

**SÉRGIO
LOPES**

Use sempre media queries baseadas no conteúdo da sua página

Uma versão mais atual, revisada e expandida desse assunto está disponível no meu [livro "A Web Mobile"](#).

Ao escrever [medias queries](#), você precisa escolher algum valor pra colocar lá. É o que chamamos dos **breakpoints**, os pontos onde seu layout vai ser ajustado por causa de uma resolução diferente. E **escrever bons breakpoints** é essencial para um design responsivo de qualidade.

E o que mais aparece de pergunta de quem tá começando com design responsivo é: *quais os valores padrões de se colocar nas media queries?* E logo surge uma lista parecida com essa:

```
@media only screen and (min-width: 480px) { ... }  
@media only screen and (min-width: 600px) { ... }  
@media only screen and (min-width: 768px) { ... }  
@media only screen and (min-width: 992px) { ... }  
@media only screen and (min-width: 1382px) { ... }
```

Péssima prática! Essa lista eu copiei do famoso projeto [320andup](#) mas você acha outras parecidas, com esse valores supostamente “comuns”. *480px* é por causa de um iPhone em landscape, *600px* é comum em tablets de 7", *768px* pega um iPad em portrait, *992px* é um Desktop de 1024 menos as barras de rolagem e *1382px* pra Desktops grandes.

O pessoal chama essa prática de **device-driven breakpoints**, pois são valores gerados a partir de tamanhos de dispositivos.

Não use device-driven breakpoints

Primeiro problema: essa lista pensa em meia dúzia de tipos de dispositivos. Mas e os *360px* de um Galaxy SIII? Ou os *533px* do Galaxy SII em landscape? Ou o iPhone 5 com *568px* em landscape? Qualquer lista de media queries padrão é **muito pobre** quando pensamos na situação atual – e futura – dos **diferentes dispositivos**.

Segundo problema: quem disse que *seu design* realmente precisa de todos esses breakpoints? Ou pior, que ele não vai ficar ruim bem entre as paradas de *480px* e *600px*? Usar esses valores de media queries **não garante que seu design funcionará em todos os dispositivos**, mas apenas nos dispositivos “padrões” – seja lá o que for isso.

Use content-driven breakpoints

Dá uma olhada em algumas das medias queries que uso aqui no blog: (PS. aliás, o código do blog tá disponível no [github](#) e já uso tudo com em por causa de [acessibilidade](#))

```
@media (min-width: 37em) { ... } /* 592px */  
@media (min-width: 41em) { ... } /* 656px */  
@media (min-width: 47em) { ... } /* 752px */  
@media (min-width: 68em) { ... } /* 1088px */  
@media (min-width: 82em) { ... } /* 1312px */  
@media (min-width: 100em) { ... } /* 1600px */
```

Da última vez que chequei, não consegui encontrar nenhum dispositivo com *656px* ou *752px*. De onde saíram então esses valores? **Do design do meu blog.**

A ideia é simples: fiz meu design responsivo com unidades flexíveis – porcentagens e em – e pra pequenos dispositivos primeiro – *mobile first*. Aí fui testando as diversas resoluções pra ver onde o design não ficava tão bom e coloquei um breakpoint lá.

É o que o pessoal chama de **content-driven breakpoints**. Ou seja, achar suas media queries a partir do seu conteúdo e do seu design. Fica bem mais fácil garantir que sua página **funciona em todos os dispositivos**.

Como achar meus breakpoints?

- Abra sua **página original** no navegador;
- Vá **redimensionando a janela devagar** até o design parecer ruim – se fez *mobile-first*, abra pequeno e vá aumentando a janela; senão, abra grande e vá diminuindo a janela;
- Quando achar um ponto em que o design quebra, copie **o tamanho da janela** e crie uma media query com esse valor lá no seu CSS;
- Recarregue a página, veja se as mudanças melhoraram o design, e continue redimensionando pra achar o **próximo breakpoint**.

Algumas ferramentas que podem te ajudar nisso:

- O [responsive mode](#) do Firefox;
- [FitWeird](#), um excelente bookmarklet que te dá o tamanho da tela, inclusive em em;
- Meu [responsive play](#).

Escrito por [Sérgio Lopes](#) em 14 fev 2013.



Compartilhar no Facebook



Compartilhar no Twitter



Comentários