

Relatório Semanal do Front-End – Semana 15.

Na tarefa de front-end desta semana, o foco foi criar uma interface usando o CSS Grid, buscando garantir a adaptação do layout a diversos tamanhos de tela e aparelhos.

O Grid Layout é uma funcionalidade do CSS pensada para dispor elementos em linhas e colunas ao mesmo tempo, servindo como uma "grade" que facilita a organização do conteúdo. Ao contrário do Flexbox, que organiza os itens em apenas uma direção (linha ou coluna), o Grid opera nas duas direções simultaneamente, proporcionando maior versatilidade no design.

Dentro do projeto, foram desenvolvidos cinco blocos de conteúdo que se ajustam de forma responsiva dentro do container principal. O CSS Grid simplifica essa tarefa ao dispensar cálculos manuais de margens e posições, tornando o código mais enxuto e eficaz.

No arquivo index.html, foi criada uma `<div class="grid-container">` que reúne os elementos:

```
<div class="grid-container">
<div class="item1"> ... </div>
<div class="item2"> ... </div>
<div class="item3"> ... </div>
<div class="item4"> ... </div>
<div class="item5"> ... </div>
</div>
```

Cada item contém um texto explicativo sobre a importância do Grid, sendo posicionado em diferentes locais conforme as regras especificadas no CSS.

Estilos CSS:

No arquivo style.css, a estrutura é definida da seguinte maneira:

```
.grid-container {
display: grid;
grid-template-columns: repeat(3, 1fr);
gap: 15px;
background-color: gold;
padding: 15px;
border-radius: 10px;
}
```

`display: grid;` → Habilita o sistema de grade.

`grid-template-columns: repeat(3, 1fr);` → Divide o espaço em 3 colunas de mesmo tamanho.

`gap: 15px;` → Determina o espaço entre os blocos.

Ademais, alguns itens ocupam mais espaço na grade, como o `.item1` e o `.item5`:

```
.item1 {  
grid-column: 1 / span 2; /* utiliza duas colunas */  
}
```

```
.item5 {  
grid-column: 1 / span 3; /* usa a linha inteira */  
}
```

Isso demonstra como o Grid permite manipular áreas maiores para certos elementos.

Responsividade:

Para assegurar que a interface funcione bem em telas menores, foi utilizada a `@media` query:

```
@media (max-width: 768px) {  
.grid-container {  
grid-template-columns: 1fr; /* transforma em 1 coluna */  
}  
}
```

Deste modo, todo o conteúdo se reorganiza em uma única coluna quando a tela é menor, como em smartphones.

Conclusão:

O CSS Grid é uma ferramenta valiosa para criar layouts modernos, adaptáveis e organizados. Ele diminui a complexidade do código, elimina a necessidade de cálculos manuais de espaçamento e gar

ante que a interface se ajuste a diversos aparelhos.

Neste projeto, o Grid foi usado para estruturar cinco blocos de conteúdo de forma clara e eficaz, evidenciando na prática por que ele é considerado uma das técnicas mais relevantes no desenvolvimento front-end atual.