

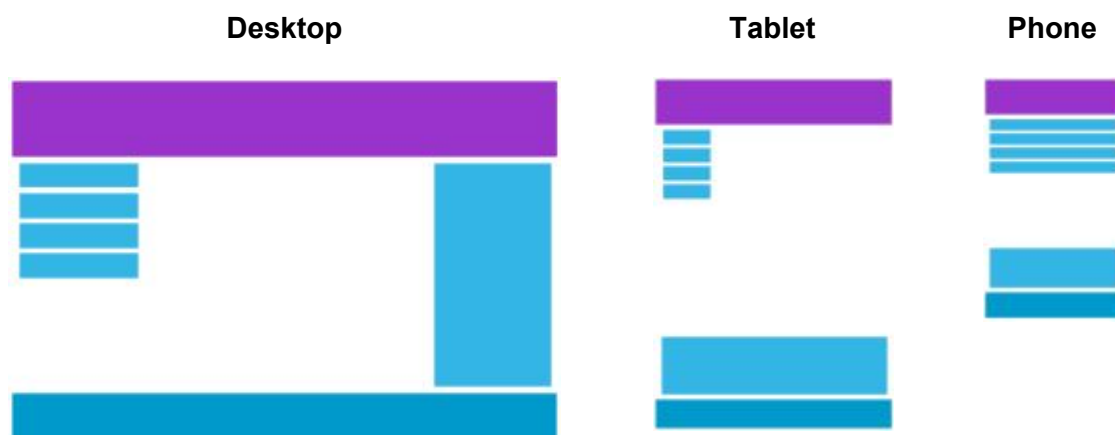
Responsive web design

Wat is responsive web design?

Responsive web design is het ontwerpen van een web applicatie met als doel de optimale webervaring (via een schaalbaar design) voor een zo breed mogelijk scala van apparaten (desktop schermen, tablets, mobiele telefoons).

De webpagina's zullen op basis van de grootte van het scherm geschaald worden voor een optimale leesbaarheid en gebruiksvriendelijkheid.

Voorbeeld schaalbaarheid op verschillende apparaten:



Een responsive web design kan men bekomen door volgende technieken:

1. Flexibele grids en afbeeldingen
= het gebruik van relatieve eenheden (%) of em) i.p.v. vaste eenheden in pixels
2. CSS Media queries

Dit kan men bereiken door gebruik te maken van HTML en CSS.

Voor het vergemakkelijken van het opstellen van de HTML en CSS kan men gebruik maken van een framework. Een vaak gebruikt framework is Bootstrap welke we in een ander hoofdstuk zullen bespreken. In dit hoofdstuk gaan we de basics voor een responsive design uitleggen via media queries.

Er zijn uiteraard meerdere frameworks beschikbaar zoals bijvoorbeeld W3.CSS, Foundation, Material UI, ...

Eén van de manieren om aan responsive web design te doen wordt "Mobile First" genoemd. Hiermee zal men de design van een web applicatie opmaken door te beginnen met het kleinste scherm (mobile).

Responsive web design basics

Hieronder bespreken we door middel van een aantal simpele stappen hoe men een web pagina schaalbaar kan maken.

1. ViewPort

De ViewPort is het door de gebruiker zichtbare deel van een web pagina.

De ViewPort varieert afhankelijk van het apparaat. Deze is namelijk kleiner op een mobiele telefoon dan op een computerscherm.

Vóór er sprake was van tablets of mobiele telefoons waren webpagina's alleen ontworpen voor desktops en was het gebruikelijk dat webpagina's een statisch ontwerp en een vaste grootte hadden.

Toen men begon te surfen op internet met tablets en mobiele telefoons, waren webpagina's met een vaste grootte te groot om in de viewport te passen. Om dit op te lossen, moeten we de browsers op die apparaten de hele webpagina laten schalen zodat ze op het scherm passen.

Om de ViewPort als designer te controleren kan men gebruik maken van de <meta> tag.

```
<meta name="viewport" content="width=device-width, initial-scale=1.0">
```

Deze tag vertelt de browser om de apparaat breedte te gebruiken als de breedte van de viewport en de initiële schaal uit te schakelen (=initiële zoom level)

De metatag moet opgenomen worden in het <head> element van de html pagina.

2. HTML structuur

Via volgend voorbeeld zullen we aantonen hoe we deze schaalbaar (responsive) kunnen maken via media queries.

De header heeft een vast hoogte van 180px, de content container is 600px breed en de sidebar is 300px breed.

Online voorbeeld: <http://webdesignerwall.com/demo/responsive-design/index.html>

Header

Tutorial by [Web Designer Wall](#) ([read related article](#))

Content

text

text

text

text

text

text

text

text

text

text

Sidebar

text

text

Footer

3. Media queries

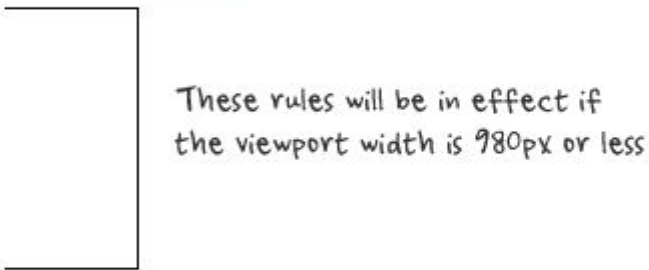
CSS3 mediaqueries is de oplossing om een pagina responsive te maken.

Je kan het vergelijken met het schrijven van "if" condities om de browser te vertellen hoe de pagina moet worden gerenderd voor de gespecificeerde viewport-breedte.

De volgende css waardes zijn van toepassing indien de viewport 980px breed is of minder.

```
/* for 980px or less */
@media screen and (max-width: 980px) {

    #pagewrap {
        width: 94%;
    }
    #content {
        width: 65%;
    }
    #sidebar {
        width: 30%;
    }
}
```




These rules will be in effect if the viewport width is 980px or less

Indien de viewport 700px of minder breed wordt geven we #content en #sidebar een automatische breedte en verwijderen we de float.

Hierdoor zullen deze de volledige breedte van de viewport innemen

```
/* for 700px or less */
@media screen and (max-width: 700px) {

    #content {
        width: auto;
        float: none;
    }
    #sidebar {
        width: auto;
        float: none;
    }
}
```



Reset container to auto width

Tenslotte voor de afmeting van een viewport van 480px of minder (bijvoorbeeld voor een mobiele telefoon) stellen we volgende waardes in

```
/* for 480px or less */  
@media screen and (max-width: 480px) {  
  
    #header {  
        height: auto;  
    }  
    h1 {  
        font-size: 24px;  
    }  
    #sidebar {  
        display: none;  
    }  
  
}
```

Buiten deze 3 voorbeelden kunnen er uiteraard zoveel media queries als gewenst opgegeven worden.

De css media queries kunnen in dezelfde stylesheet geplaatst worden als in een apart css bestand.

More

Meer gedetailleerde info over media queries door middel van een voorbeeld kan je terug vinden op <http://webdesignerwall.com/tutorials/responsive-design-with-css3-media-queries>

CSS tricks

<http://webdesignerwall.com/tutorials/5-useful-css-tricks-for-responsive-design>

DEMO sites

<http://webdesignerwall.com/demo/adaptive-design/final.html>

<https://responsivewebdesign.com/robot/>