Interface

homem-máquina

Inatel | Engenharia de Software

S205 - Interface Homem Máquina

Prof. Raphael C. M. Pereira

Módulos da Disciplina

Habilidades desenvolvidas em laboratório.

Roteiro da Aula

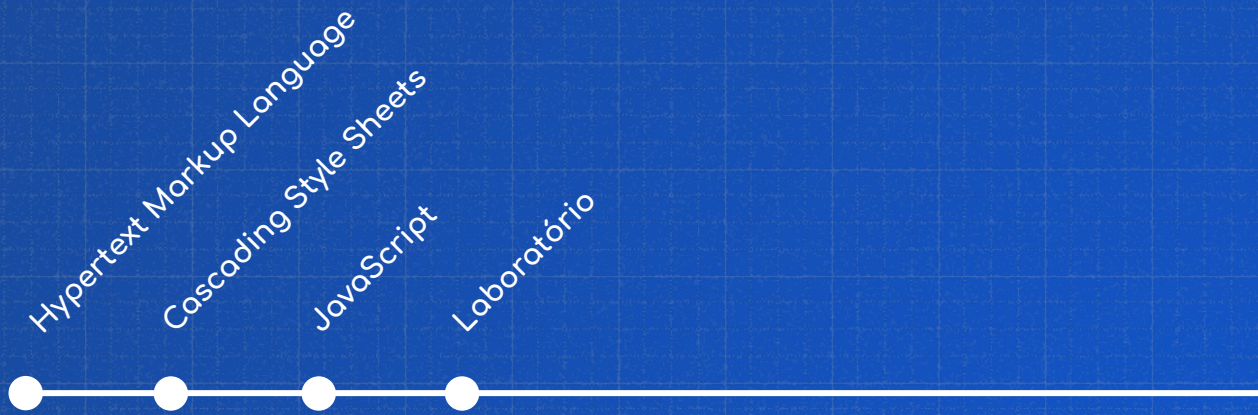
Principios de Design

- ✓ HTML, CSS & JS
- ✓ Grid e Responsividade
- ✓ Cores e Tipografia
- ✓ Imagens e Vídeos
- ✓ Interação e Animação
- ✓ Componentes
- ✓ Formulários e Visualização de Dados
- ✓ Testes de Usabilidade

HTML, CSS & JS.

HTML, CSS & JS.

Linha de Pensamento



HTML, CSS & JS

HTML, CSS e JS formam o trio fundamental da criação de páginas web. HTML (Hypertext Markup Language) define a estrutura e conteúdo, CSS (Cascading Style Sheets) controla a apresentação e estilo visual, enquanto JS (JavaScript) adiciona interatividade dinâmica e comportamento à página. Juntos, essas linguagens possibilitam a construção de experiências web ricas e envolventes.

HTML - HyperText Markup Language

O HTML define a estrutura e o layout de um documento da web usando uma variedade de **elementos (tags)** e **atributos** para formatar texto, inserir imagens, criar links, construir tabelas e muito mais. Ele fornece os blocos de construção básicos que os navegadores da web usam para renderizar conteúdo na internet. Os documentos HTML são interpretados pelos navegadores da web para exibir páginas da web aos usuários.

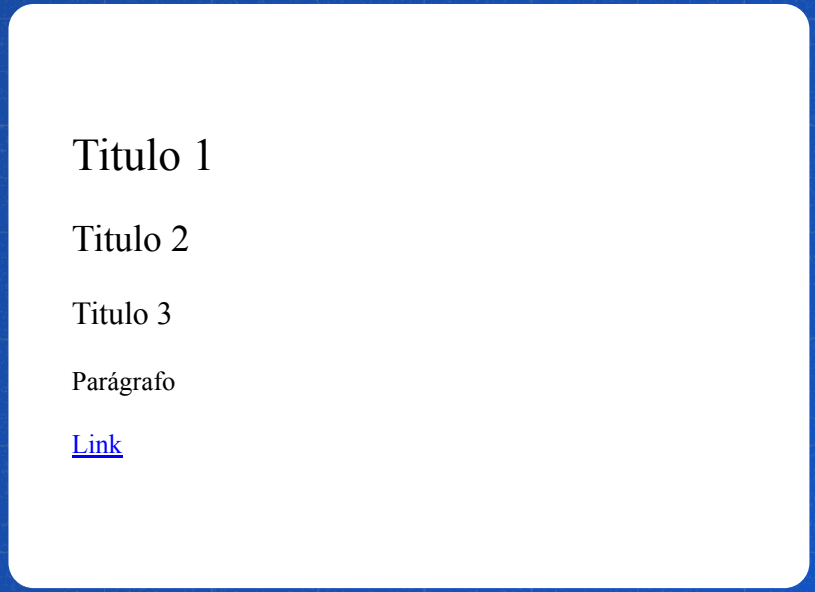
```
<html>
  <head>
    <meta>
    <title> título da página </title>
    <link> scripts, css… </link>
  </head>
  <body>
    <p> parágrafo </p>
    <img> imagem </img>
    <div> conteudos </div>
    <a> javascript </a>
  </body>
</html>
```

Categ.: heading embedded flow interactive

Estrutura do HTML

A estrutura básica do HTML consiste em elementos essenciais que organizam o conteúdo da página. Inicia-se com o elemento <html> que engloba todo o conteúdo. O <head> contém metadados, título da página e referências a scripts e estilos. O <body> inclui o conteúdo visível, como textos, imagens e links. Esses elementos, junto com outros como <title>, formam a estrutura fundamental do HTML, definindo a hierarquia e organização da página da web.

```html
<html>
  <head>
    <meta>
    <title> S205 - Exemplo Html </title>
  </head>
  <body>
    <h1> Titulo 1 </h1>
    <h2> Titulo 2 </h2>
    <h3> Titulo 3 </h3>
    <p> Parágrafo </p>
    <a><p> Link </p></a>
  </body>
</html>
```



Exemplos de tags de conteúdo:

`<header>`	Geralmente contém o cabeçalho da página, como logotipo e navegação.
`<main>`	Contém o conteúdo principal da página.
`<article>`	Contém um conteúdo independente e autocontido, como uma postagem de blog.
`<footer>`	Contém informações de rodapé, como direitos autorais e links de contato.

Elementos de Grupo de Conteúdo

A tag é um elemento genérico usado para agrupar e organizar conteúdo em uma página web, e não possui significado semântico próprio. Por outro lado, existem variações especializadas para organizar conteúdo, fornecendo significado semântico claro para o conteúdo dentro delas, como cabeçalho, navegação e conteúdo principal.

Exemplos de tags de texto:

```
<h1>  Títulos de diferentes níveis
      de importância.
<p>   Parágrafos de texto.
<ul>  Lista não ordenada.
<ol>  Lista ordenada.
<li>  Elemento de lista em <ul> ou <ol>.
<dl>  Lista de definição.
<dt>  Termo em uma lista de definição.
<dd>  Definição em uma lista de definição.
```

Link da documentação com todas as tags.

Elementos de Texto

As tags de texto são elementos usados para estruturar e formatar conteúdo textual em uma página da web. Elas são utilizadas para definir o tipo de texto (como parágrafos, títulos, listas etc.) e aplicar formatação (como negrito, itálico, sublinhado etc.), permitindo assim a criação e organização de conteúdo visualmente atraente e legível.

Exemplos de tags de mídia:

```
<img>     Inserção de imagens.
<audio>   Reprodução de áudio.
<video>   Reprodução de vídeo.
<iframe>  Incorporação de conteúdo de
          outros sites.
```

Link da documentação com todas as tags.

Elementos de Mídia

As tags de multimídia são elementos utilizados para incorporar e controlar conteúdo multimídia, como vídeos e áudios, em páginas da web. As tags mais comuns são <video> e <audio>, que permitem especificar fontes de mídia, controles de reprodução e outras propriedades para oferecer uma experiência interativa ao usuário.

```
Exemplo da estrutura da tag
<NomeDaTag> Conteúdo </NomeDaTag>

Exemplo da estrutura da tag com atributos
<NomeDaTag Atributo1="valor1"
Atributo2="valor2"> Conteúdo </NomeDaTag>

Exemplo de tag DIV com atributos
<div id="container" class="content"
style="background-color: #f0f0f0; padding:
20px;"> Conteúdo da div aqui </div>

Link da documentação com todas as tags.
```

Estrutura do Elemento

A estrutura básica de uma tag HTML consiste em um nome de tag entre colchetes angulares. Além disso, podem conter atributos dentro do par de colchetes. Os atributos fornecem informações adicionais sobre o elemento. Alguns atributos comuns incluem "id" para identificação única, "class" para aplicação de estilos em CSS, "src" para especificar a origem de elementos como imagens e "href" para links. Há também atributos específicos para acessibilidade, como "alt" para descrever imagens.

Exemplo dos atributos de um link

```
<a href="https://www.example.com"
   target="_blank"
   rel="noopener noreferrer"
   media="screen"
   hreflang="en"
   type="text/html">
   Link Exemplo
</a>
```

Link da documentação com todas as tags.

Atributos

Os atributos em elementos HTML fornecem informações adicionais ou comportamentos específicos. Por exemplo, o atributo rel="noopener noreferrer" é usado em links <a> para melhorar a segurança ao abrir páginas externas em uma nova aba (target="_blank").

Noopener: evita que a página externa acesse o contexto da página atual, ajudando a prevenir ataques de janela (window.opener). Noreferrer: oculta o URL de referência (a URL da página atual) do cabeçalho HTTP enviado para a página externa, protegendo a privacidade do usuário.

CSS

CSS (Cascading Style Sheets) é uma linguagem de estilo usada para controlar a apresentação e o design de páginas web. Com CSS, você pode definir cores, fontes, layout e outros estilos visuais para tornar o conteúdo HTML mais atraente e acessível. Ele separa a estrutura (HTML) do estilo, permitindo maior flexibilidade e consistência no design da web.

```css
/* Exemplo da estrutura css */
seletor {
    propriedade: valor;
}

/* Exemplo da estrutura css de título */
h1 {
    color: blue;
    font-size: 24px;
}
```

Link da documentação com todas os estilos.

Estrutura do CSS

A estrutura do CSS segue um padrão básico que envolve a seleção de elementos HTML e a aplicação de estilos a eles. Os principais componentes incluem: **Seletor:** Identifica o elemento HTML ao qual o estilo será aplicado. **Propriedade:** Define a característica visual a ser modificada, como cor, tamanho ou posição. **Valor:** Especifica o valor da propriedade a ser aplicada ao seletor. Juntos, esses componentes compõem uma regra de estilo.

```
<div id=”head” class=”red”>
  <p> conteúdo </p> </div>

div {background-color: red;}
.red {background-color: red;}
div.red {background-color: red;}
#head {background-color: blue !important;}
.red > p {font-size: 24px;}

<a href="https://www.inatel.com"
rel="nofollow">link</a>

a[rel="nofollow"] {font-size: 24px;}
```

Seletores

Seletores são padrões usados para identificar quais elementos HTML devem receber estilos. Eles podem ser baseados em tipos de elementos (por exemplo, p para parágrafos), classes (por exemplo, .classe), IDs (por exemplo, #identificador), atributos (por exemplo, [type="text"]) ou relações entre elementos (por exemplo, div > p para parágrafos dentro de divs). Os seletores ajudam a aplicar estilos de forma seletiva e eficiente a elementos específicos em uma página da web.

```
/* inline: */
<div style="background-color: #f0f0f0;
padding: 20px;"></div>

/* externo ou interno */
<head>
<link rel="stylesheet" href="estilos.css">
<style>
@import
url('https://fonts.googleapis.com/css2?fami
ly=Roboto:wght@400;700&display=swap');
</style>
</head>
```

Vinculação do CSS

As formas de aplicar o CSS incluem: Inline (diretamente nos elementos com style), interno (no <style> dentro do <head>), externo (em arquivo CSS vinculado via <link>), e importado (usando @import dentro do <style>).

JavaScript

O JavaScript é uma linguagem de programação utilizada para tornar páginas web interativas e dinâmicas. Executa ações no navegador do usuário, como manipulação de elementos HTML, interações de formulários, requisições de rede assíncronas (AJAX), animações, validação de dados e muito mais.

Características

O JavaScript é considerado uma linguagem de **alto nível** por sua abstração de detalhes de baixo nível, permitindo uma sintaxe mais legível e próxima da linguagem humana. É **dinâmica** porque suporta alterações durante a execução, como adicionar novas propriedades a objetos. **Interpretada** significa que o código é executado diretamente por um interpretador no navegador, sem necessidade de compilação. É **multiplataforma** porque pode ser executado em qualquer navegador ou ambiente que suporte JavaScript, independentemente do sistema operacional.

```
// inline
<button onclick="minhaFuncao()">
Clique Aqui </button>

<script>
    function minhaFuncao() {
        alert('Olá, mundo!');
    }
</script>

// externo
<script src="meu-script.js"></script>
```

Vinculação do JS

As formas de aplicar o JS pode ser inline, no corpo do html, dentro do elemento <head> ou <body>. E também pode ser vinculado através de um arquivo JS externo.

De acordo com a documentação, colocar scripts na parte inferior do elemento <body> melhora a velocidade de exibição, porque a interpretação do script retarda a exibição.

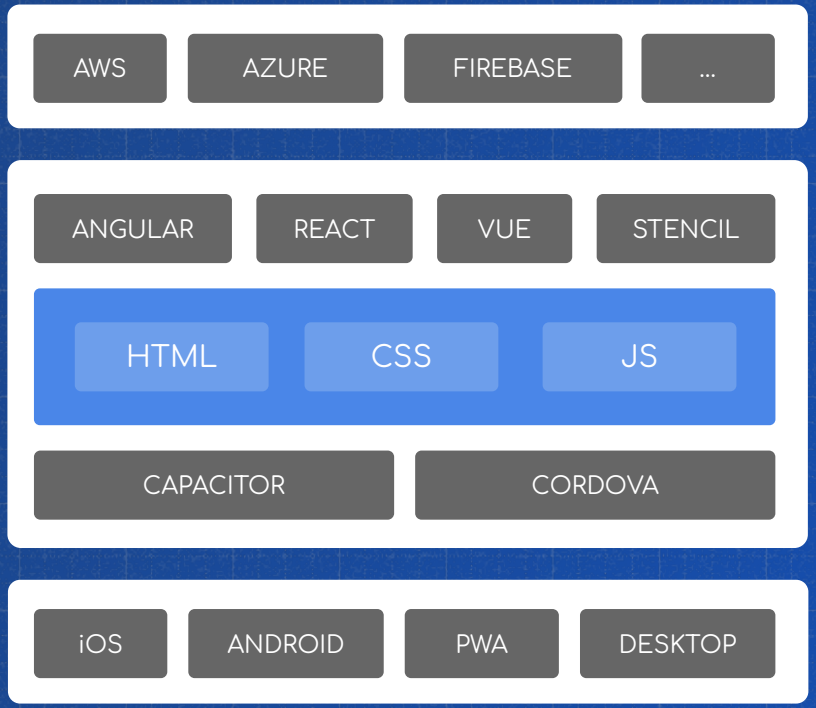
```
// mudar um conteúdo html
document.getElementById("demo").
innerHTML = "Hello JavaScript";

// mudar um conteúdo html
document.getElementById("demo").
style.fontSize = "35px";

// ocultar/mostrar um conteúdo html
document.getElementById("demo").
style.display = "block";
```

Interatividade com HTML e CSS

O JavaScript pode mudar ou ocultar um elemento html, seus atributos e valores. Também pode mudar os estilos (css), modificando os características visuais dos elementos de página.

JavaScript pode "exibir" dados escrevendo em um elemento HTML, usando innerHTML, na saída HTML usando document.write(), em uma caixa de alerta, usando window.alert(), no console do navegador, usando console.log().

HTML, CSS & JS.



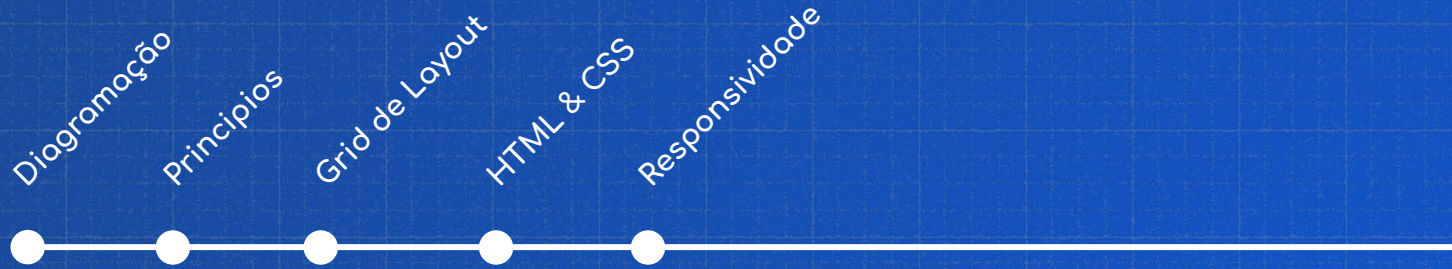
Laboratório

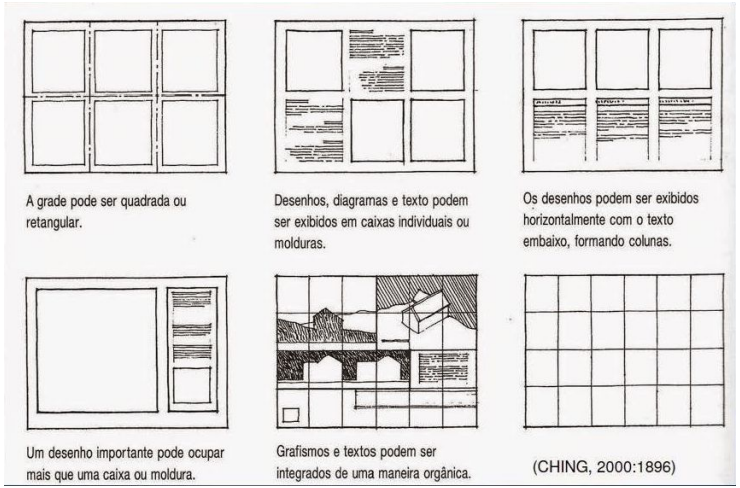
Faremos o desenvolvimento centrado em html, css e js. Mas os alunos que desejarem utilizar algum framework por sua conta, teremos apenas algumas dicas para implementação com Angular.

A principio, não exploraremos frameworks para acesso a recursos de usabilidade do celular, como Capacitor ou Cordova.

Grid e Responsividade

Linha de Pensamento





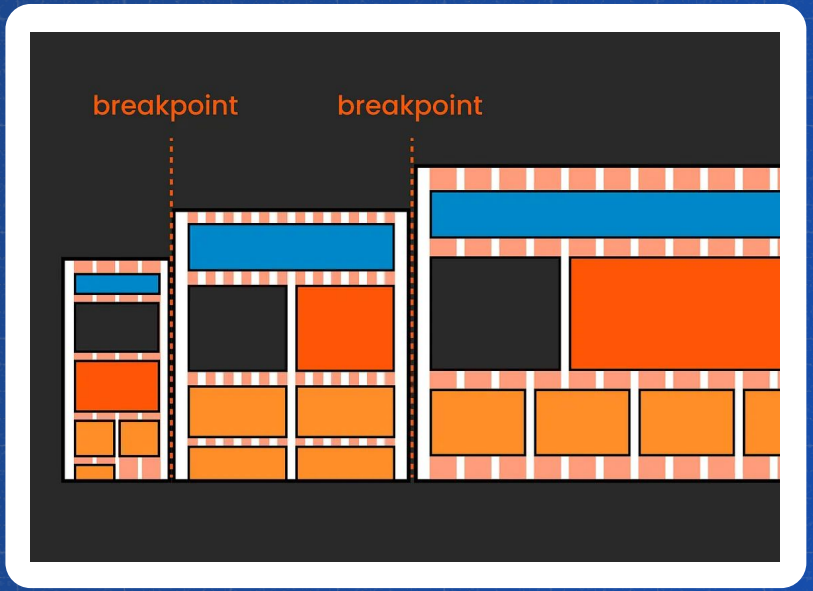
Diagramação

Diagramação é o processo de organizar elementos visuais, como texto, imagens e espaços vazios, em um layout harmonioso e eficaz. Envolve considerações de hierarquia, equilíbrio, proporção e alinhamento para melhorar a legibilidade, usabilidade e estética de um design. A diagramação é aplicada em diversos contextos, como design gráfico, editorial, web design e arquitetura da informação, buscando criar composições visualmente atrativas e funcionais.

- Hierarquia Visual
- Espaço e Alinhamento
- Equilíbrio
- Contraste
- Repetição e Consistência
- Simplicidade e Clareza
- Proporção e Escala

Princípios da Diagramação

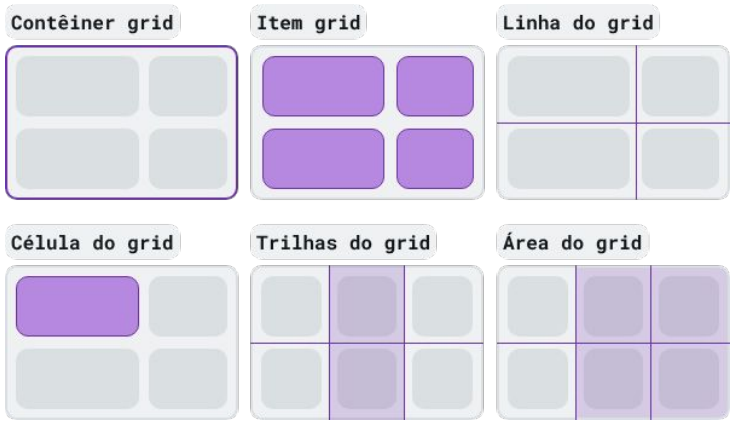
Os princípios da diagramação são diretrizes essenciais para criar layouts visualmente eficazes e esteticamente agradáveis. Incluem hierarquia visual (ênfase e organização de elementos), equilíbrio (distribuição uniforme de peso visual), contraste (destaque de elementos importantes), alinhamento e espaçamento adequados, repetição de elementos para consistência, uso de proporções e escalas harmoniosas, e busca por simplicidade e clareza. Esses princípios são fundamentais em design gráfico, editorial, web design e outras disciplinas visuais.



Grid de Layout

Um grid de layout é uma estrutura composta por linhas e colunas que define áreas e espaços onde os elementos de um design podem ser colocados. Os principais componentes de um grid de layout incluem: linhas e colunas, gutters, módulos ou células, margens e preenchimentos.

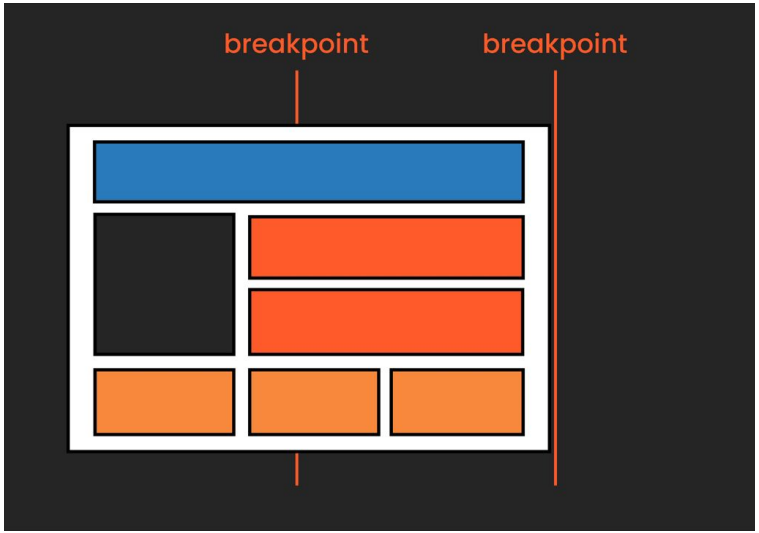
No grid clássico os ajustes do layout são realizados com base em breakpoint de tamanho de tela.



CSS Grid

O CSS Grid é uma técnica de layout bidimensional em CSS que permite criar grades flexíveis e complexas. Você define linhas e colunas explicitamente usando display: grid;, grid-template-columns, e grid-template-rows. Os itens são posicionados com grid-row e grid-column, permitindo controle preciso sobre o layout. É ideal para designs responsivos e substitui técnicas mais antigas de layout em CSS.

```
// mudar um conteúdo html
.container {
    display: grid;
    grid-template-columns: 1fr 2fr; /*
Duas colunas: a primeira com 1 parte e a
segunda com 2 partes */
    grid-template-rows: auto; /* Altura
automática para as linhas */
    grid-gap: 20px; /* Espaçamento entre
as células */
}
```



Obrigado!