

"NOT ALL SCREENS ARE MADE EQUAL"

# AYOUTSTRATEGIER FOR WEBDESIGN

# DAGENS PROGRAM

- Lidt baggrund
- Et par eksempler
- Øvelser
- Opgave

# HVAD ER DET LIGE DET HANDLER OM?

- Enheder har forskellige skærmstørrelser
  - - og forskellige oplosninger
- Designere ønsker at brugere får en optimal oplevelse, uanset enhed
- Som udviklere ønsker vi at kode så lidt som muligt
  - ikke flere versioner der skal vedligeholdes samtidig
- Som site owner ønsker vi god SEO

# LAYOUTFORMER

- Fire overordnede tilgange
  - Statisk (static, fixed width)
  - Flydende (fluid, liquid, flydende, elastisk)
  - Adaptiv (adaptive, ADW)
  - Responsiv (responsive, RWD)

# STATISK WEBDESIGN

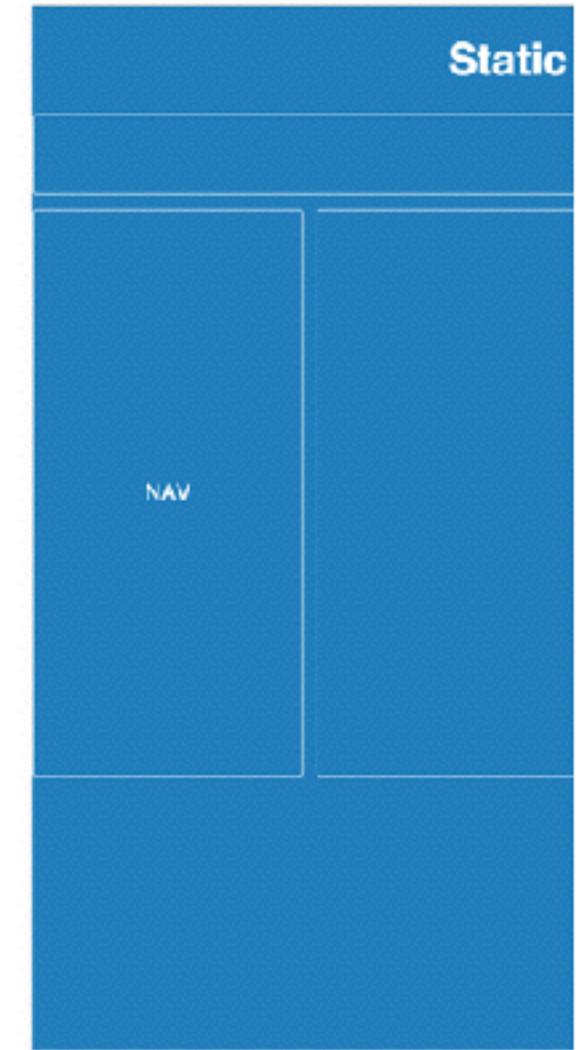
- Layoutet baseres på faste mål (fixed width)
- Ændres ikke ved forskellige skærmstørrelser
- Ekstra plads på skærmen bliver til "luft" i siderne
- Beskæres typisk i siden på desktop
- Skaleres (zoomes) typisk på mobile enheder
- Meget dominerende i www's tidlige år
- Designeren har mere kontrol over slutresultatet (hvis skærmen er stor nok)
- Fungerer som regel dårligt på andre skærmstørrelser end den tiltænkte



## Static Layout Example



1440px

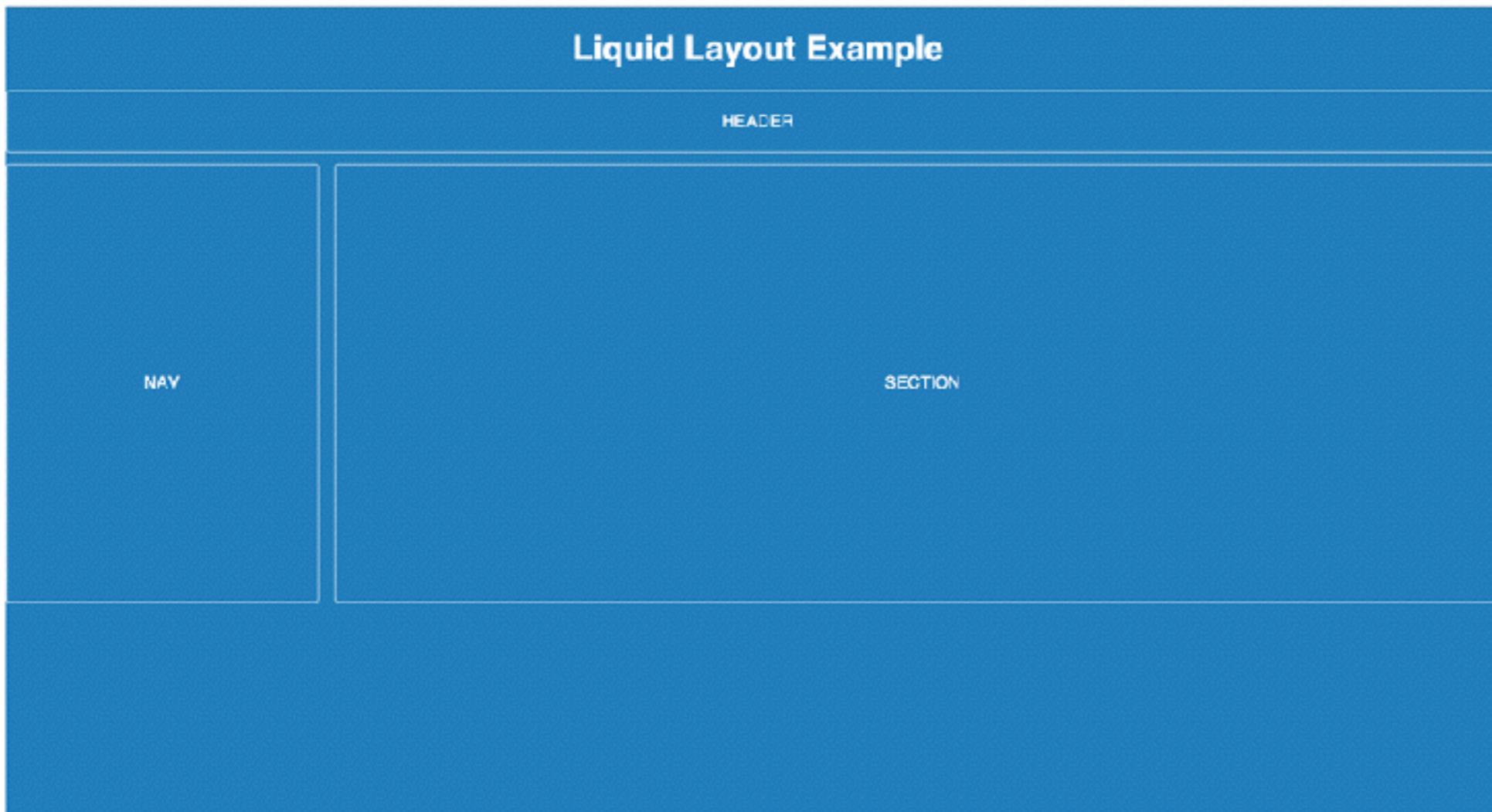


400px

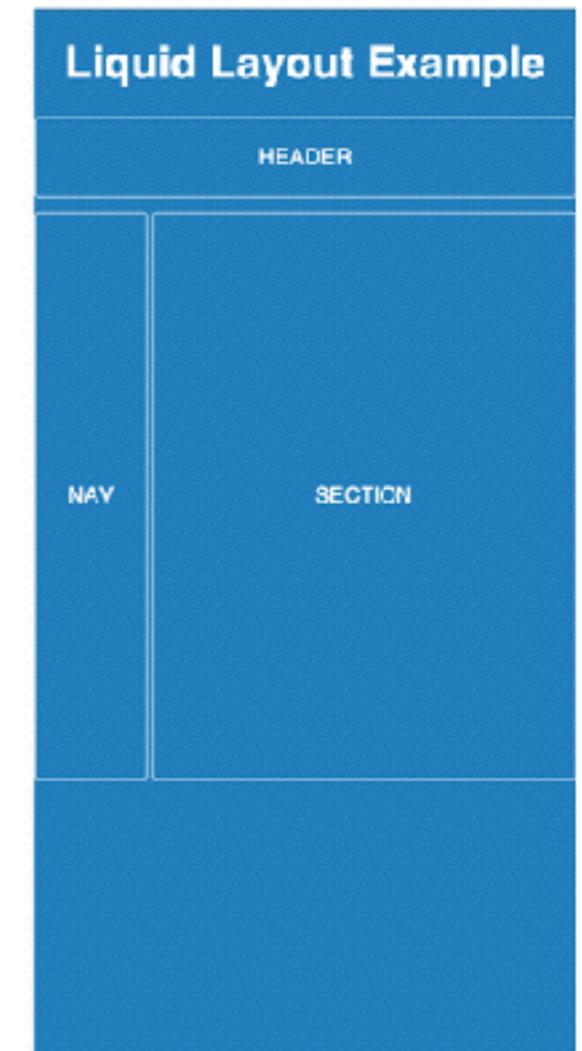
# FLUID WEBDESIGN

- Elementer tilpasser sig bredden
- Angiver bredder relativt, typisk i procent
- Elementer kan blive meget smalle
- Ofte er der grænser for hvor smalt indhold kan blive
- Foretrækkes hvor indhold og funktionalitet vægter højere end grafisk design
- Udskældt af æstetisk-orienterede designere
- Udnytter pladsen bedre
- Ser bedre typisk bedre ud
- Giver ofte en bedre brugeroplevelse





1440px



400px

# RESPONSIVE LAYOUT

- Består typisk af et fluid layout
  - og en række regler der ændrer på layoutet afhængig af skærmbredde
  - @media
- Samme HTML og CSS-filer, uanset enhed
- Dagens standard
- Ikke specielt vanskeligt
- Nødvendigt med planlægning

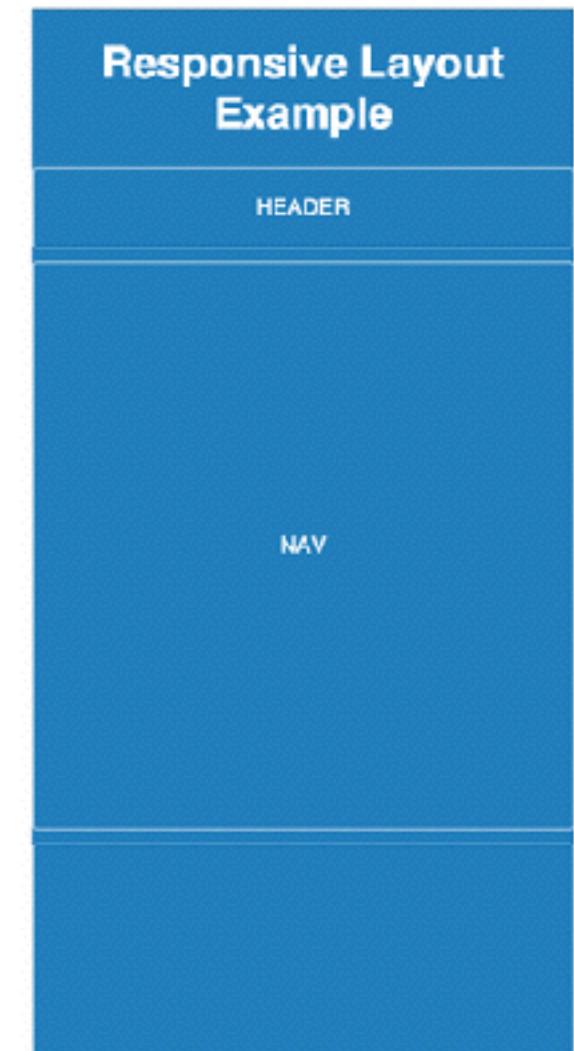


## Responsive Layout Example



1440px

## Responsive Layout Example



400px



The image displays the Jyske Bank digital ecosystem, featuring a desktop website, a tablet, and a smartphone, all sharing a consistent design language.

**Desktop Website:**

- Header:** JYSKE BANK, Produkter, Finansnyt, TV, Kontakt, Logg, LOG PÅ NETBANK.
- Main Content:** A large green banner with the headline "Få en pension, du selv har valgt". Below it, text reads: "Er du ung, ældre eller midt i livet? Dit behov for forsikring og pension afhænger af din situation – af dit liv nu og her og dine ønsker til fremmoden." A "Bliv kunde" button is visible.
- Right Sidebar:** Three cards: "Lån til erhvervskunder" (with a coin icon), "Gå i banken på din iPad" (with a tablet icon), and "Spar op til pension" (with a hot air balloon icon).
- Bottom Navigation:** Buttons for Investingering, Pension, Kort, Lån, Valuta, and categories for Privat, Erhverv, and Private Banking. A "Bliv kunde" button is also present here.
- Footer:** jyskebank .tv, Finansnyt, and a news item about Chr. Hansen.

**Tablet View:**

- Header:** JYSKE BANK, Produkter, Finansnyt, TV, Kontakt, Logg, LOG PÅ NETBANK.
- Content:** Shows the same main banner and sidebar as the desktop site.
- Bottom Navigation:** Similar to the desktop, with buttons for various services and a "Bliv kunde" button.

**Smartphone View:**

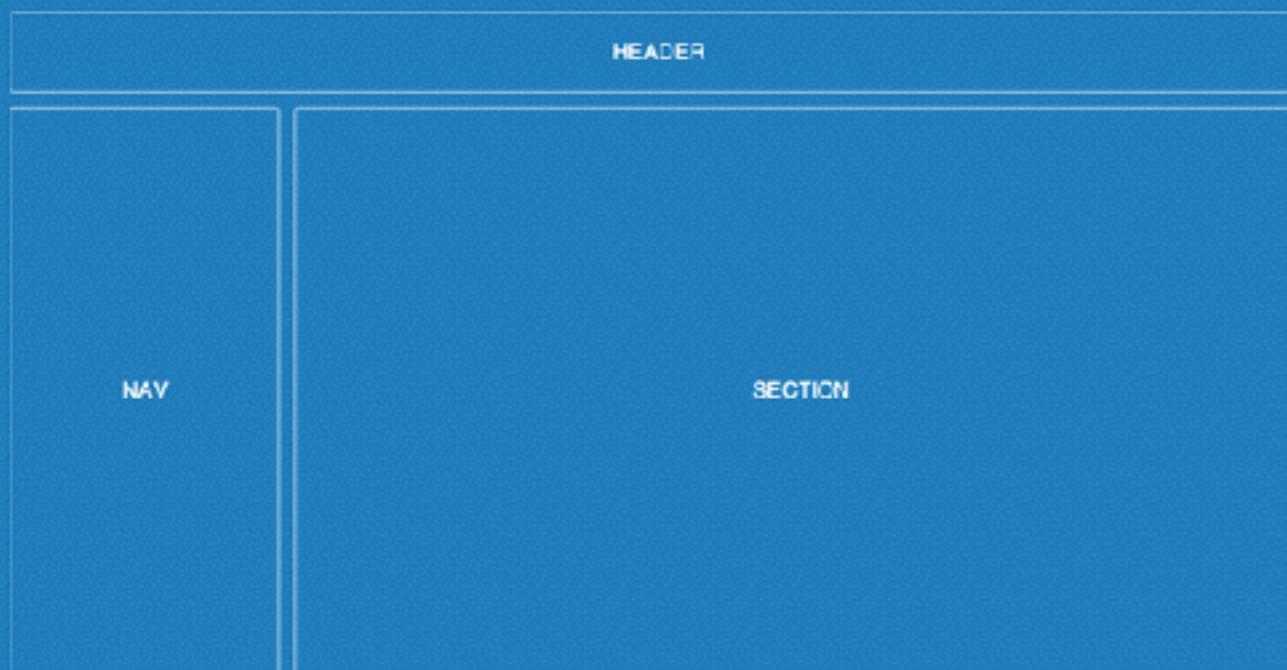
- Header:** JYSKE BANK, Menu og tag.
- Content:** Shows the same main banner and sidebar as the desktop site.
- Bottom Navigation:** Standard smartphone navigation buttons.

# ADAPTIVE LAYOUT

- En mellemting mellem statisk og responsive'
- Består af en række statiske layouts
- Styres af @mediaqueries
- På skærme der ikke rammer breakpoints vil der typisk være for meget eller for lidt plads
- Sjældent noget man vælger, hvis man har muligheden for at gøre det responsivt
- Anvendes typisk hvis et statisk layout skal konverteres til responsivt, uden at skrive det hele om
- Dermed bliver det typisk mobile-last

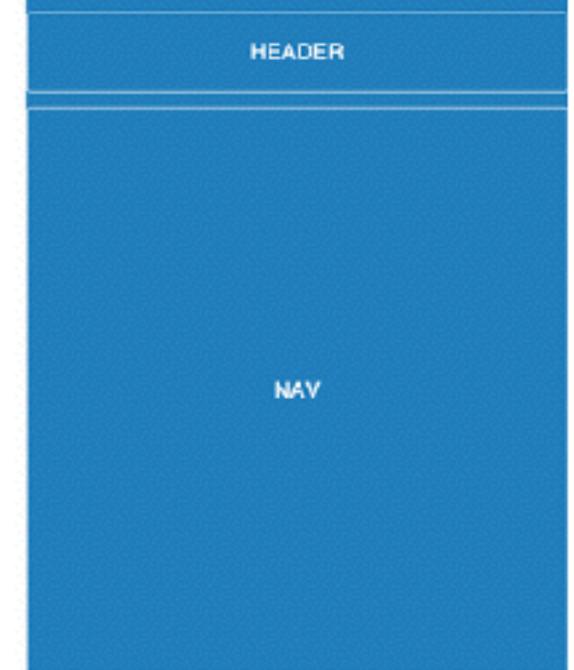


## Adaptive Layout Example



1440px

## Adaptive Layout Example



400px

ET ONLINEEKSEMPEL

[HTTP://WWW.LIQUIDAPSIVE.COM](http://www.liquidapsive.com)

# Responsive

Responsive

## Menu 1

Menu 1 a  
Menu 1 b  
Menu 1 c

## Menu 2

Menu 2 a  
Menu 2 b  
Menu 2 c

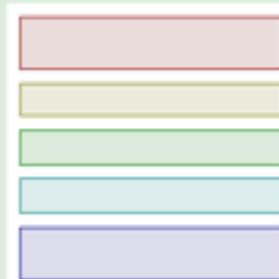
## Menu 3

Menu 3 a  
Menu 3 b  
Menu 3 c

## How To Use

Ever wondered what the difference between Adaptive, Responsive, Static and Liquid sites is? Had someone try and explain it only to be left more confused?

Pick a flavor from the drop down on the top of the page then drag your window narrower and wider, taller and shorter. It will make much more sense when you see for yourself how the approach works.



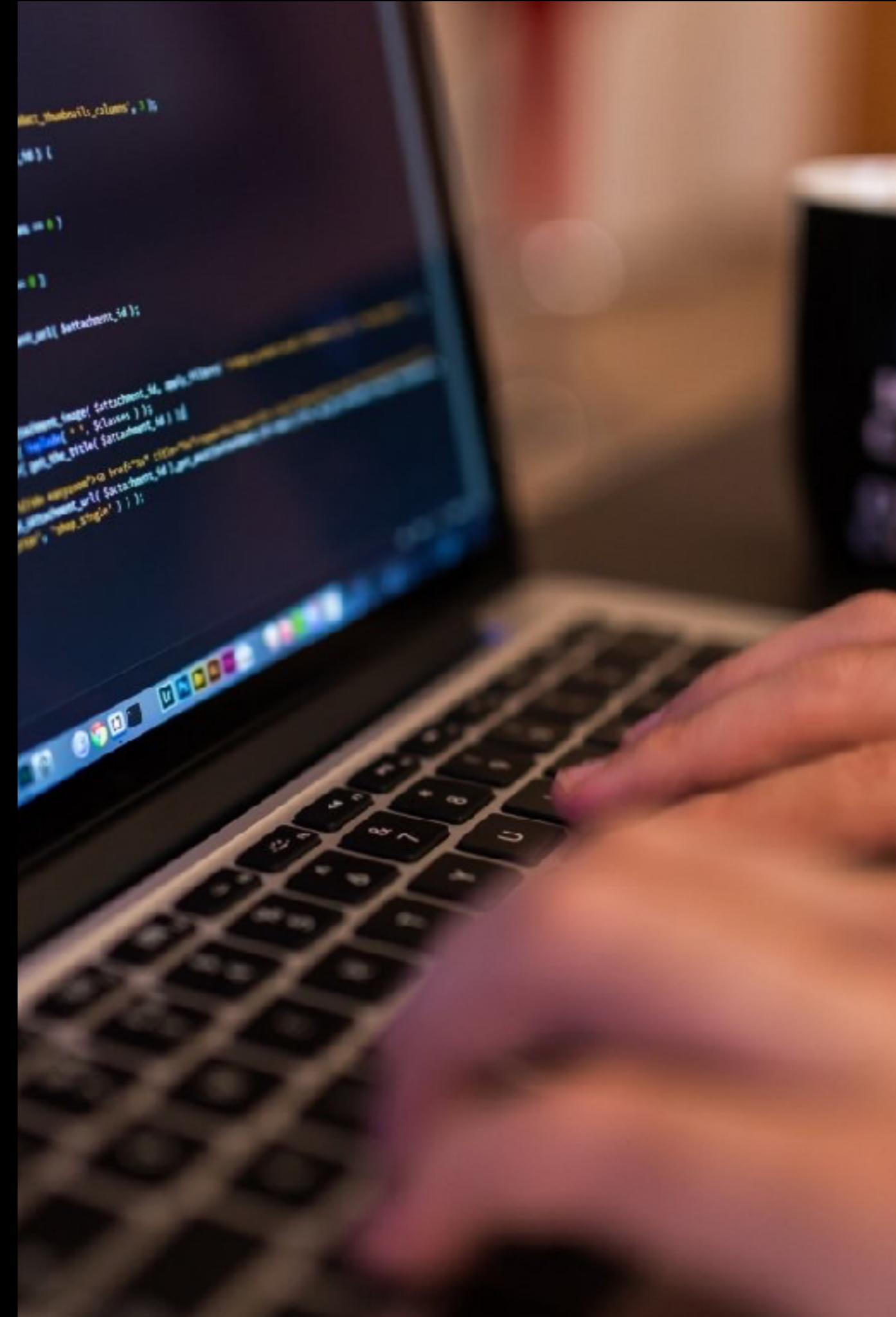
## Responsive

Responsive is characterized by having defined layouts for different resolutions. Within each layout, the design is liquid and resizes the width of elements relative to the changing window size.

You can think of Responsive as a series of [Liquid](#)

ET EKSEMPEL

I PRAKSIS



# EKSEMPEL 1

- Baggrund og tekstfarve skal være forskellig på stor og lille skærm
- Et breakpoint ved 800 pixels
- Gul på sort på lille skærm
- Sort på gul på stor skærm



```
1 <!DOCTYPE html>
2 <html lang="da">
3   <head>
4     <meta charset="utf-8">
5     <title>Responsive eksempel 1</title>
6     <link rel="stylesheet" href="ovelse1.css">
7
8   </head>
9   <body>
10  <div id="pagewrap">
11    <section>
12      <h1>Denne side er
13        responsive</h1>
14      <p>Hvis skærmen er mere end 500
15        pixels bred, er teksten gul på
16        sort, er skærmen mindre er
17        teksten sort på gul.</p>
18
19    </section>
20
21    <footer>
22      <p>&copy; Copyright Niels
23        Østergaard 2017</p>
24
25  </div><!-- pagewrap -->
26
27  </body>
28
29 </html>
```

```
1 /* eksempel1
2  (C) Copyright 2017: Niels Østergaard
3  Responsive webdesign */
4
5
6 /* Først en del der ikke er påvirket af regler */
7
8 body {
9   font-family: sans-serif;
10  background-color: #fc0;
11  margin: 1em;
12  color: #000;
13 }
14
15 footer {
16   font-size: 0.8em;
17   text-align: right;
18 }
19
```

## Denne side er responsive

Hvis skærmen er mere end 500 pixels bred, er teksten gul på sort, er skærmen mindre er teksten sort på gul.

© Copyright Niels Østergaard 2017

## Denne side er responsive

Hvis skærmen er mere end 500 pixels bred, er teksten gul på sort, er skærmen mindre er teksten sort på gul.

© Copyright Niels Østergaard 2017

# VI TILFØJER ET BREAKPOINT

- @media
  - angives til screen
    - Kan også angive print mv.
  - der angives en max-bredde som reglen dækker
  - Der angives CSS der gælder hvis reglen er i kraft

```
19  
20 /* så en del der kun gælder hvis skærmen er over  
500 pixels */  
21  
22 @media screen and (min-width: 500px) {  
23  
24     /* gentages body-selector */  
25 ▼   body {  
26         background-color: #000;  
27         color: #fc0;  
28     }  
29     /* de properties der ikke angives i reglen, er  
uændrede (eks. margen og skritttype)  
30 }|  
31
```

## Denne side er responsive

Hvis skærmen er mere end 500 pixels bred, er teksten gul på sort, er skærmen mindre er teksten sort på gul.

© Copyright Niels Østergaard 2017

## Denne side er responsive

Hvis skærmen er mere end 500 pixels bred, er teksten gul på sort, er skærmen mindre er teksten sort på gul.

© Copyright Niels Østergaard 2017

# UDFORDRING:

- Få footeren til at stå centreret når skærmen er bred



# LIDT OM TEKNIKKEN

- @media er en regel i CSS
- Kan angive om det er til:
  - skærm (screen)
  - papir (print)
  - tale (speech)
  - alle typer enheder (all)
- Vi anvender kun screen, men tilføjer parametre (media features)

# MEDIA FEATURES

- Der findes rigtigt mange!
- De typiske er:
  - min-width og/eller max-width
- - men kan også være:
  - antal farver enheden kan vise
  - skærmopløsningen i dpi/dpcm
  - flere andre enheder der sjældent anvendes
  - se: [https://www.w3schools.com/cssref/css3\\_pr\\_mediaquery.asp](https://www.w3schools.com/cssref/css3_pr_mediaquery.asp)

# BREAKPOINTS

- Et breakpoint er den bredde hvor en regel overtager fra en anden
- En responsive webside har typisk mindst to og gerne flere breakpoints
- Eks: Mobil - mindre end

# GLEM IKKE VIEWPORT

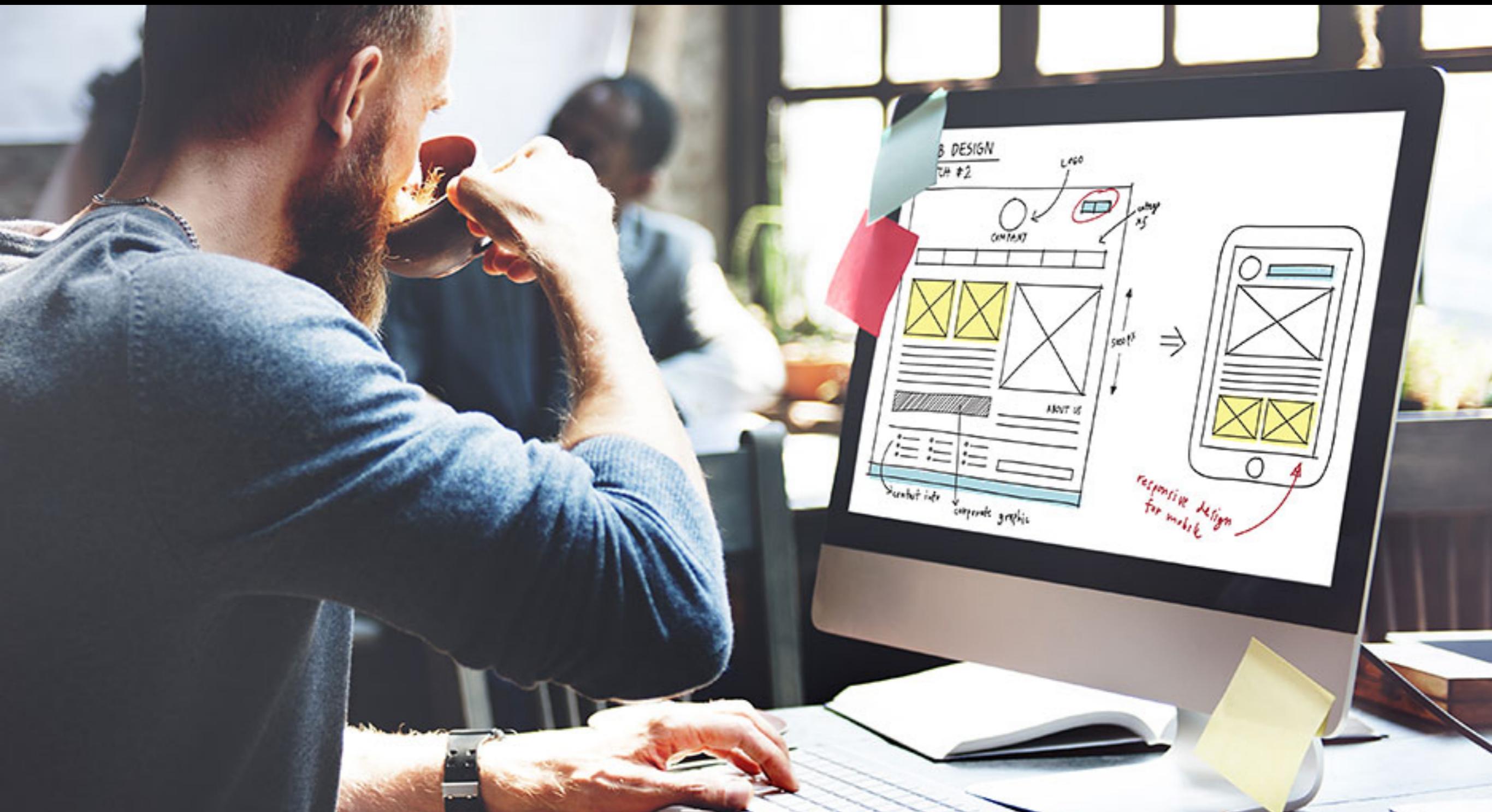
- Et meta-tag
- Kan fortælle at mobile enheder ikke skal forsøge at skalere (og dermed presse et desktop-site ned på arealet)
- Hvis den ikke angives, vil mediaqueries typisk ikke virke



```
<!DOCTYPE html>
<html lang="da">
  <head>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0"/>
```

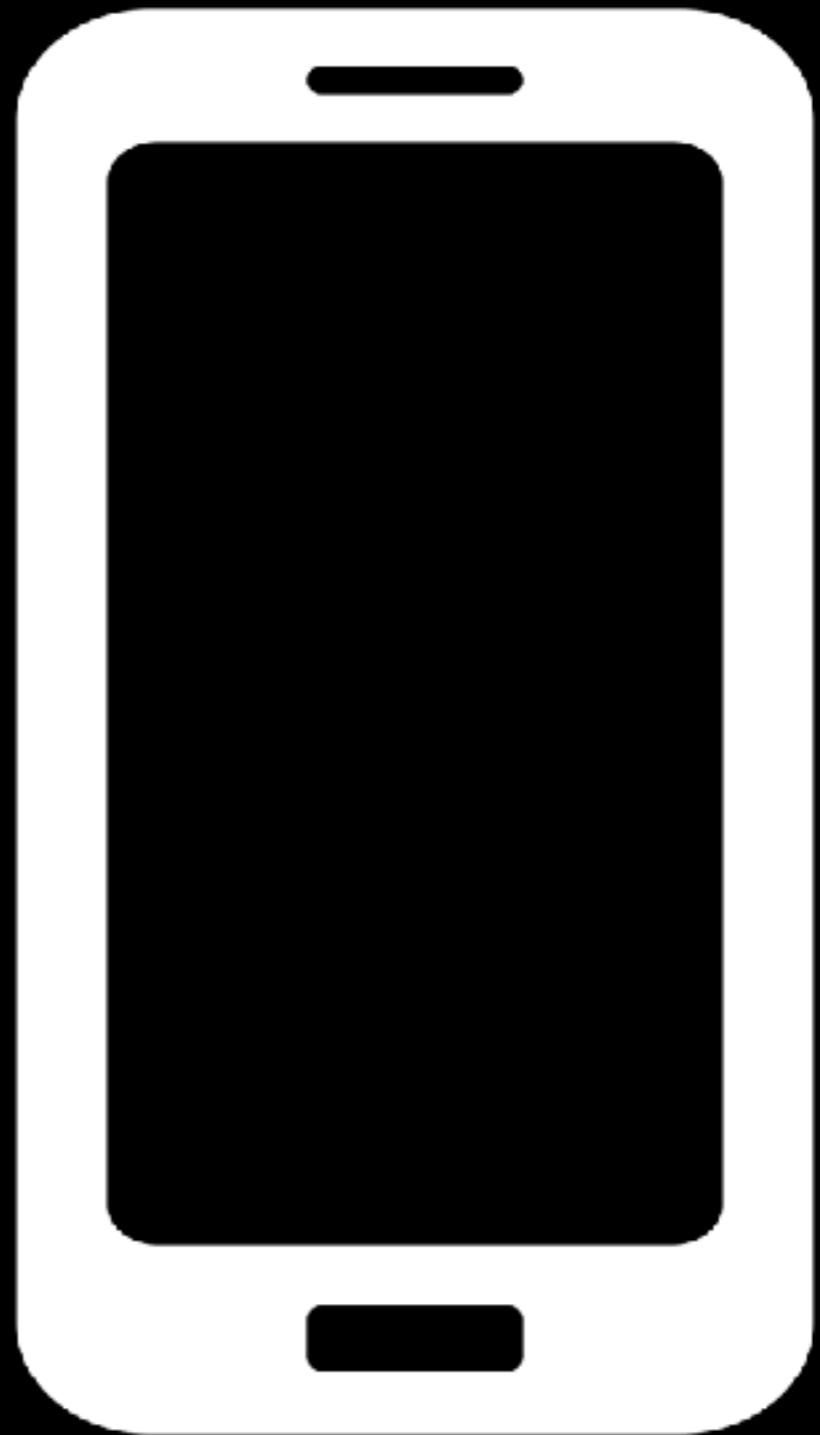
VIGTIGE OVERVEJELSER

# DESIGNSTRATEGIER



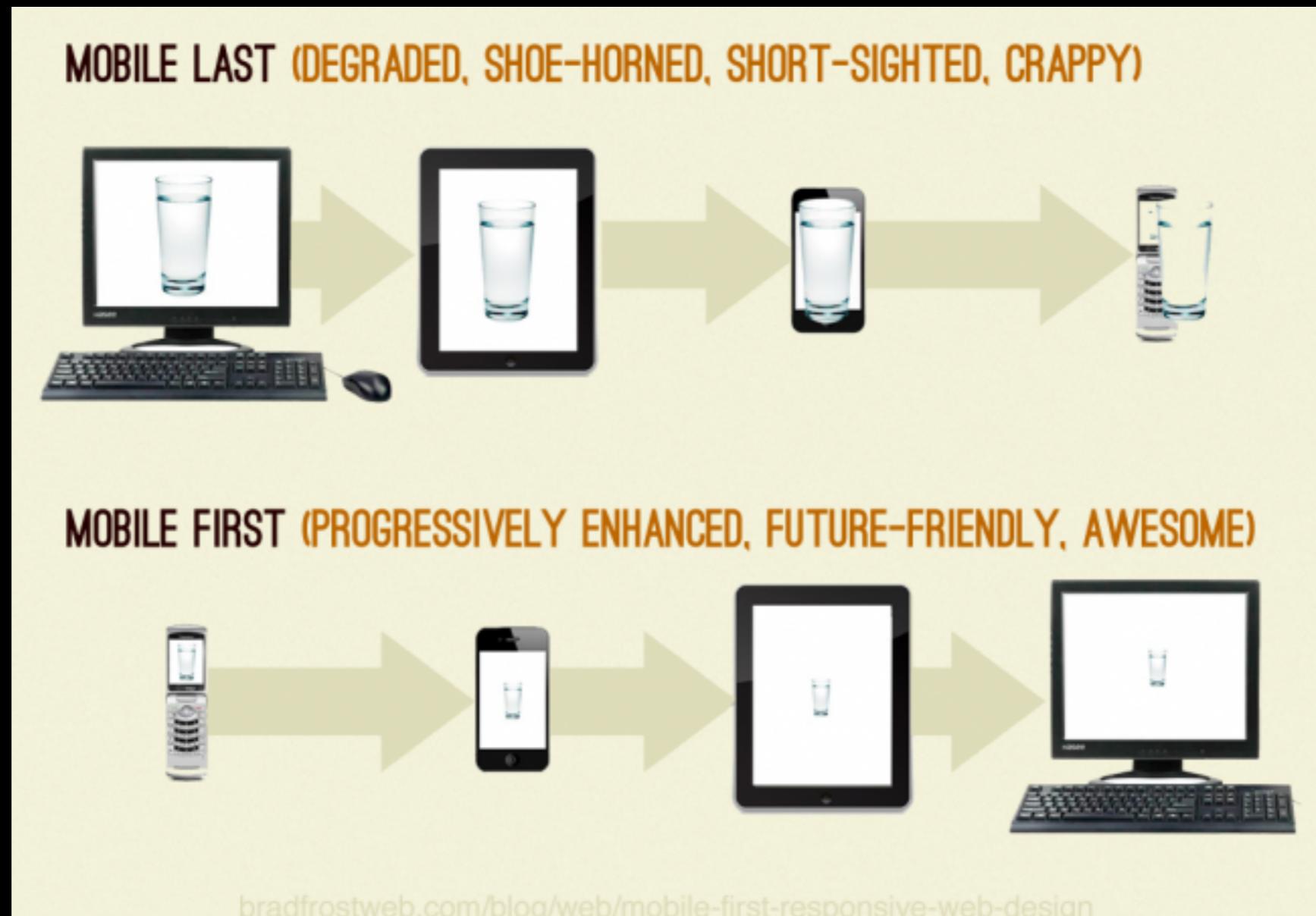
# MOBILE FIRST

- Hvis man tænker at tilpasning til mobil er en lille opgave man løser senere TAGER MAN FEJL!
- Det er altid vanskeligst at tilpasse informationen på begrænset plads
- Mobile enheder bruges oftere end laptops og stationære
- Brugeren er ofte i en kontekst hvor der sker noget andet - så det skal være veldesignet



# MOBILE FIRST

- Tager udgangspunkt i indholdshierarki
- Når man har skitseret for mobil, kan man gradvist skitsere til større skærme



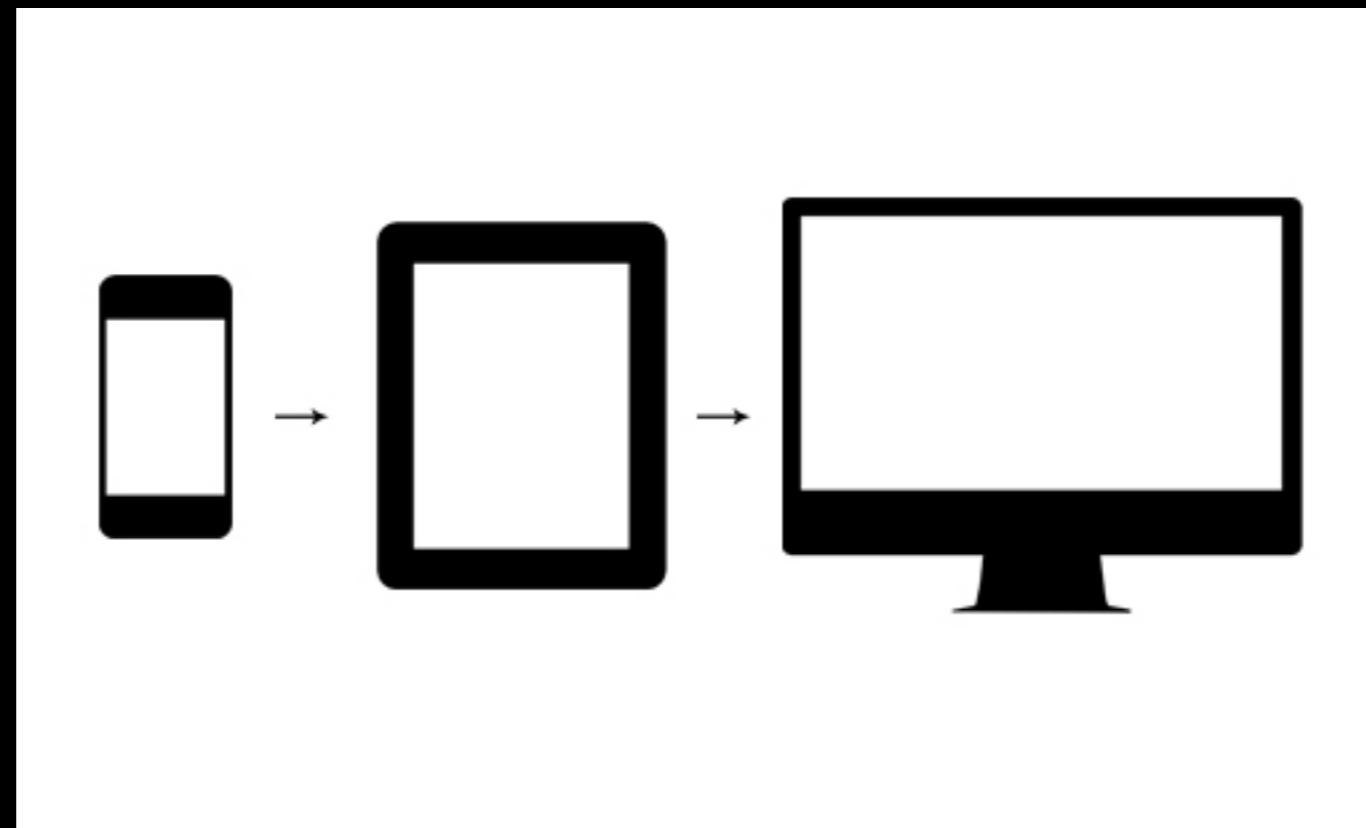
# MOBILE-FIRST CSS

- Betyder at du starter med et CSS som tilpasses den mobile enhed
  - Testes løbende på mobil
- Derefter tilføjes breakpoints (hvis større end)
  - Der testes løbende på enhed der opfylder breakpoint
- Der findes plugins der kan justere din browser..



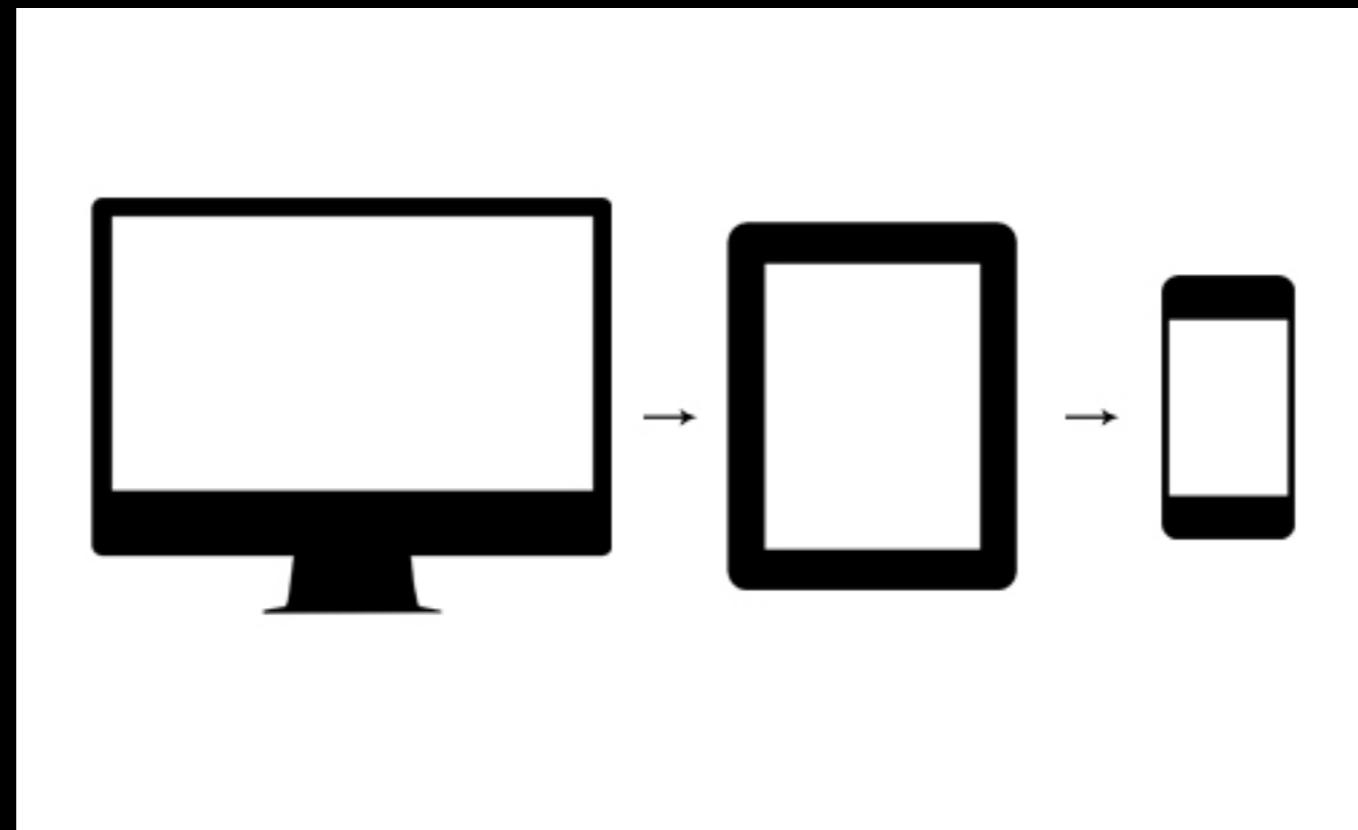
# PROGRESSIVE ENHANCEMENT

- Når man efterhånden tilføjer flere features på enheder der kan håndtere dem
- Typisk når man designer Mobile First



# GRACEFUL DEGRADATION

- Når man tager hensyn til brugeroplevelsen skal også være god på enheder med færre muligheder
- Typisk når man designer Desktop First



# MOBILE FIRST

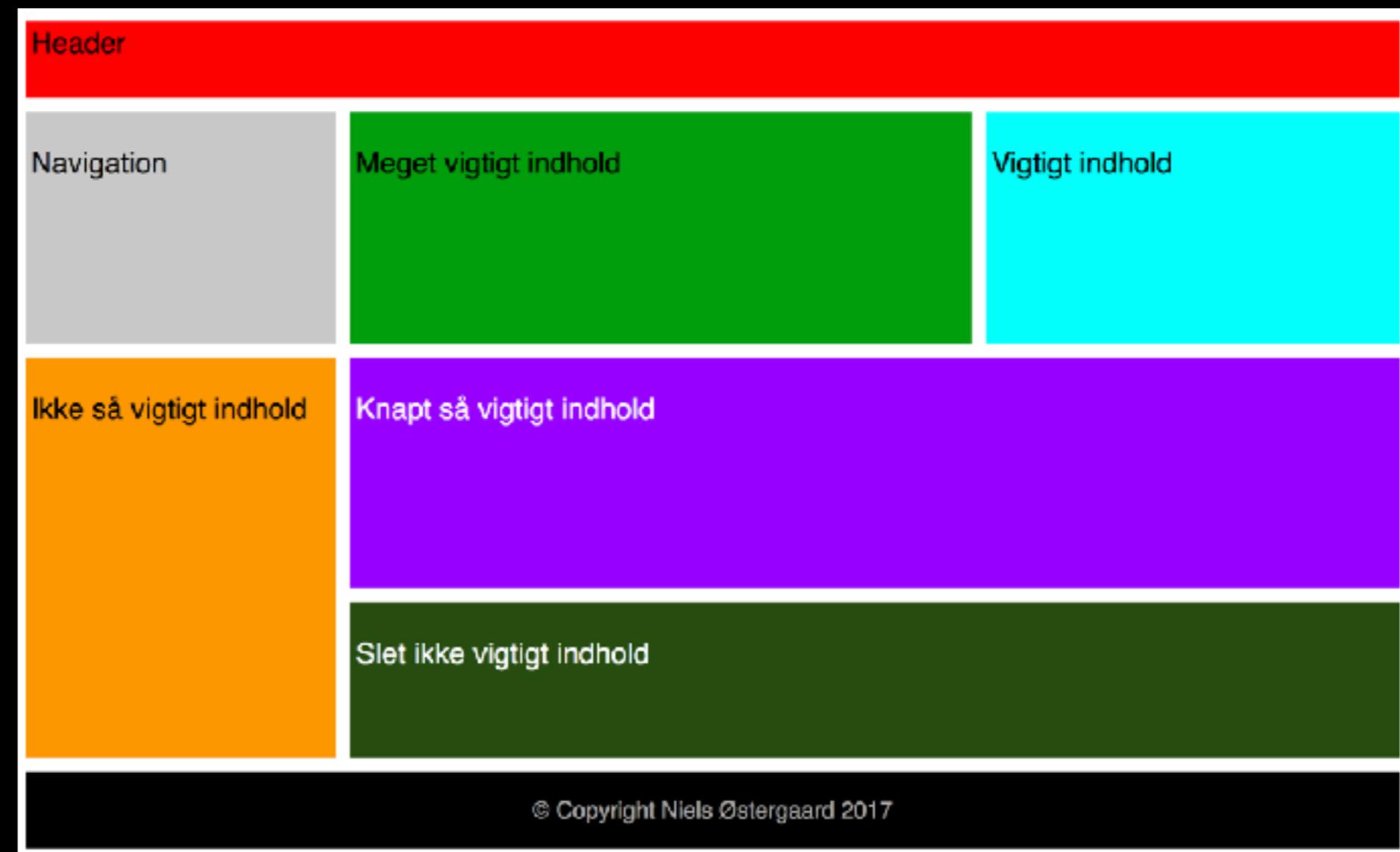
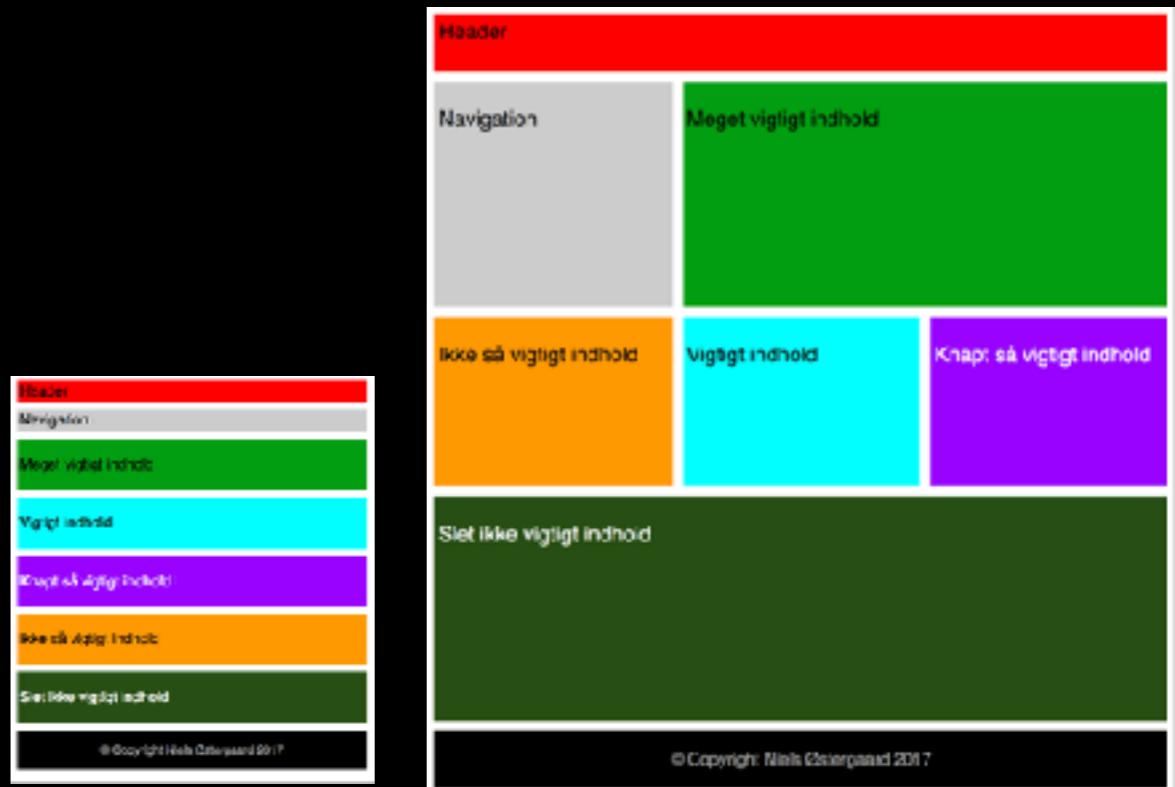
- Den foretrukne designstrategi
- Først CSS der fungerer på mobil
- Så CSS der fungerer på gradvist større enheder
- min-width: gradvist større tal...

# DESKTOP FIRST

- Typisk hvis man forsøger at gøre et eksisterende desktop-site responsivt
- CSS der fungerer på desktop
- Så CSS der fungerer på gradvist mindre enheder
- max-width: med gradvist mindre tal...

# MERE PRAKSIS

# RESPONSIVE TEKNIKKER SAMMEN MED GRID



# ØVELSE 2



```
1  <!DOCTYPE html>
2  <html lang="da">
3  <head>
4      <meta charset="utf-8">
5      <meta name="viewport" content="width=device-width, initial-scale=1.0"/>
6
7      <title>Responsive eksempel 2</title>
8      <link rel="stylesheet" href="ovelse2_a.css">
9  </head>
10 <body>
11 <div id="pagewrap">
12     <header> Header </header>
13     <nav> Navigation </nav>
14     <section id="megetvigtigt">
15         <p>Meget vigtigt indhold</p>
16     </section>
17     <section id="vigtigt">
18         <p>Vigtigt indhold</p>
19     </section>
20     <section id="knapsaavigtigt">
21         <p>Knapt så vigtigt indhold</p>
22     </section>
23     <section id="ikkesaavigtigt">
24         <p>Ikke så vigtigt indhold</p>
25     </section>
26     <section id="sletikkevigtigt">
27         <p>Slet ikke vigtigt indhold</p>
28     </section>
29
30     <footer>
31         <p>&copy; Copyright Niels Østergaard 2017</p>
32     </footer>
33 </div><!-- pagewrap -->
34 </body>
35 </html>
```

```
1 /* eksempel2|
2 (C) Copyright 2017: Niels Østergaard
3 Responsive webdesign */
4
5 /* først en del der ikke er påvirket af regler */
6
7
8 body {
9     font-family: sans-serif;
10    background-color: #fff;
11    margin: 0.5em;
12    color: #000;
13 }
14
15 header {
16     grid-area: a;
17    background-color: #F00;
18    padding: 0.2em;
19    margin-bottom: 0.5em;
20 }
21
22 nav {
23     grid-area: b;
24    background-color: #ccc;
25    padding: 0.2em;
26    margin-bottom: 0.5em;
27 }
28
29 #megetvigtigt{
30     grid-area: c;
31    background-color: #009e0f;
32    padding: 0.2em;
33    margin-bottom: 0.5em;
34 }
35
36 #vigtigt{
37     grid-area: d;
38    background-color: #0ff;
39    padding: 0.2em;
40    margin-bottom: 0.5em;
41 }
42
43 #knaptsaavigtigt{
44     grid-area: e;
45    background-color: #90f;
46    color: #fff;
47    padding: 0.2em;
48    margin-bottom: 0.5em;
49 }
50
51 #ikkesaavigtigt{
52     grid-area: e;
53    background-color: #f90;
54    padding: 0.2em;
55    margin-bottom: 0.5em;
56 }
57
58 #sletikkevigtigt{
59     grid-area: e;
60    background-color: #274e13;
61    color: #fff;
62    padding: 0.2em;
63    margin-bottom: 0.5em;
64 }
65
66 footer {
67     grid-area: h;
68    font-size: 0.8em;
69    text-align: center;
70    background-color: #000;
71    color: #aaa;
72    padding: 0.1em;
73 }
74
```

# MOBIL- UDGAVE

- Da alle er fuld bredde, er det let at opbygge uden CSS grid
- Felterne får højde efter indholdet



# FØRSTE BREAKPOINT (TABLET)

- Der indsættes mediaquery
- der angives en min. og max. bredde
- Der indsættes grid
- Margener fjernes, da der bruges grid-gap
- Skriftstørrelse tilpasses

```
75  /* så en del der kun gælder hvis skærmen er over
    500 pixels */
76
77 ▼ @media screen and (min-width: 500px) {
78
79 ▼   #pagewrap {
80     display: grid;
81     grid-gap: 0.5em;
82     grid-template-columns: 1fr 1fr 1fr;
83     grid-template-rows: 1fr 4fr 3fr 4fr 1fr;
84     grid-template-areas:
85       "a a a"
86       "b c c"
87       "f d e"
88       "g g g"
89       "h h h";
90   }
91
92 ▼   body {
93     font-size: 1.2em;
94   }
95 ▼   header, nav, #megetvigtigt, #vigtigt,
#ikkesaavigtigt, #sletikkevigtigt,
#knaptsaavigtigt { /* Vi fjerner margen, da
vi bruger grid-gap */
96     margin-bottom: 0.0em;
97   }
98 }
99 }
```

Header

Navigation

Meget vigtigt indhold

Ikke så vigtigt indhold

Vigtigt indhold

Knapt så vigtigt indhold

Slet ikke vigtigt indhold

# OPRET SELV SIDSTE BREAKPOINT

Header

Navigation

Meget vigtigt indhold

Vigtigt indhold

Ikke så vigtigt indhold

Knapt så vigtigt indhold

Slet ikke vigtigt indhold

# OPGAVER

- Opgave 1: med udgangspunkt i øvelse 2, tilføj et breakpoint høj oplosning hvor nogle elementer får rigtig meget plads
- Opgave 2: Fronter er ikke responsiv - kom med et bud...
  - Skitser en responsiv udgave af holdets rum på fronter. Der skal være tre bredder
  - Lav en fungerende mock-up med brug af grids og responsive teknikker
  - Lav en beskrivelse af løsningen til brug i læringsportfolio. Brug illustrationer og kommenteret kode
- Opgave 3 (Ekstra) Lav en responsiv udgave af dit nyhedssite fra i tirsdags