

# Formulários e Validação

Usabilidade, desenvolvimento web, mobile e jogos

Prof. Me. Gustavo Torres Custódio  
gustavo.custodio@anhembi.br

# Conteúdo

## Formulários



Formulários e Validação

# Formulários

# Formulários

- Um formulário (<form>) HTML permite que um usuário insira informações e envie.
  - O <form> é enviado para um destino, chamado form-handler.
  - E o form-handler é definido pelo atributo action.
    - Exemplo: <form action="/action\_form">

## Enviando um Formulário

- O form-handler é normalmente um arquivo no servidor com um *script* para processar inputs.
  - Em nosso caso, será um servidor rodando uma aplicação Flask.

## Enviando um Formulário

- Outro atributo que é definido em um `<form>` é o `method`.
- Este atributo define qual método HTTP será utilizado para enviar o formulário.
  - Se não definirmos nenhum `method`, por padrão, será utilizado o GET.
  - Para envio de formulários, normalmente o método de envio utilizado é o POST.

# GET e POST

- GET:
  - Número máximo de caracteres.
  - Dados enviados na URL.
- POST:
  - Sem restrição de tamanho.
  - Dados enviados ficam no corpo da requisição.

## Enviando um Formulário

- Vamos construir uma aplicação em Flask que recebe um formulário
  - e envia de volta para um usuário uma tabela contendo os dados do formulário em letras maiúsculas.
- Primeiro vamos criar o formulário e criar uma URL para acessá-lo.



## Enviando um Formulário

- Em uma pasta templates, crie o arquivo formulario.html

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="UTF-8">
  <title>Meu formulário</title>
  <style type="text/css">
    form input {
      margin-top: 10px;
      margin-bottom: 10px;
    }
  </style>
</head>
```

# Enviando um Formulário

```
<body>
  <h1>Meu formulário</h1>
  <form action="/processar" method="POST">
    <fieldset>
      <legend>Formulário de Cadastro</legend>
      <input type="text" name="nome"
        placeholder="Digite seu nome" required> <br>
      <input type="password" name="senha"
        placeholder="Digite uma senha" required> <br>
      <input type="email" name="email"
        placeholder="E-mail" required>
    </fieldset>
    <input type="submit" value="Submeter Dados">
  </form>
</body>
</html>
```

## Enviando um Formulário

- Observe a adição do atributo `required`.
  - Isso define o campo como obrigatório.
- Agora criamos um arquivo para o servidor Flask chamado `backend.py`.
  - E dentro dele, criamos uma rota para o formulário que criamos.

## Enviando um Formulário

```
from flask import Flask, render_template, request

app = Flask(__name__)
app.config["TEMPLATES_AUTO_RELOAD"] = True

@app.route('/formulario')
def abrirform():
    return render_template('formulario.html')
```

## Enviando um Formulário

- localhost:5000/formulario:

### Meu formulário

Formulário de Cadastro

## Enviando um Formulário

- Agora criamos uma rota para /processar.
  - No entanto, essa rota vai tratar uma requisição HTTP do tipo POST
    - não GET, como fizemos até o momento.

## Enviando um Formulário

```
@app.route('/processar', methods = ['POST',])
def criar_tabela_dados():
    nome = request.form["nome"]
    senha = request.form["senha"]
    email = request.form["email"]
    return f"""<table>
        <tr><td>Nome</td><td>{nome.upper()}</td></tr>
        <tr><td>Senha</td><td>{senha.upper()}</td></tr>
        <tr><td>E-mail</td><td>{email.upper()}</td></tr>
    </table>"""
```

## Enviando um Formulário

Nome JOÃO

Senha 12345678

E-mail JOAO123@EMAIL.COM

- O servidor nos manda uma resposta contendo o conteúdo do formulário em uma tabela.



## Referências

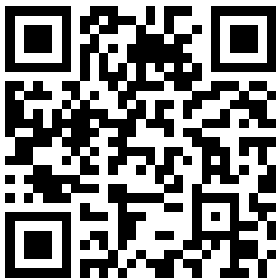


Paul, S. (2020).

Read html form data using get and post method in node.js.

<https://medium.com/swlh/read-html-form-data-using-get-and-post-method-in-node-js-8d2c7880adbf>.

## Conteúdo



<https://gustavotcustodio.github.io/usabilidade.html>

Obrigado

[gustavo.custodio@anhembi.br](mailto:gustavo.custodio@anhembi.br)