

Responsive webdesign

- Flexible
 - Font sizing met rem
 - Grid
 - Media
- Meta tag: viewport
- Media queries

- Strategie
- Flexible
 - Font sizing met rem
 - Grid
 - Media
- Media queries
 - viewport
 - CSS media query

Strategie

- Een design maken dat in élke resolutie past
- Praktisch
 - Weg met horizontale scrollbalken
 - Mobile first!
 - Layout veranderen wanneer layout onduidelijk wordt
 - -> weg van vaste breekpunten

Font sizing met rem

- Vroeger met em
 - Font size wordt berekend op basis van de font-size van de container
 - Veel rekenwerk, overbodig -> betere oplossing: rem
- Rem
 - één keer een meeteenheid aanduiden in html-selector in %
 - Standaard staat de rem van elke pagina op 100% (van 16px = standaard font size)

Font sizing met rem

- één keer een meeteenheid aanduiden in html-selector in %
 - Standaard staat de rem van elke pagina op 100%
 - =16px, de standaard font-size
 - Deze wordt manueel op 62,5% gezet
 - = 10px
 - Dit om duidelijker en makkelijker te werken met de rem-eenheid (1,6em = 16px)

Font sizing met rem

- Alle andere elementen gebruiken font-size met als eenheid rem.
 - Verwijst naar de de % gedefinieerd in de html-selector voor de font-size
 - Vaak dubbele notatie (pixels & rem) voor crossbrowser-doeleinden
 - font-size : 30px;
font-size : 3rem;
 - Laatste regel overschrijft de voorgaande, op voorwaarde dat de eenheid van deze regel herkend wordt door de browsers
- http://snook.ca/archives/html_and_css/font-size-with-rem

Flexible grids

- Gebruik altijd een grid bij het ontwerpen
 - Maakt implementeren van responsive makkelijker
- Bij omzetten van grid naar HTML
 - Altijd procentueel tewerk gaan
 - ALTIJD PROCENTUEEL TEWERK GAAN!
 - Geen pixels gebruiken bij het definiëren van breedte, margin, padding
 - » Uitzondering voor margin/padding wanneer er gebruik wordt gemaakt van BOX-SIZING
 - » <http://www.paulirish.com/2012/box-sizing-border-box-ftw/>

Flexible media

– Media

- Images
- Videos
- Embedded (flash, ...)

– Probleem: breken vaak uit container

- Hebben een vaste breedte toegekend
- Oplossing:
 - **max-width:100%**
 - Schaalt automatisch naar de maximum mogelijke grootte van de container

Meta-tag: viewport

– Viewport

- Staat toe om in te stellen hoe breed je fysieke scherm is

– <meta>-tag in <head>

- `<meta name="viewport" content="">`
- `<meta name="viewport" content="initial-scale=1">`
 - Er kan geen zoom worden toegepast
 - De breedte is niet per se de breedte van het toestel
 - » Het toestel kan zo ingesteld zijn dat er automatisch wordt uitgezoomd op een schaal van bv. 1:2
 - Dan is de scale van de viewport 1200px breed (= initial scale)
 - Best niet gebruiken -> neemt controle van de gebruiker weg

Meta-tag: viewport

- `<meta name="viewport" content="width=device-width, initial-scale=1">`
 - Er kan gezoomd worden
 - De breedte is de breedte van het toestel
 - » Overschrijft arbitraire zoom ingesteld door het toestel
 - Beste optie voor responsive design
- Meer info:
<http://webdesign.tutsplus.com/articles/quick-tip-dont-forget-the-viewport-meta-tag--webdesign-5972>

Media queries

- Conditionele CSS-operator die kijkt welk medium (print, screen, projector, ...) gebruikt wordt én welke dimensies dit medium heeft
- Op basis van medium/dimensies kan een andere css-stylesheet ingeladen worden
- Voor responsive webdesign: enkel luisteren naar wanneer een bepaalde resolutie (= breedte) wordt overschreden
- Meerdere queries op elkaar stapelen om zo voor meerdere resoluties te ontwerpen

Media queries

– Strategie:

- Ontwerp éérs de mobiele versie
 - (van lage resolutie naar hoge resolutie)
- Daarna per dimensie dingen toevoegen dmv media query
- Niet werken met vaste breekpunten (320px, 480px, 680px, ...)
 - Kijk naar design en beslis waar het design niet meer duidelijk is

Media queries

- Technisch:

```
@media all and (min-width: 480px)
{
  body { background: #ccc; }
}
```

- Meer info: <http://css-tricks.com/css-media-queries/>