formulários**. O manipulador de formulários é normalmente um arquivo **no servidor** com um *script* para processar dados de entrada. O manipulador de formulários é especificado no atributo **action** do formulário.

```
<form action="/action_page.php">  
  <label for="fname">First name:</label><br>  
  <input type="text" id="fname" name="fname" value="John"><br>  
  <label for="lname">Last name:</label><br>  
  <input type="text" id="lname" name="lname" value="Doe"><br><br>  
  <input type="submit" value="Submit">  
</form>
```

- **O atributo de action**

- O atributo **action** define a ação a ser executada quando o formulário é enviado. Normalmente, os dados do formulário são enviados para um arquivo no servidor quando o usuário clica no botão enviar. No exemplo anterior, os dados do formulário são enviados para um arquivo denominado *"action_page.php"*. Este arquivo contém um script do lado do servidor que lida com os dados do formulário.

- **O atributo de method**

- O atributo do **method** especifica o método HTTP a ser usado ao enviar os dados do formulário. Os dados do formulário podem ser enviados como variáveis de URL (com **method** = "**get**") ou como transação HTTP post (com **method** = "**post**"). O método HTTP padrão ao enviar dados do formulário é GET.
- Notas sobre **GET**:
 - anexa os dados do formulário ao URL, em pares nome-valor
 - **NUNCA** use GET para enviar dados confidenciais! (os dados enviados são visíveis no URL!)
 - O comprimento de um URL é limitado (2.048 caracteres);
 - Útil para envios de formulários onde um usuário deseja marcar o resultado
 - GET é bom para dados não seguros, como strings de consulta no Google.
- Notas no **POST**:
 - Acrescenta os dados do formulário dentro do corpo da solicitação HTTP (os dados enviados não são mostrados no URL)
 - O POST não tem limitações de tamanho e pode ser usado para enviar grandes quantidades de dados.
 - Os envios de formulários com POST não podem ser marcados.

- **O atributo de autocomplete**

- O atributo **autocomplete** especifica se um formulário deve ter o preenchimento automático ativado ou desativado. Quando o preenchimento automático está ativado, o **navegador** completa os valores automaticamente com base nos valores que o usuário inseriu anteriormente.

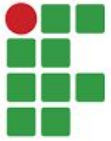
- **O Atributo de target**

- O atributo target especifica onde exibir a resposta recebida após o envio do formulário. O atributo target pode ter um dos seguintes valores:
 - ***_blank*** → A resposta é apresentada em uma nova guia ou janela;
 - ***_self*** → A resposta é apresentada na mesma guia ou janela;
 - ***_parent*** → A resposta é apresentada no quadro pai;
 - ***_top*** → A resposta é apresentada no corpo todo da janela;
 - ***framename*** → A resposta é apresentada no *iframe* identificado;

- **HTML input types**

- Aqui estão os diferentes tipos (**types**) de entrada (**input**) que podemos usar em HTML:

- `<input type="button">`
- `<input type="checkbox">`
- `<input type="color">`
- `<input type="date">`
- `<input type="datetime-local">`
- `<input type="email">`
- `<input type="file">`
- `<input type="hidden">`
- `<input type="image">`
- `<input type="month">`
- `<input type="number">`
- `<input type="password">`
- `<input type="radio">`
- `<input type="range">`
- `<input type="reset">`
- `<input type="search">`
- `<input type="submit">`
- `<input type="tel">`
- `<input type="text">`
- `<input type="time">`
- `<input type="url">`
- `<input type="week">`



Desenvolvimento Web I

Resumo

- **Formulários HTML**
 - Introdução;
 - Projetando o formulário;
 - O elemento form;
 - O elemento input;
 - Campos de texto;
 - O elemento label;
 - Radio buttons;
 - Caixas de seleção;
 - O atributo name;
 - O botão enviar;
 - O atributo action;
 - O atributo method
 - O atributo autocomplete;
 - O atributo target;
 - Tipos de entrada em HTML.





Referências Básicas

SILVA, Maurício Samy. CSS3: desenvolva aplicações web profissionais com uso dos poderosos recursos de estilização das CSS3. São Paulo: Novatec, 2012.

SILVA, Maurício Samy. HTML 5: a linguagem de marcação que revolucionou a web. São Paulo: Novatec, 2011.

NIEDERAUER, Juliano. Desenvolvendo websites com PHP: aprenda a criar websites dinâmicos e interativos com PHP e bancos de dados. 2. ed. rev. e atual. São Paulo: Novatec, 2011.

Referências Complementares

FLANAGAN, David. **o guia definitivo**. . O Really. 2012

SILVA, Maurício Samy. **Criando sites com HTML: sites de alta qualidade com HTML e CSS**. . Novatec. 2010

SOARES, Wallace. **PHP 5: conceitos, programação e integração com banco de dados**. . Érica. 2010

DALL'OGGIO, Pablo. **PHP: programando com orientação a objetos**. . Novatec. 2009

DEITEL, Paul J. Ajax,. **Rich Internet applications e desenvolvimento Web para programadores**. . Pearson Prentice Hall. 2009

IEPSEN, Edécio Fernandes. **Lógica de Programação e Algoritmos com JavaScript**. Novatec. 2018.

Referências na Internet

<https://www.w3schools.com>

<https://developer.mozilla.org/pt-BR/docs/Web>

<https://illustrated.dev/advancedjs>