

Programação em Python

UC2: Desenvolver aplicações front-end para web

Sobre o curso

- Web Designer x Front-end
- Wireframe e protótipos
- Linguagem de Marcação (HTML)
- Linguagem de Estilo (CSS)
- javascript
- Bootstrap
- Principais Frameworks front-end

Aula 01

- arquitetura web (Básico)
- Web Designer x Front-end
- navegadores
- Editores de código
- instalar o VSCode e a extensão Live Server
- Linguagem de Marcação (HTML)
- Exemplo: montar uma página exibindo a mensagem “Olá, Mundo!”
- Principais Frameworks front-end

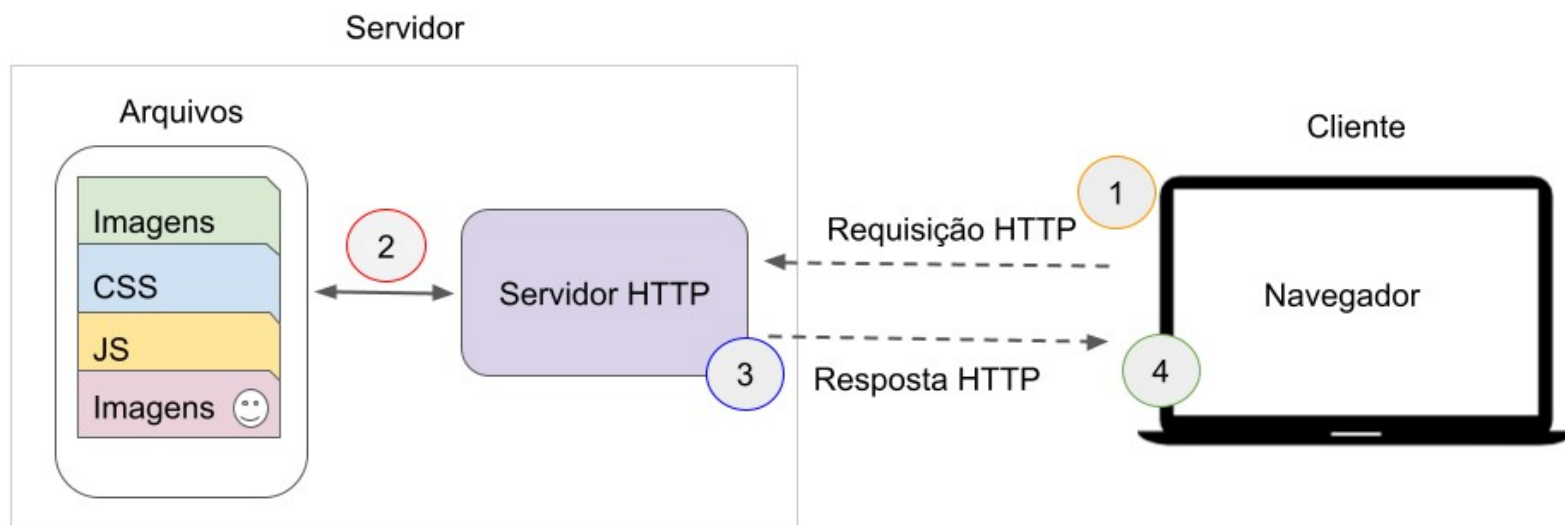
Arquitetura Web

A arquitetura de aplicações web descreve a estrutura interna e interações entre seus componentes, bancos de dados e sistemas externos. A arquitetura da web é por natureza distribuída, baseada no modelo cliente-servidor. Chamamos de clientes os dispositivos de acesso as páginas web por meio de um navegador.

Utilizando o modelo cliente-servidor é possível estruturar uma aplicação web de diferentes maneiras, dependendo do tamanho, tecnologias utilizadas e quantidade de acessos.

UC2: Desenvolver aplicações front-end para web

Aplicações Estáticas

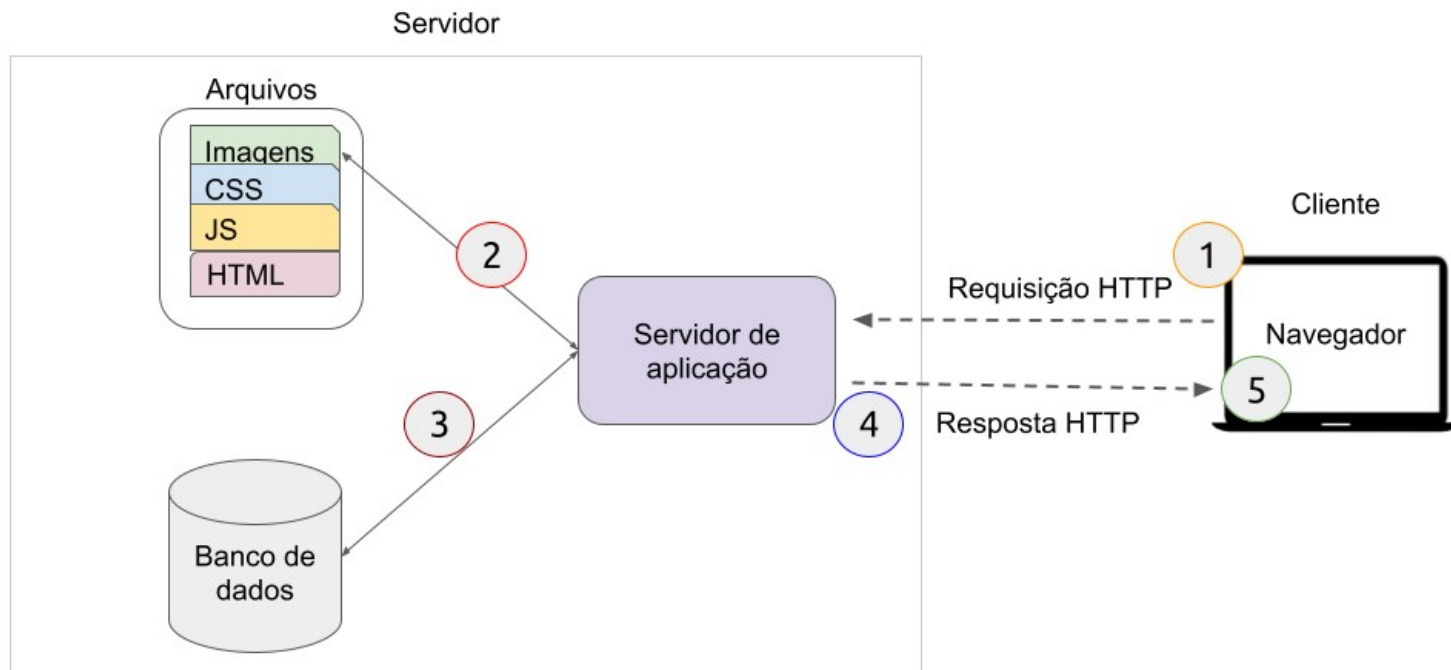


Aplicações Estáticas

- Site estático é o modelo mais simples de aplicação web, construído com HTML, CSS e Javascript (opcional).
- Um site estático é aquele que retorna o mesmo conteúdo que está salvo no servidor sempre que uma determinada página é solicitada.

UC2: Desenvolver aplicações front-end para web

Aplicações Dinâmicas

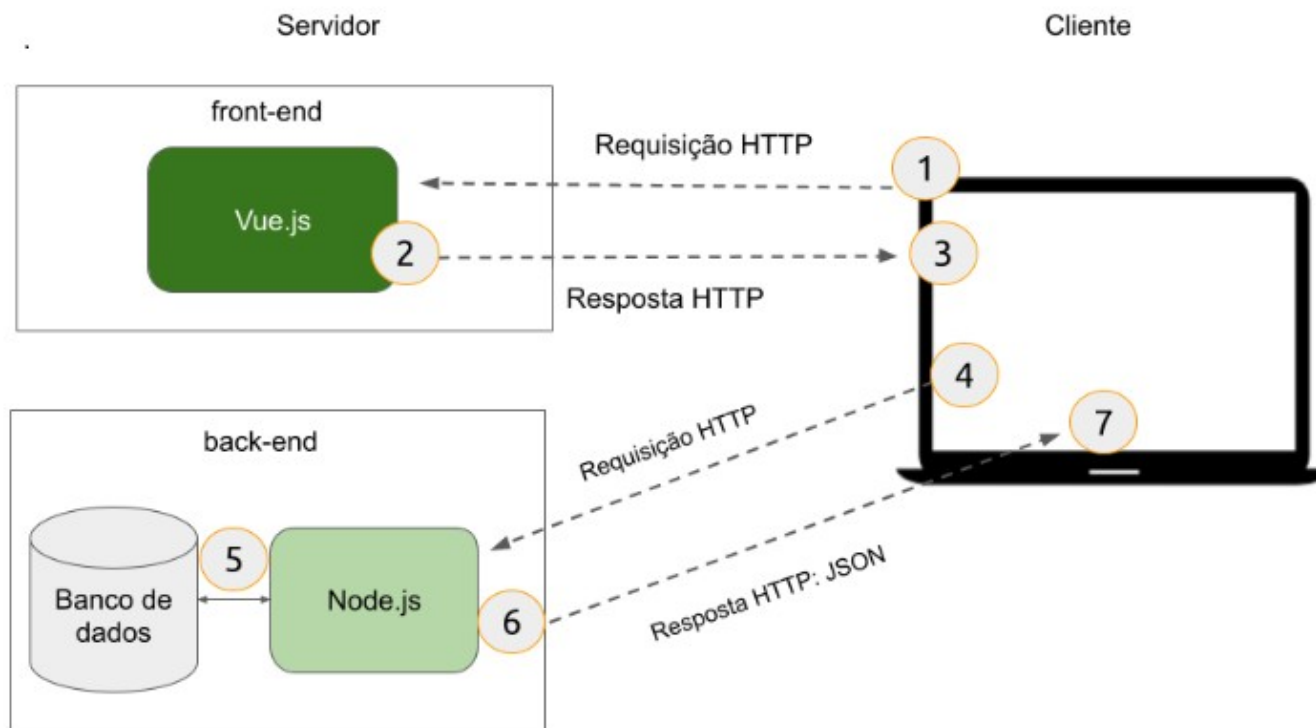


Aplicações Dinâmicas

- Sites dinâmicos são aqueles que geram dinamicamente parte do conteúdo que é retornado para o cliente.
- conteúdo dinâmico geralmente é armazenado em um banco de dados.
- Front-end pode estar num servidor diferente do servidor com back-end.
- Sistemas web são exemplos de aplicações dinâmicas

UC2: Desenvolver aplicações front-end para web

Aplicações Dinâmicas



Web Designer x Front-end

O profissional desenvolvedor front-end é o responsável por colocar em prática, através de códigos, o design de um site ou interface. Enquanto o web design projeta o visual de um site, o desenvolvimento front-end implementa esse design através de códigos, como HTML, CSS e JavaScript.

- Web Designer veio no início da Web 1.0 que eram páginas estáticas e imagens tratadas faziam a diferença.
- Web design é uma espécie de Designer para web.
- Front-end nasce com web 2.0, onde as páginas são mais dinâmicas e com possibilidade de mais interação do usuário.
- Front-end é um programador que trabalha com a interface gráfica.

Web Designer

- Executa o tratamento de imagens para inclusão no site, criação de banners para o site e para campanhas de trade marketing.
- Cria graficamente o visual da página das aplicações com imagens e textos., desenvolve a navegação da página.
- Cria banners, folders, cartões e gifs animados com identidade visual.

Desenvolvedor Front-end

Enquanto o web design projeta o visual de um site, o desenvolvimento front-end implementa esse design através de códigos, como HTML, CSS e JavaScript. Ou seja, é o frontend que coloca “para funcionar” toda a parte visual de uma interface, como a página de um site ou de um aplicativo.

Web Designer x Front-end

Mercado - Salários:

- Web Designer - **R\$ 2.968,82 (Analista)**
- Desenvolvedor Front-End: **R\$ 4.814,00 (Média)**

Ambiente Web

- Navegadores (Chrome, Edge, Firefox, Brave, Safari, Opera, ...)
- Editor de código (Sublime Text, VsCode, Notepad++,...)
- Servidor

Usaremos o VsCode como editor com extensão Live Server para usar como servidor

Outra opção é abrir o arquivo chamando o navegador.

UC2: Desenvolver aplicações front-end para web



Ambiente Web

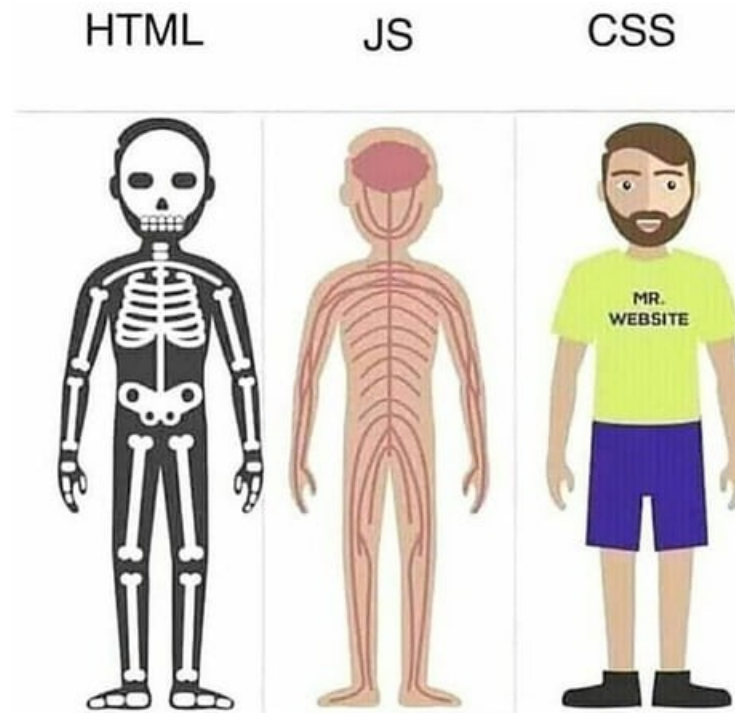
Download do Vs Code: <https://code.visualstudio.com/download>

HTML, CSS e javaScript

- **HTML** - Linguagem de marcação (Esqueleto da página).
- **CSS** - Tudo aquilo que é relacionado ao estilo da página.
- **javaScript** - Linguagem de programação que manipula a página.

UC2: Desenvolver aplicações front-end para web

HTML, CSS e javaScript



HTML

HTML

HTML é a sigla de Hyper-Text Markup Language, ou Linguagem de Marcação de Hipertexto em português.

Cria uma estrutura de blocos onde serão inseridos os conteúdos de uma página.

Estes blocos são em formato de tag <>, com finalidades específicas.

O documento HTML é lido de cima para baixo.

UC2: Desenvolver aplicações front-end para web



HTML

```
<!DOCTYPE html>

<html>

<head>

<title>Página HTML</title>

</head>

<body>

<h1>Meu primeiro título</h1>

<p>Meu primeiro parágrafo.</p>

</body>

</html>
```

HTML

- `<!DOCTYPE html>` declaração que o documento é um documento HTML5
- `<html>` é o elemento raiz de uma página HTML
- `<head>` elemento que contém as metas informações sobre a página HTML
- `<title>` elemento que especifica um título para a página HTML (aparece na barra de título do navegador)
- `<body>` elemento corpo do documento, contém toda a área visível em tela como imagens, títulos, parágrafos, tabelas, listas, hiperlinks e etc.
- `<h1>` elemento que define um cabeçalho destacado(título maior)
- `<p>` elemento que define um parágrafo.

HTML

Primeira atividade:

Criar um arquivo chamado index.html:

- criar uma estrutura html
- título da página será "Meu primeiro HTML"
- No corpo da página, colocar o título "Olá, Mundo!!!"

HTML

TAG

são elemento envolvidos com <> que tem um objetivo dentro do HTML.
São abertas por <tag> e fecha com </tag>.

exemplo: <h1>Título</h1>

Algumas são fechadas nelas mesma.

**exemplo: **

As tags possuem atributos que auxiliam nas suas funções.

**exemplo: **

HTML

TAG

Principais tags

- **headers**: possui um cabeçalho para títulos que vai de 1 a 6 (h1,h2,...,h6).
- `<p>`: tag específica para parágrafo.
- `<a>`: tag de link para acessar outras página. Usa o atributo href para apontar o endereço do link.
- ``: tag de inserção de imagem. Necessita do atributo **src** para apontar o local que a imagem está armazenada.

OBS: Uma tag pode usada dentro de outra tag

HTML

TAG

Atividade

- Adicionar ao exercício anterior, 3 imagens com links, cada uma, para os sites do Google, Amazon e Senac.

HTML

TAG

Semântica - a tag possui especificação para uso.

ex: <p> é usada para parágrafo.

Não semântica - tag genérica que pode ser usada em qualquer situação.

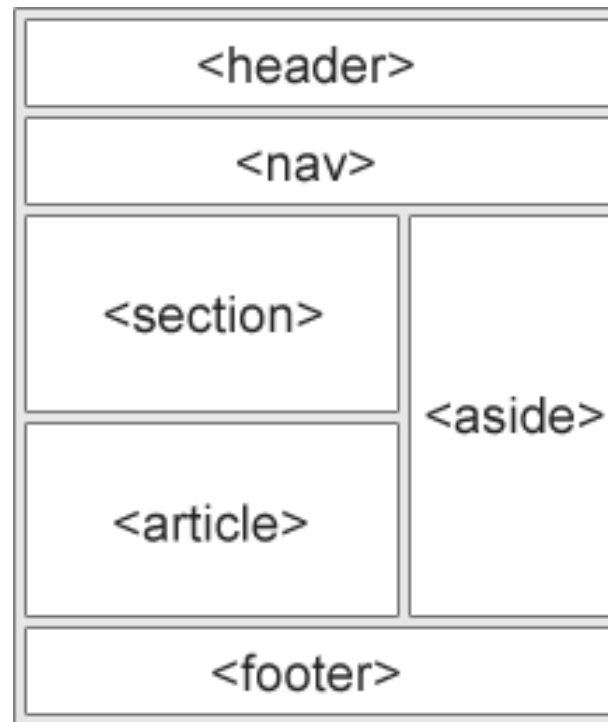
ex: <div> e

HTML

TAG Semântica

- `<article>` - artigos
- `<aside>` - barra lateral
- `<details>` - detalhes
- `<figcaption>` - texto de imagem
- `<figure>` - figura
- `<footer>` - rodapé
- `<header>` - cabeçalho
- `<main>` - tag principal
- `<mark>` - marcação
- `<nav>` - navegação
- `<section>` - seção
- `<summary>` - sumário
- `<time>` - tempo

HTML TAG Semântica



Até a próxima...

