CSS GRID LAYOUT

CRIANDO LAYOUTS CSS PROFISSIONAIS

Maurício Samy Silva

Novatec

Editor: Rubens Prates
Revisão gramatical: Tássia Carvalho
Capa: Carolina Kuwabata
Editoração eletrônica: Carolina Kuwabata

ISBN: 978-85-7522-632-2

Histórico de impressões:

Novembro/2017 Primeira edição

Novatec Editora Ltda.
Rua Luís Antônio dos Santos 110
02460-000 – São Paulo, SP – Brasil
Tel.: +55 11 2959-6529
E-mail: novatec@novatec.com.br
Site: www.novatec.com.br
Twitter: twitter.com/novateceditora
Facebook: facebook.com/novatec
LinkedIn: linkedin.com/in/novatec

CAPÍTULO 1

Introdução e terminologia

1.1 Introdução

A especificação do W3C denominada *CSS Grid Layout Module Level 1* é o Módulo Nível 1 para layout CSS com uso de grid. Trata-se da mais recente e fantástica especificação para criação de layout CSS colocada à disposição dos desenvolvedores.

A história e a trajetória dos métodos de criação de layout CSS desde seu início, em meados dos anos 1990 até o início do ano de 2017 (portanto por mais de 25 anos), vêm sendo um dos maiores desafios e causadores de dores de cabeça aos desenvolvedores frontend.

Criar layout CSS com uso das funcionalidades de posicionamento, floats, display table e toda a parafernália prevista nas CSS2, por mais diversificadas que fossem aquelas técnicas, sempre e invariavelmente se baseava e baseia em soluções dependentes de hacks, inconsistentes e à mercê das versões de navegadores e quase sempre dos malfadados prefixos proprietários.

O resultado final era um layout criado com marcação inflada com elementos que se prestam unicamente para possibilitar obtenção de efeitos de apresentação, sem qualquer valor semântico e carregando consigo o peso de aumentar o tempo de carregamento da página.

Simples efeitos, tais como o arredondamento de bordas, eram obtidos com acréscimo de vários elementos extras na marcação cuja finalidade era apenas servir de container para imagens de fundo. O custo de arredondar bordas com uso de imagens é inadmissível no desenvolvimento atual. Se para um simples arredondamento de bordas era assim, o que dizer da construção do layout completo da página?

Nenhuma especificação CSS anterior às das CSS3 progrediu no ritmo que avança nos dias atuais. E assim foi com a especificação para CSS Grid Layout. Ela já está aí. O suporte nos navegadores atuais é uma realidade e essa é a hora de se aprender a novíssima técnica.

Seja bem-vindo às maravilhas previstas na especificação para CSS Grid Layout.

A partir daqui, usaremos os termos CSS Grid Layout ou simplesmente Grid Layout com o mesmo significado.

1.2 Terminologia

Os conceitos e as técnicas próprias do Módulo para CSS Grid Layout foram criados com base em um algoritmo de posicionamento dos componentes do layout (elementos da marcação) que diferem de quase tudo que aprendemos com o estudo das CSS2 para criar layout. Portanto, é importante que não se façam suposições baseadas no que já sabemos com relação a posicionamentos, floats, clear e técnicas relacionadas.

Por exemplo: floats e clear simplesmente não funcionam, portanto não são usadas; medidas fixas podem ser combinadas com medidas flexíveis para preencher um container sem necessidade de uso da função `calc()`. Alinhamento vertical, horizontal, margem automática, entre outros, são efeitos obtidos com uma simples declaração CSS.

1.2.1 CSS Grid Layout

A especificação do W3C o define assim:

> CSS Grid Layout é um sistema para criação de layout baseado em uma grade bidimensional e otimizado para design de interfaces de usuário. Nesse modelo de grade, os elementos-filhos do container que define a grade podem ser posicionados livremente em espaços criados na grade, quer ela tenha sido definida com suas dimensões flexíveis ou fixas.

A criação de layout com uso de uma grade não é um conceito atual. A mídia impressa utiliza a técnica de grades para distribuir conteúdos há várias décadas. A maioria das técnicas tipográficas e de apresentação de interfaces na web baseia-se naquelas empregadas na mídia impressa.

Para suprir as inconsistências próprias do emprego das variadas técnicas CSS2 para criar layouts, surgiram os nossos conhecidos e muito bem-vindos frameworks CSS. Entre os muitos existentes, destacamos os pioneiros Blueprint e Grid960, o Foundation e o mais famoso e usado atualmente, o Bootstrap.

Frameworks CSS são sem dúvida uma ferramenta poderosíssima na criação de layout. São fáceis de se aprender e usar e os resultados obtidos são fantásticos, criando layouts consistentes e responsivos. Contudo, a facilidade na criação do layout tem um custo que não cabe discutir no contexto deste livro, mas convém citar.

A necessidade de se carregar com a página a exibir uma série de arquivos necessários para fazer o framework funcionar e mais a quantidade de marcação extra com suas inúmeras classes e atributos. Essas duas exigências por si só comprometem a performance e a semântica da marcação e, assim sendo, usar um framework é uma decisão que precisa e deve ser analisada e muito bem estudada.

Com a implementação das funcionalidades de Grid Layout nativamente nos navegadores, vários efeitos facilmente obtidos com uso de frameworks CSS já podem ser obtidos com simples regras CSS, sem necessidade de carregamento de arquivos adicionais e sem ter que se inflar a marcação HTML com inúmeras classes e atributos, muito menos com os indesejáveis elementos extras na marcação.

Uma série de novas propriedades, funções e valores CSS foi criada pelas especificações e já implementada nos navegadores modernos que definitivamente chegaram para revolucionar a maneira como se cria layout CSS.

Podemos gritar aos quatro cantos do mundo: agora sim, temos uma ferramenta fantástica para criar layout CSS.

Estude, aprenda e seja feliz!

1.2.2 Terminologia do Grid Layout

Antes de começar a explorar as funcionalidades do Grid Layout, precisamos nos familiarizar com alguns termos próprios dessa técnica, pois não estamos acostumados com eles já que não existem nas versões anteriores das CSS.

O perfeito entendimento dos referidos termos é necessário para que se entenda o exato sentido dos textos que explicam a teoria quando neles tais termos forem inseridos.

1.2.2.1 Grid

Em desenvolvimento de interfaces, grid (grade ou malha em português) é uma estrutura geométrica gráfica constituída por eixos horizontais e verticais e destinada a facilitar a distribuição, o posicionamento e o alinhamento dos elementos visuais que compõem a interface.

O grid é constituído por colunas e linhas. Na Figura 1.1, mostramos um exemplo de dois grids. À esquerda, o grid 3 x 3 é constituído por três colunas e três linhas e, à direita, o grid 6 x 4 por seis colunas e quatro linhas.

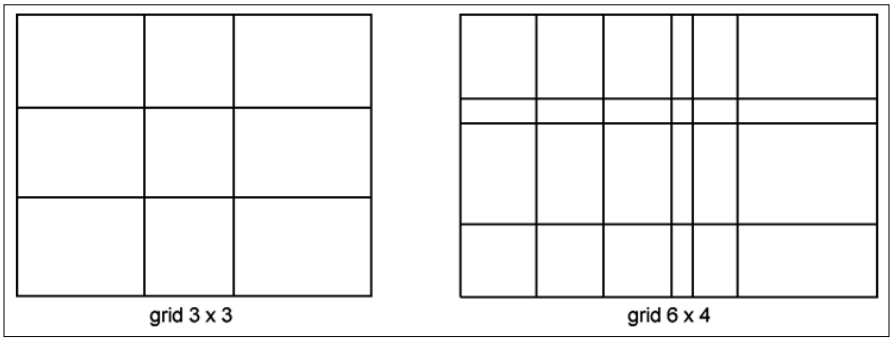

Figura 1.1 – Grid.

1.2.2.2 Grid lines

Grid lines são os eixos verticais e horizontais que constituem o grid. Em CSS, por padrão, eles são nomeados com números sequenciais a partir de um, da esquerda para a direita quando se trata de eixos verticais e de cima para baixo quando se trata de eixos horizontais.

Essa numeração é importante, pois ela será declarada como valores para algumas das novas propriedades CSS de Grid Layout como veremos adiante. O grid se constitui em um verdadeiro sistema de coordenadas para posicionamento de conteúdos.

Observe na Figura 1.2 as *grid lines* e seus números.

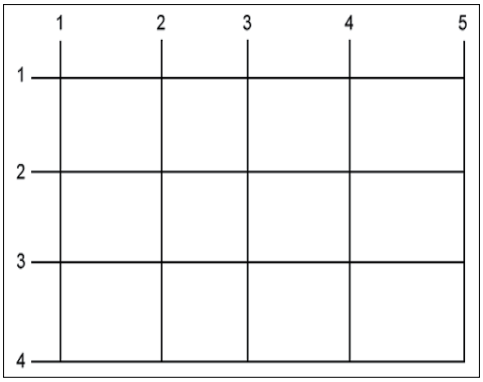

Figura 1.2 – Grid lines.

1.2.2.3 Grid track

Grid track é o termo usado para designar o espaço compreendido entre duas *grid lines* consecutivas. Observe na Figura 1.3 dois grid tracks, um horizontal (linha) e um vertical (coluna) mostrados em destaque.

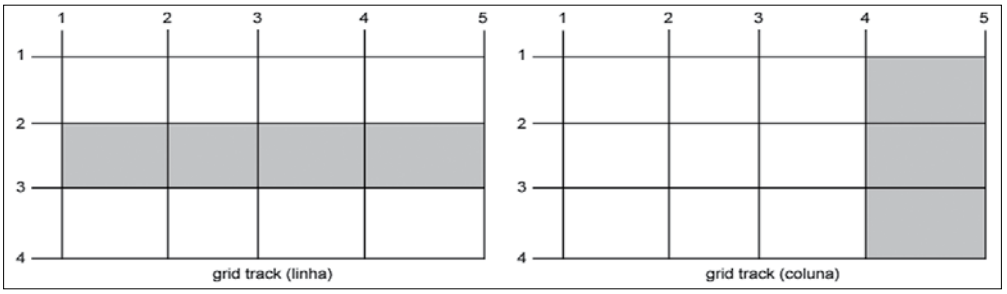

Figura 1.3 – Grid track.

Em inglês, os termos *grid line* e *row* são bem distintos e não dão margem à troca de interpretação. Em português, a tradução técnica para aqueles termos é linha de grid e linha do grid. A diferença sintática dos termos em português é sutil e poderá gerar confusão. Neste livro, para evitar confusão, usaremos o termo inglês *grid line* para designar os eixos horizontais e verticais do grid e o termo linha ou linha do grid para designar o espaçamento horizontal entre *grid lines*, ou seja, a *row* ou o grid track horizontal.

1.2.2.4 Grid cell

Grid cell é o termo usado para designar as células do grid. Célula é a área do quadrilátero compreendido entre duas *grid lines* horizontais e verticais consecutivas ou, ainda, a intersecção de uma coluna com uma linha do grid. Observe na Figura 1.4 um exemplo mostrando, em destaque, três células do grid.

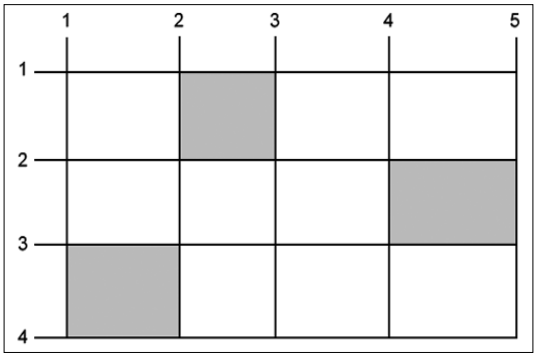

Figura 1.4 – Grid cell.

1.2.2.5 Grid gap

Grid gap é o espaçamento opcional a ser definido entre as células do grid. As figuras mostradas anteriormente foram de grids sem espaçamentos entre células, sem *grid gap*.

A especificação prevê a possibilidade de se definirem espaçamentos, tanto horizontal como vertical, entre as células do grid.

Observe na Figura 1.5 um exemplo de grid 4 x 3 com espaçamentos mostrados em destaque.

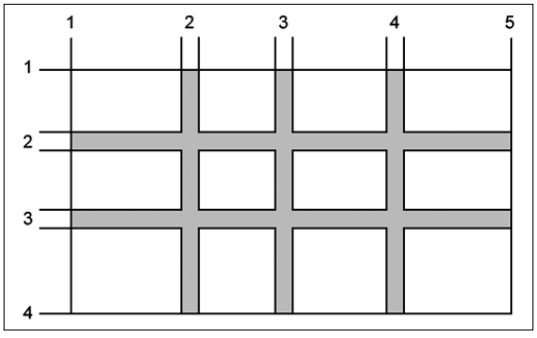

Figura 1.5 – Grid gap.

Convém notar, conforme claramente mostrado na figura, que a definição de *grid gap* com uso de uma propriedade CSS, a ser estudada adiante, não cria novas *grid lines*.

1.2.2.6 Grid area

Grid area é o termo usado para designar uma área do grid formada por uma ou mais células adjacentes e limitada por quatro *grid lines*, uma em cada lado da área.

Observe na Figura 1.6, mostrada em destaque, uma das inúmeras possíveis *grid areas* de um grid 4 x 3.

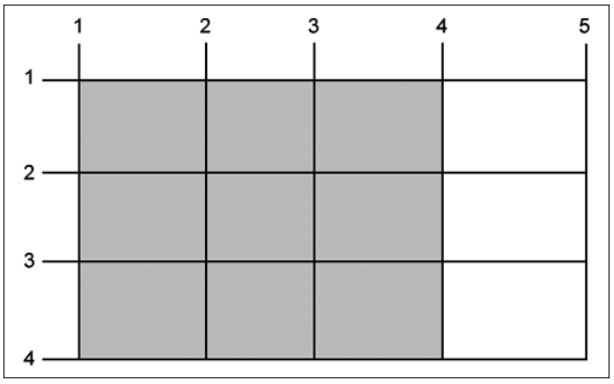

Figura 1.6 – Grid area.

1.2.2.7 Grid container

Grid container (neste livro, chamaremos simplesmente container) é um elemento da marcação contendo elementos-filhos e destinado a criar um contexto de formatação de grid para seus elementos-filhos.

A fim de facilitar a compreensão, fazemos uma comparação com CSS2 que estabelece os nossos conhecidos conceitos de formatação nível de bloco e `inline`.

Em Grid Layout, o contexto de formatação estabelecido é o "nível de grid", com suas particularidades próprias como: posicionar todos os elementos-filhos em linha, não reconhecer as propriedades `float` e `clear`, entre outras.

Para que um container estabeleça um contexto de formatação do tipo "nível de grid", ou seja, entre em modo grid, a especificação criou o valor `grid` para a já existente propriedade CSS `display` conforme mostrado no código a seguir:

- CSS

```
.container {
   display: grid;
}
```

1.2.2.8 Grid item

Grid items são os elementos-filhos de um *grid container*.

Neste livro, adotaremos o termo em inglês *grid item* para designar os itens de um grid.

Considere os códigos conforme mostrados a seguir:

- **HTML**

```
<div class="container">
   <div class="box box1">1</div>
   <div class="box box2">2</div>
   <div class="box box3">3</div>
   <div class="box box4">4</div>
</div>
```

- **CSS**

```
.container {
   display: grid;
}
```

Ao se declarar o container no modo grid layout como mostrado na regra CSS, cada um dos quatro elementos `div` filhos do container passa a ser considerado e se comporta como um *grid item*.