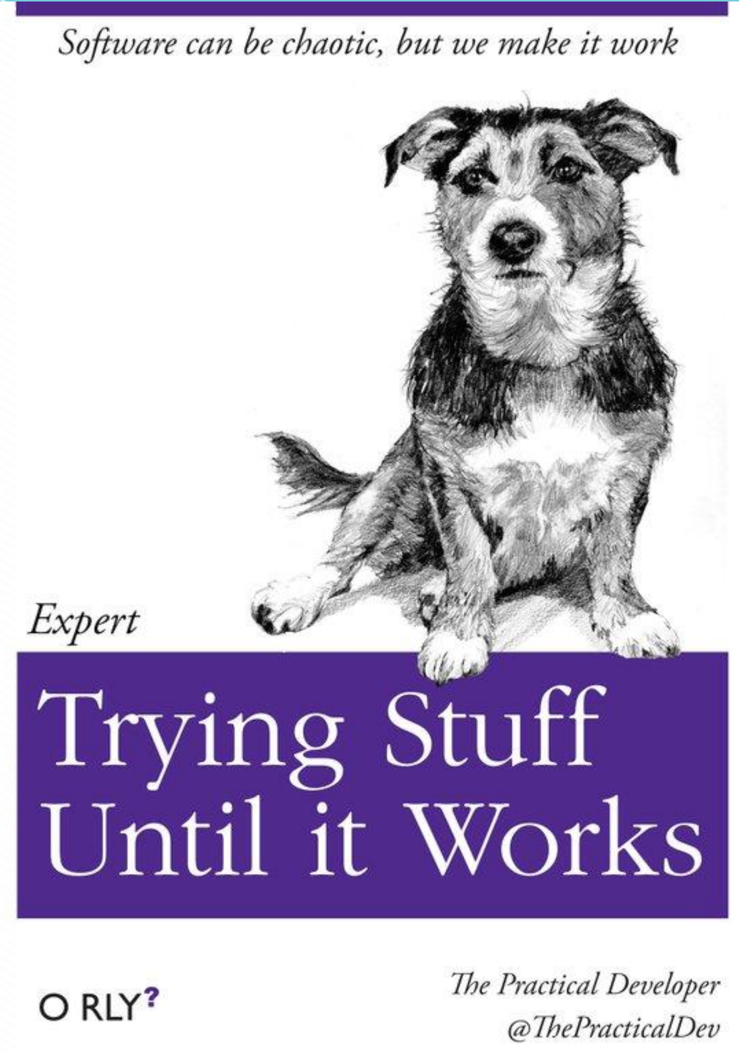
CSS Responsive Design

(I love how your website is impossible to read on my smartphone)

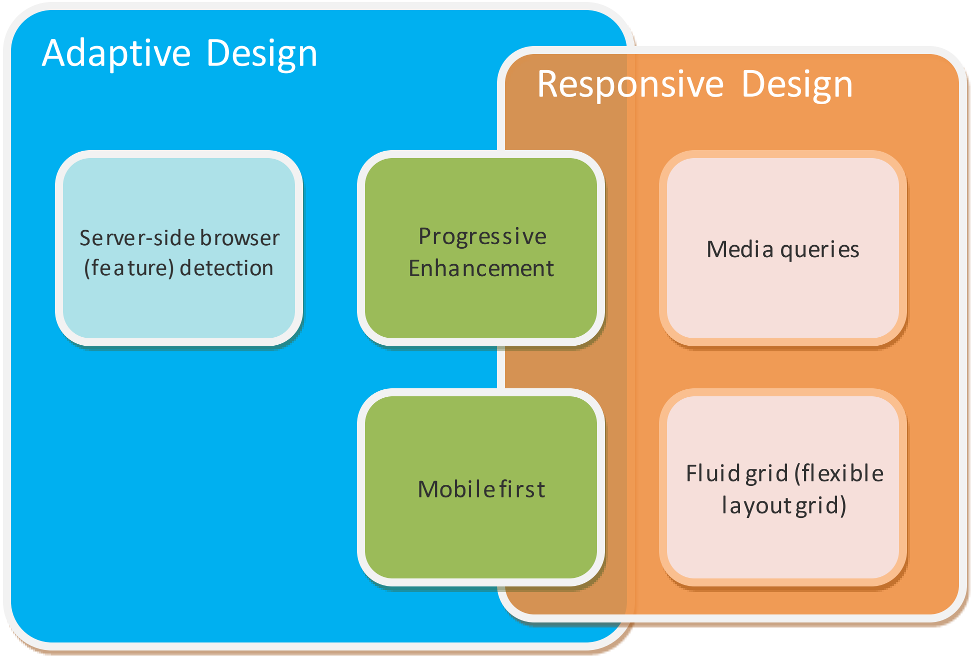
Frontend Junior Program - 2022


Software can be chaotic, but we make it work
Expert
Trying Stuff Until it Works
O RLY?
The Practical Developer
@ThePracticalDev

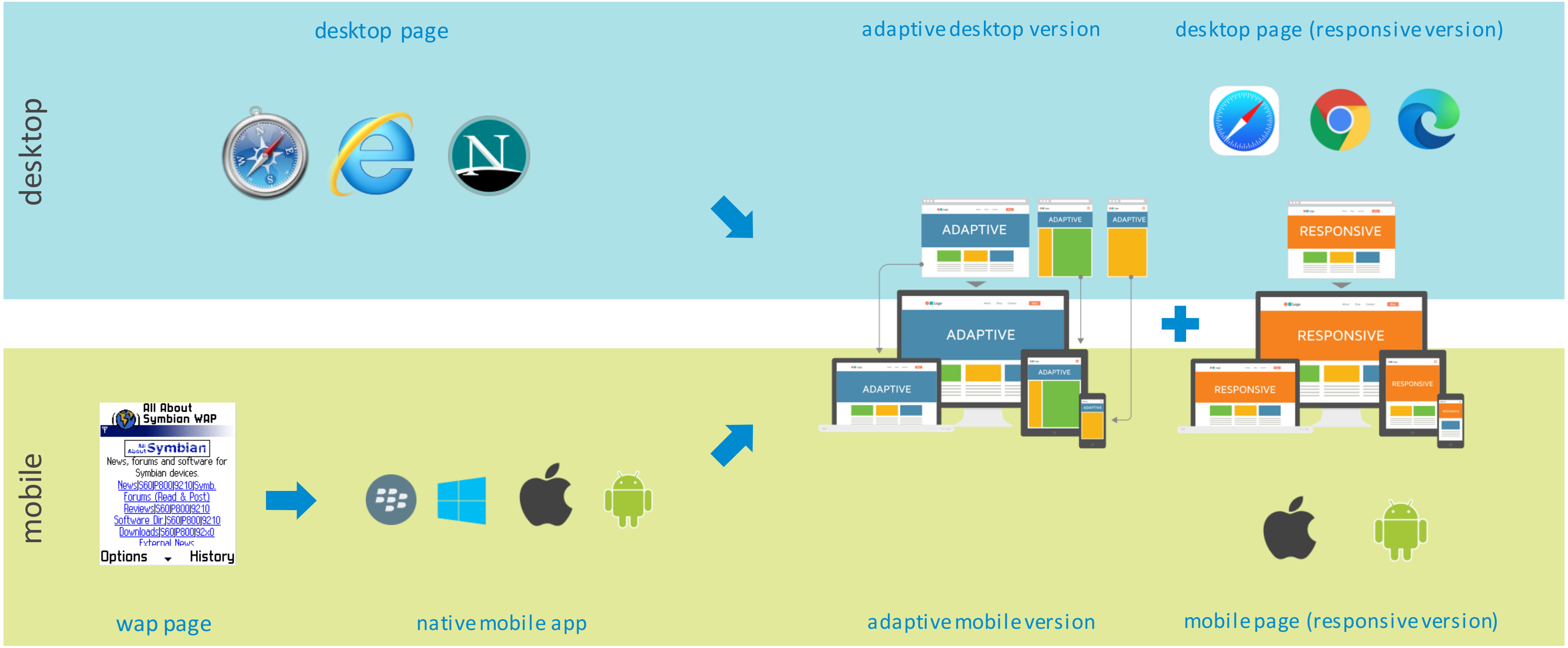
Agenda

Responsive Web Design & Adaptive Web Design

Both adapts the site design, the look and feel according to the device capabilities and viewport sizes to provide the best user experience suitable for every targeted media.



History



All About Symbian WAP

All About Symbian

News, forums and software for Symbian devices.

News|S60|P800|9210|Symb.
Forums (Read & Post)
Reviews|S60|P800|9210
Software Dir.|S60|P800|9210
Downloads|S60|P800|92x0
External News

Options History

With many clients…

we had an honor to experience the history

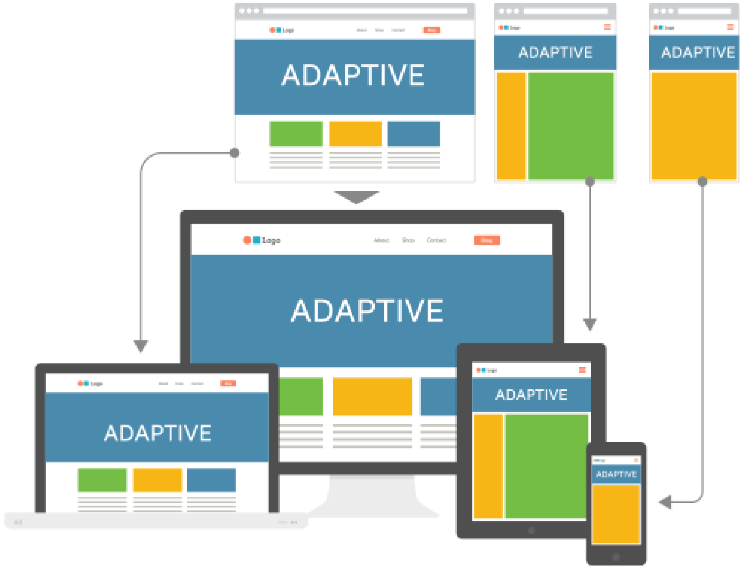
- we started with a desktop web site
- then a native mobile application emerged
- shortly after started the development of the adaptive version (vastly different from the desktop)
- and the story continued with a fully responsive site
- keeping the native mobile app alive (utilizing parts of the responsive site), still, providing a completely different user experience

Terminology

Terminology - Responsive (RWD) and Adaptive Web Design (AWD)

RESPONSIVE
RESPONSIVE
RESPONSIVE
RESPONSIVE
RESPONSIVE

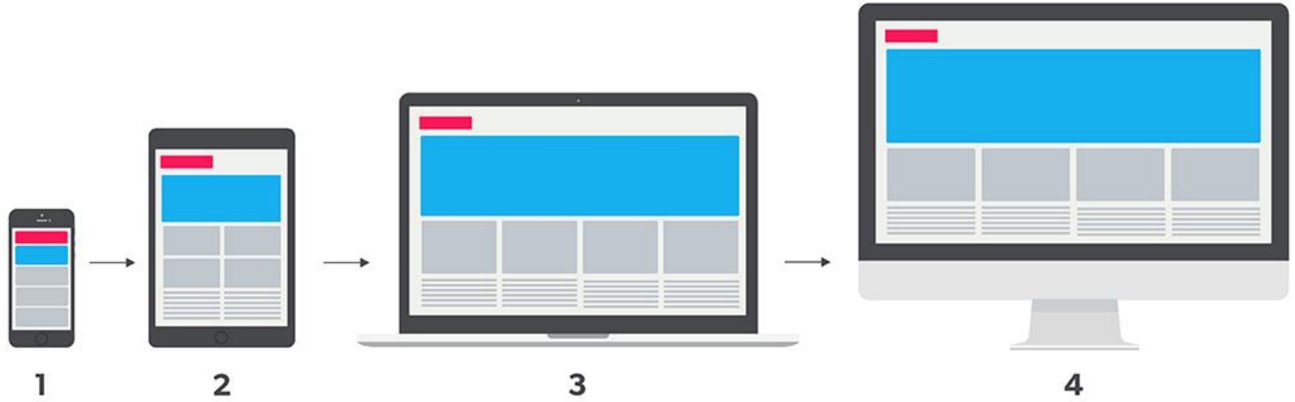
one site for every device



different solutions for different devices

Terminology - Mobile first

“Mobile will ultimately be the way you provision most of your services. The way I like to put it is, the answer should always be mobile first. You should always put your best team and your best app on your mobile app.”



[Eric Schmidt at Mobile World Congress: Mobile first]

Terminology - Mobile first

The term, mobile first could refer to many different concepts, or you may consider this in different levels, such as…

- a product principle: the business and the production team must focus on the mobile solution first
- design methodology: start with the mobile, and create other versions later based on the starting point - the mobile
- a low-level implementation approach: the first CSS is created for the mobile and other versions added with the media query *min-width*

A BOOK APART
Brief books for people who make websites
Nº 6
Luke Wroblewski
MOBILE FIRST
FOREWORD BY Jeffrey Zeldman

[Luke Wroblewski: MOBILE FIRST]

Terminology - Graceful degradation vs progressive enhancement

Progressive enhancement is a design philosophy that provides a baseline of essential content and functionality to as many users as possible, while delivering the best possible experience only to users of the most modern browsers that can run all the required code.

It can be said that both progressive enhancement and graceful degradation try to do the same thing: keep our products useful to every user. Graceful degradation can be used more easily as a patch for an already existing product; it means harder maintenance later on but requires less initial work.

Graceful degradation is a design philosophy that centers around trying to build a modern web site/application that will work in the newest browsers but falls back to an experience that while not as good still delivers essential content and functionality in older browsers.

MDN: Progressive Enhancement

MDN: Graceful degradation

W3C: Graceful degradation versus progressive enhancement

Graceful degradation

"experience that while not as good still delivers essential content and…"

"…functionality in older browsers."

Progressive enhancement

"…provides a baseline of essential content and functionality to as many users as possible, while delivering the best possible experience only to users of the most modern browsers that can run all the required code"

Graceful degradation and Progressive enhancement are still relevant?

Yes, we cannot support every possible browser. From the users (of unsupported browsers) perspective, the websites seem gracefully degraded. In projects, we provide support for these as a best effort.

For supported targets, the functionality and user experience is well defined and required to fulfil. This applies to browsers, devices, views, resolutions, portrait and landscape mode, network differences, etc.

Adaptive and Responsive Web Design

Adaptive Web Design (AWD)

ADAPTIVE
ADAPTIVE
ADAPTIVE
ADAPTIVE
ADAPTIVE
ADAPTIVE
ADAPTIVE

- Uses the server to detect the device / user / network the website is being viewed on (desktop, tablet, smartphone) and sends specific files for that device.
- It usually means different .css files, different images and many times different documents (HTML) with different features as well.
- While many parts can be shared between the variations, we need to develop (+ test and maintain) different versions of the same site.

Adaptive vs Adaptive

Even the term "Adaptive" are used in different ways: adaptive could mean adaptive in any way (on client side as well).

Here, we need to focus on solutions and technologies, i.e., the adaptivity could happen on

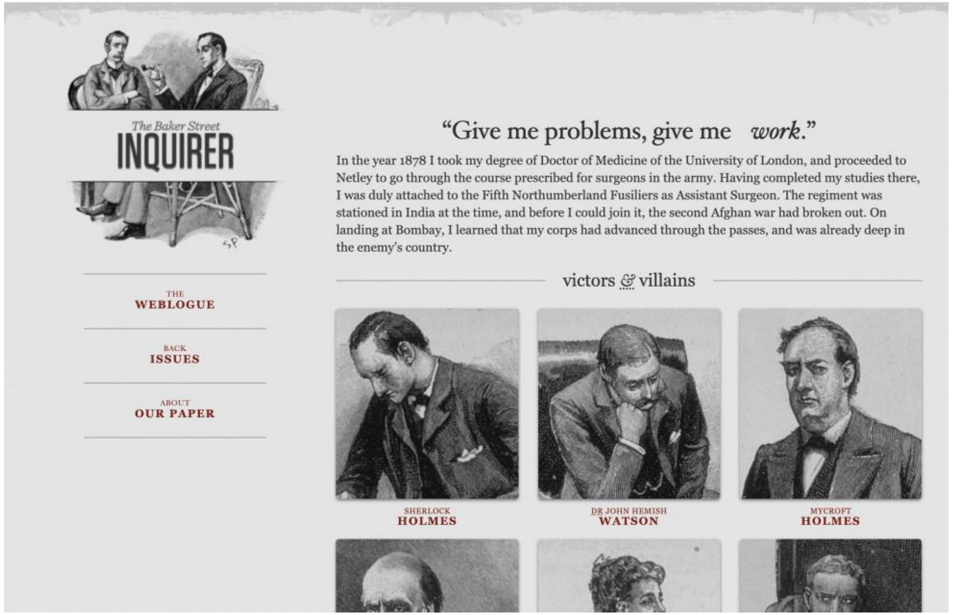
- backend, server-side
- client-side

and usually, it happens in both ways to an extent.

"Just one more thing…"

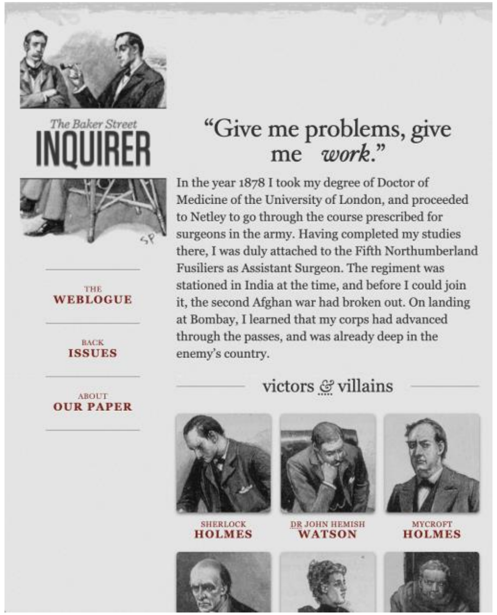
Flexible everything

"*Responsive Web Design*" by Ethan Marcotte, 2010 at alistapart.com



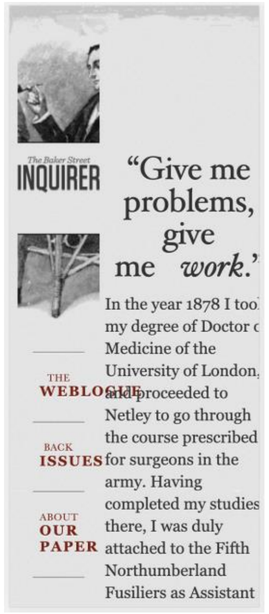
"Give me problems, give me *work.*"

In the year 1878 I took my degree of Doctor of Medicine of the University of London, and proceeded to Netley to go through the course prescribed for surgeons in the army. Having completed my studies there, I was duly attached to the Fifth Northumberland Fusiliers as Assistant Surgeon. The regiment was stationed in India at the time, and before I could join it, the second Afghan war had broken out. On landing at Bombay, I learned that my corps had advanced through the passes, and was already deep in the enemy's country.



"Give me problems, give me *work.*"

In the year 1878 I took my degree of Doctor of Medicine of the University of London, and proceeded to Netley to go through the course prescribed for surgeons in the army. Having completed my studies there, I was duly attached to the Fifth Northumberland Fusiliers as Assistant Surgeon. The regiment was stationed in India at the time, and before I could join it, the second Afghan war had broken out. On landing at Bombay, I learned that my corps had advanced through the passes, and was already deep in the enemy's country.



"Give me problems, give me *work.*'

In the year 1878 I too
my degree of Doctor c
Medicine of the
University of London,
and proceeded to
Netley to go through
the course prescribed
for surgeons in the
army. Having
completed my studies
there, I was duly
attached to the Fifth
Northumberland
Fusiliers as Assistant

well, always test your site on all the required devices / screen dimensions ;)

Responsive Web Design

Uses the client-side to detect the device features (e.g., width, resolution) to utilize specific content (CSS, media) providing different user experience for different devices / viewport widths.

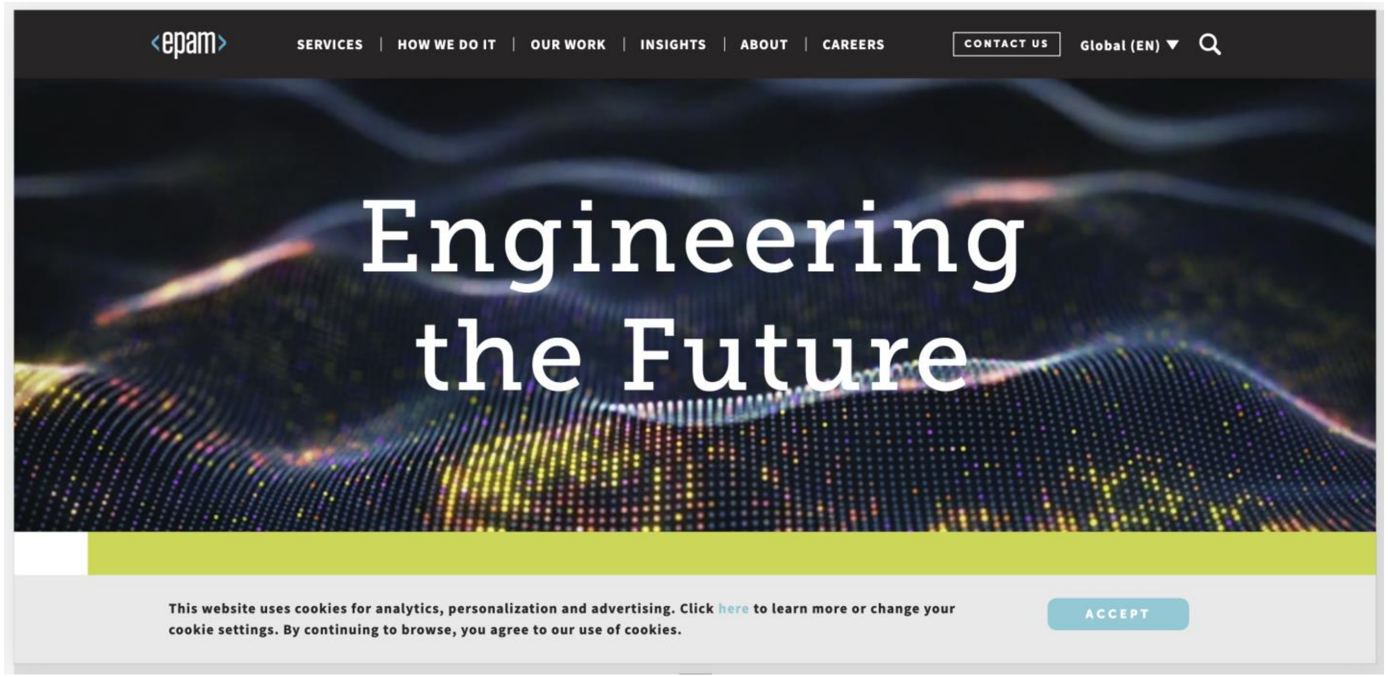
Do we develop responsive sites?

Yes, almost exclusively. Even the native mobile applications (iOS, Android) use components and pages of the responsive version (e.g., in a webview).

What about adaptive sites?

Just to complicate things, our solutions utilize parts of the adaptive approach as well: server-side rendered pages for SEO, server-side image optimization, even different versions served depending on a user group (e.g., based on geolocation).

Responsive Design Example – desktop

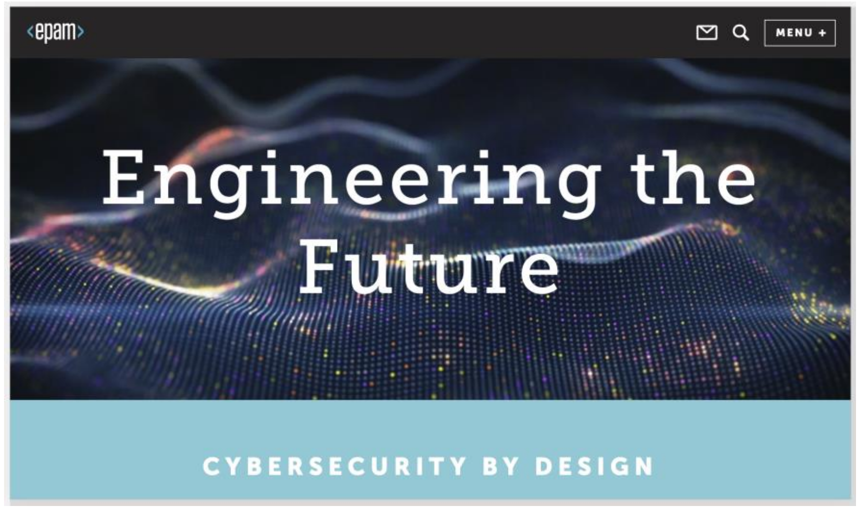


This website uses cookies for analytics, personalization and advertising. Click here to learn more or change your cookie settings. By continuing to browse, you agree to our use of cookies.

full experience

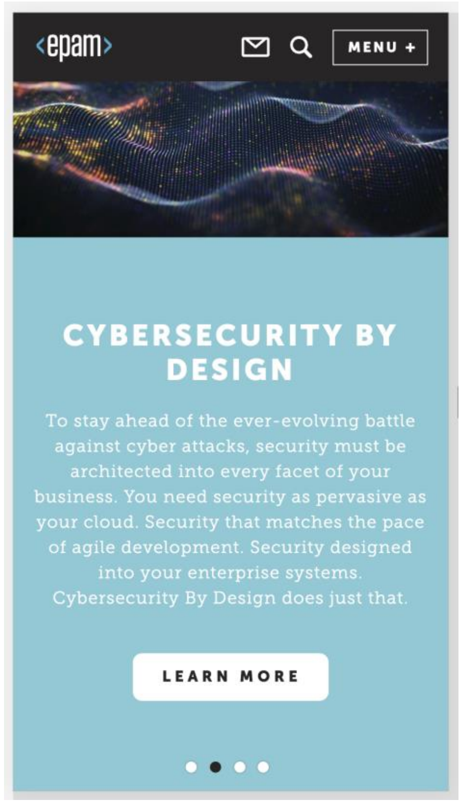
“Here, take a cookie. I promise, by the time you're done eating it, you'll feel right as rain.”

Responsive Design Example – mobile views

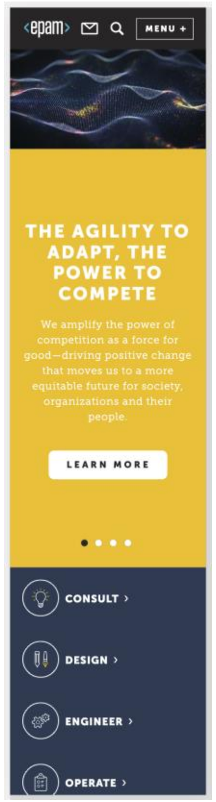


tablet view

menu is hidden under the menu button



To stay ahead of the ever-evolving battle against cyber attacks, security must be architected into every facet of your business. You need security as pervasive as your cloud. Security that matches the pace of agile development. Security designed into your enterprise systems. Cybersecurity By Design does just that.



We amplify the power of competition as a force for good—driving positive change that moves us to a more equitable future for society, organizations and their people.

mobile views

different widths – different views

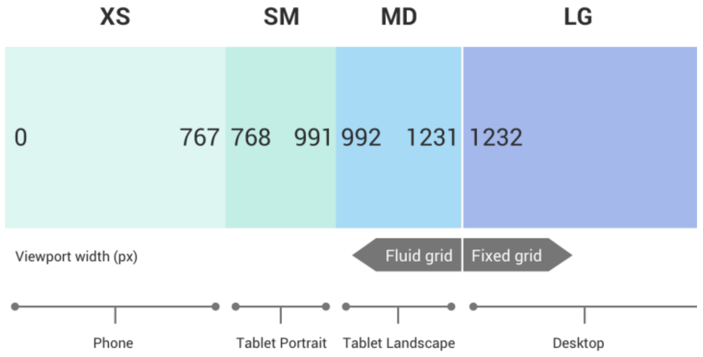
Goals

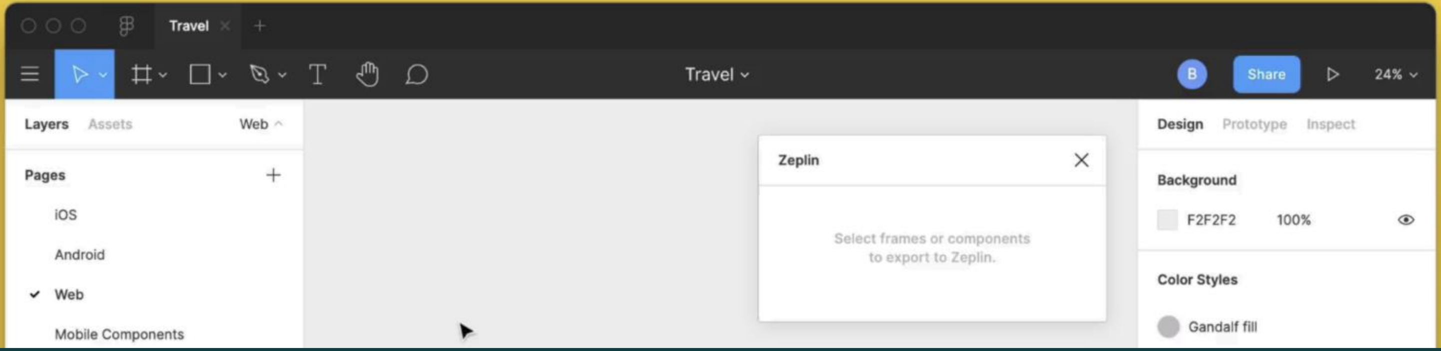
Views

Views and breakpoints

While we are talking about well defined views (at breakpoints, let's discuss this later), the transition between views should be flawless: the site should look intact at every device width.

These different views are defined in the Style Guide and in the UX Design.





So, *what you're saying* is that not all views for every pixel width are well defined?

Well, there are no 1024 different designs for a site ;)

What we have is some distinct views – sometimes just a desktop and a mobile view – and it is the responsibility of the developer to make sure that it will look perfect at every possible width.

Usually, the developers work in a close collaboration with the designer team and the client to clarify details regarding this.

Because it's not a trivial task by any means, there is a room for creativity for visually sensitive and talented developers.

Automatic page rescale on mobile devices

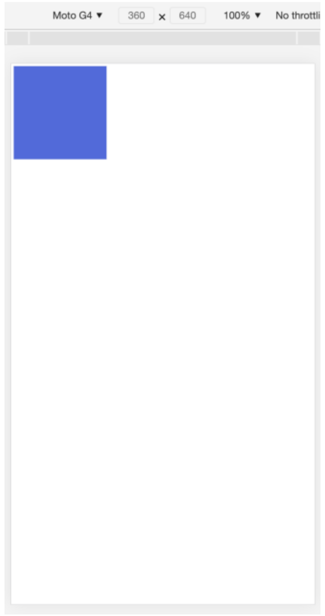
The browser's viewport is the area of the window in which web content can be seen.

On smartphones pages automatically rescale to fit the tiny screen. A full-sized page, unless specified otherwise, would just shrink proportionally for the tiny browser. Then, the user could easily zoom in and out as necessary.

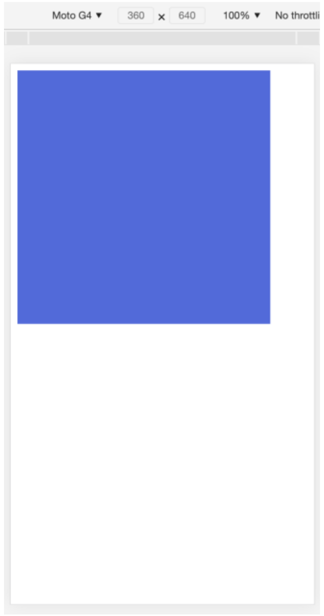
However, if your page style is optimized for mobile devices, we need to tell the mobile device to display the page without any zooming:

```html
<html>
<head>
  <meta charset="utf-8" />
  <meta name="viewport" content="width=device-width,initial-scale=1">
  <title>Document title</title>
</head>

<body style="width:960px;">
  <div style="width:300px; height: 300px; background: royalblue">
  </div>
</body>
</html>
```



without the meta…

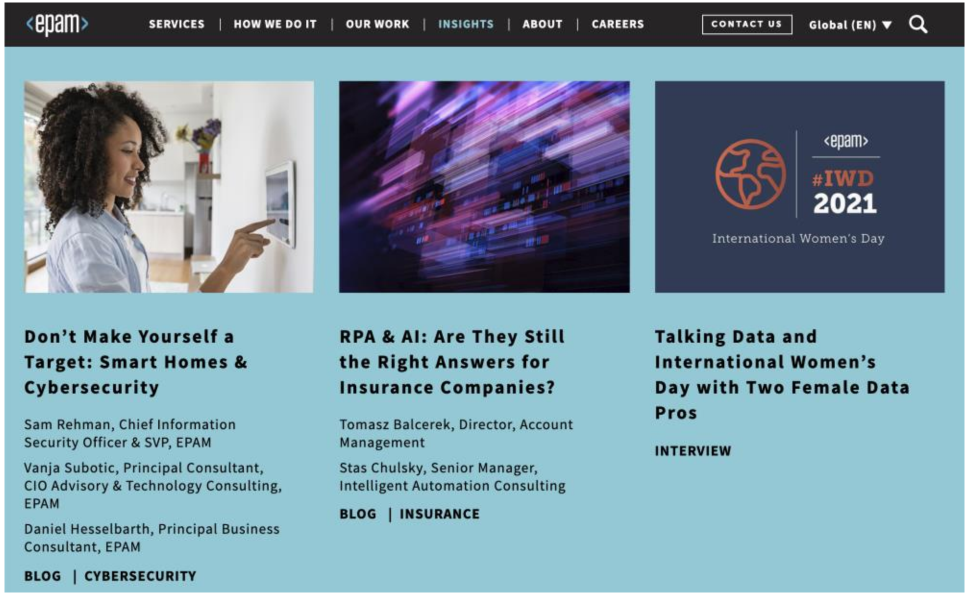


with the meta…

[MDN: Using the viewport meta tag]

Custom layout structure

We may want to change the layout, either through a separate style sheet or, more efficiently, through a CSS media query.



<epam>
SERVICES | HOW WE DO IT | OUR WORK | INSIGHTS | ABOUT | CAREERS
CONTACT US
Global (EN)

Don't Make Yourself a Target: Smart Homes & Cybersecurity

Sam Rehman, Chief Information Security Officer & SVP, EPAM

Vanja Subotic, Principal Consultant, CIO Advisory & Technology Consulting, EPAM

Daniel Hesselbarth, Principal Business Consultant, EPAM

BLOG | CYBERSECURITY

RPA & AI: Are They Still the Right Answers for Insurance Companies?

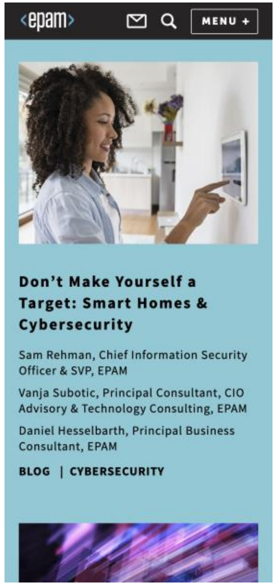
Tomasz Balcerek, Director, Account Management

Stas Chulsky, Senior Manager, Intelligent Automation Consulting

BLOG | INSURANCE

Talking Data and International Women's Day with Two Female Data Pros

INTERVIEW



Don't Make Yourself a Target: Smart Homes & Cybersecurity

Sam Rehman, Chief Information Security Officer & SVP, EPAM

Vanja Subotic, Principal Consultant, CIO Advisory & Technology Consulting, EPAM

Daniel Hesselbarth, Principal Business Consultant, EPAM

BLOG | CYBERSECURITY

Touchscreens vs. cursors

Touchscreens obviously come with different design principles than a cursor-based interaction* and have different capabilities as well.

Touchscreens have no capability to display CSS *:hover*** because there is no cursor; once the user touches the screen, they click.

** For accessibility reasons, it won't hurt if users are able to interact with your site with keyboard only, as well*

*** Well, iOS could present surprises here*

T'hover, or not t'hover, that i*o*s the question

[MDN: :hover]

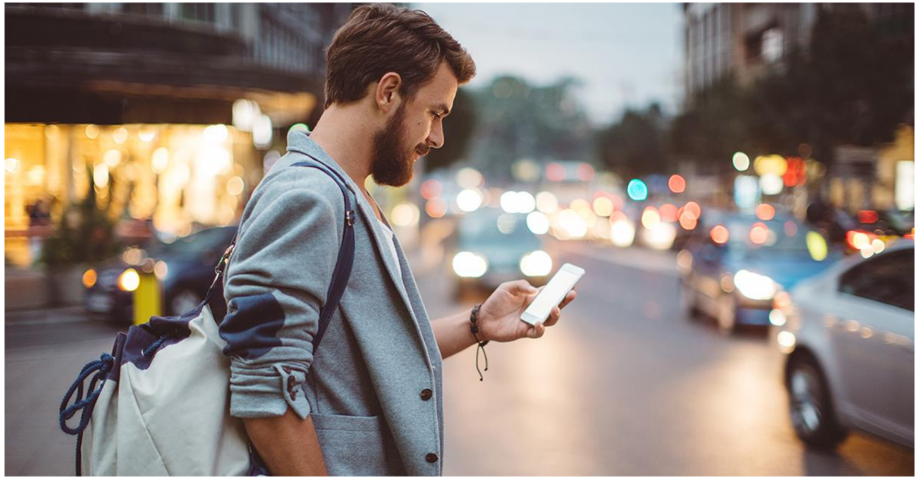
Performance

The primary goal of mobile stylesheets is to alter the layout for a smaller display and to reduce the bandwidth need* for slower mobile networks.

** That is far from a trivial task. Think about retina images and extreme mobile resolutions.*

If the site chooses images (from an image set) according to the screen resolution, it may happen that a mobile will require images with higher resolution than a desktop site.

A possible solution for this problem could be to alter the compression factor for mobile images, providing images with higher resolution, but smaller file size. The compression artifacts are not that visible than the loss in quality due to the lower resolution.


mobile on the go: could be very challenging to achieve good performance here

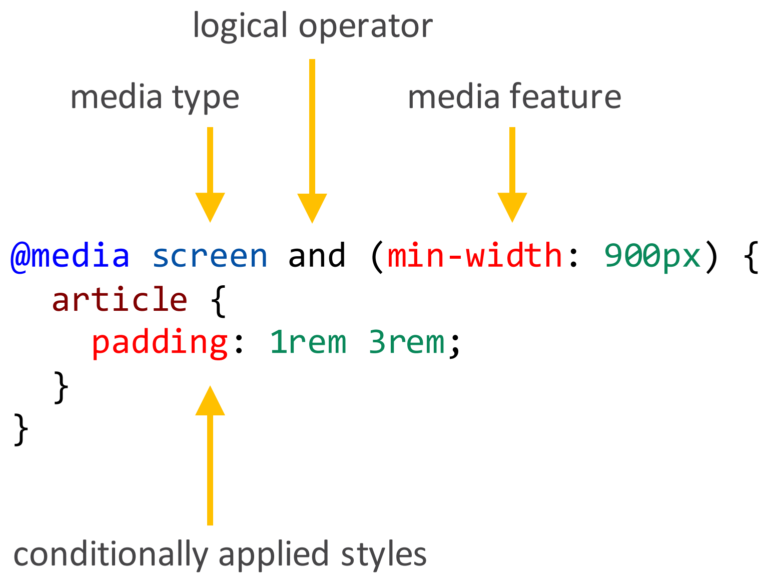
MEDIA QUERIES

Media queries

Media queries let you adapt your site or app depending on the presence or value of various device characteristics and parameters.

They are a key component of responsive design. For example, a media query can shrink the font size on small devices, increase the padding between paragraphs when a page is viewed in portrait mode, or bump up the size of buttons on touchscreens.

In CSS, use the @media at-rule to conditionally apply part of a style sheet based on the result of a media query. Use @import to conditionally apply an entire style sheet.



```css
@media screen and (min-width: 900px) {
  article {
    padding: 1rem 3rem;
  }
}
```

MDN: Media queries

MDN: Using media queries

Media Types

all	*Suitable for all devices*
print	*Intended for paged material and documents viewed on a screen in print preview mode.*
screen	*Intended primarily for screens.*
speech	*Intended for speech synthesizers.*

```
@media screen and (min-width: 900px) {
  article {
    padding: 1rem 3rem;
  }
}
```

[MDN: Media types]

Logical operators

and	*Used for combining multiple media features together into a single media query. It is also used for joining media features with media types.*
not	*Used to negate a media query, returning true if the query would otherwise return false.*
only	*Prevents older browsers that do not support media queries with media features from applying the given styles. It has no effect on modern browsers.*
,	*Commas are used to combine multiple media queries into a single rule.*

```
@media screen and (min-width: 900px) {
  article {
    padding: 1rem 3rem;
  }
}
```

[MDN: Logical operators]

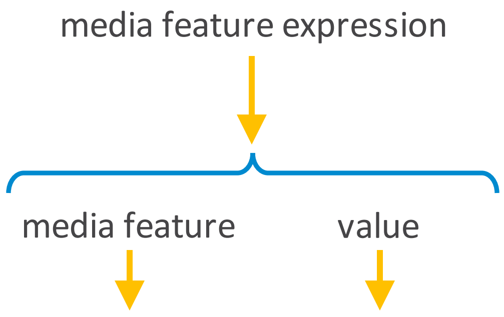
Media Features

Media features describe specific characteristics of the user agent, output device, or environment. Media feature expressions test for their presence or value and are entirely optional.

Each media feature expression must be surrounded by parentheses.

While there are many different media features, you will predominantly use the

width	width of the viewport including width of scrollbar



```
@media screen and (min-width: 900px) {
  article {
    padding: 1rem 3rem;
  }
}
```

MDN: Media features

Media Features - width

Basically, it is used it as min-width and max-width

breakpoint

```
/* Minimum width */
@media (min-width: 320px) {
  div {
    background: yellow;
  }
}
```

breakpoint

```
/* Maximum width */
@media (max-width: 768px) {
  div {
    border: 2px solid blue;
  }
}
```

The width CSS media feature can be used to test the width of the viewport (or the page box, for paged media).

The width feature is specified as a <length> value representing the viewport width. It is a range feature, meaning that you can also use the prefixed min-width and max-width variants to query minimum and maximum values, respectively.

MDN: @media/width

Breakpoints

Breakpoints will determine the layout at particular viewport widths.

The more breakpoint you create, the more ~~bugs~~ control you have over the user experience.

```
/* Small devices (landscape phones, 576px and up) */
@media (min-width: 576px) { ... }

/* Medium devices (tablets, 768px and up) */
@media (min-width: 768px) { ... }

/* Large devices (desktops, 992px and up) */
@media (min-width: 992px) { ... }

/* Extra large devices (large desktops, 1200px and up) */
@media (min-width: 1200px) { ... }
```

MDN: How to choose breakpoints

Breakpoints

```
/* Minimum width */
@media (min-width: 320px) {
  div {
    background: yellow;
  }
}
```

← mobile first approach

```
/* Maximum width */
@media (max-width: 768px) {
  div {
    border: 2px solid blue;
  }
}
```

← desktop first approach

```
/* Maximum width */
@media (min-width: 320px) and (max-width: 768px) {
  div {
    border: 2px solid blue;
  }
}
```

← range

Why this is mobile first?

Because the default css code – without any media queries – will provide the narrower view, the mobile.

Please don't confuse it with the design / product principle, though. You can use max-width queries even when your first target is the mobile.

CSS Media Queries Test

What media query to use then?

Any approach (using min-width, max-width, range, and combinations of these) could be perfectly viable.

In projects, however, we usually use @mixins (or similar tools), to incorporate breakpoints, which defines the values and usages.

That being said, it is safer to use *min-widths* or *max-widths* exclusively, and use *range* only when it is needed.

We suggest not mixing *min-widths* and *max-widths.*

CSS pixel

"Just one more thing…"

this is not in device pixels, this is in *CSS (logical) pixels*

```
@media screen and (min-width: 900px) {
  article {
    padding: 1rem 3rem;
  }
}
```

MDN: CSS pixel

RESPONSIVE IMAGES

Retina displays

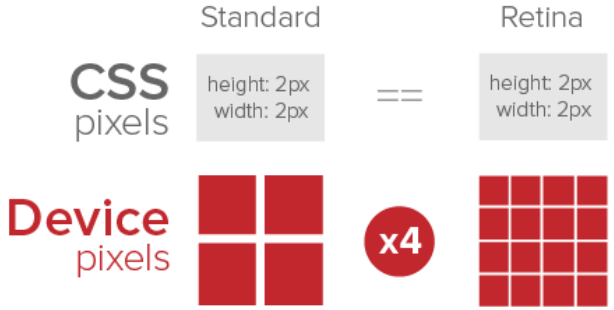
Traditionally, it *was* true that CSS pixel = device pixel.

However, with the high-resolution retina devices this has been changed.

Nowadays, it is true for almost all mobile devices that

CSS pixel ≠ device pixel

In media queries, it makes sense to use CSS pixels, so that the site could look similar on every devices (with the same display size), regardless of the resolution.



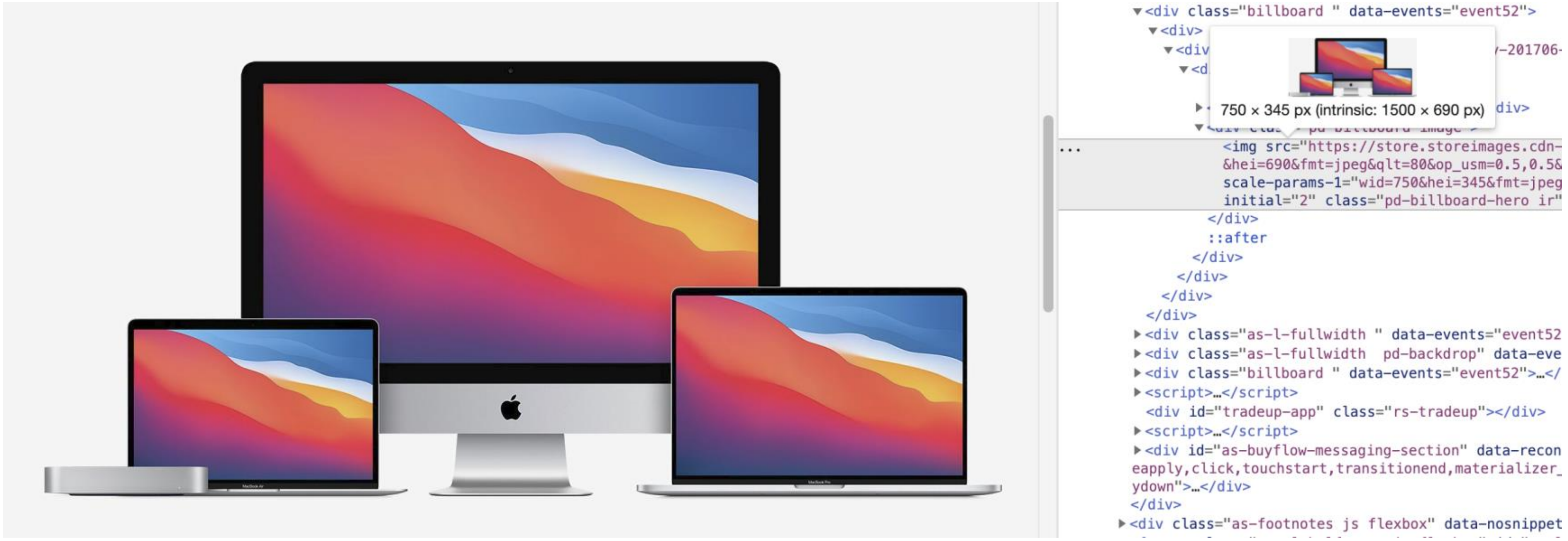
MDN: CSS Length Explained

Retina on desktop (HiDPI)

It also means, that these kind of images look breath-taking on your site only, if you use high resolution images.

Retina on desktop (HiDPI)

This is how it is done: the original image is higher resolution than the box itself



750 × 345 px (intrinsic: 1500 × 690 px)

```
▼<div class="billboard " data-events="event52">
  ▼<div>
    ▼<div                                         /-201706-
      ▼<d
        ▶                                        div>
        ▼
          <img src="https://store.storeimages.cdn-
          &hei=690&fmt=jpeg&qlt=80&op_usm=0.5,0.5&
          scale-params-1="wid=750&hei=345&fmt=jpeg
          initial="2" class="pd-billboard-hero ir"
         </div>
         ::after
        </div>
      </div>
    </div>
  </div>
▶<div class="as-l-fullwidth " data-events="event52
▶<div class="as-l-fullwidth  pd-backdrop" data-eve
▶<div class="billboard " data-events="event52">…</
▶<script>…</script>
 <div id="tradeup-app" class="rs-tradeup"></div>
▶<script>…</script>
▶<div id="as-buyflow-messaging-section" data-recon
eapply,click,touchstart,transitionend,materializer_
ydown">…</div>
</div>
▶<div class="as-footnotes js flexbox" data-nosnippet
```

Image as a background - by Apple

```css
.main [data-module-template="heroes"] .unit-image-wrapper .unit-image.unit-image-iphone-12-iphone-12-us {
  width: 3010px;
  height: 624px;
  background-size: 3010px 624px;
  background-repeat: no-repeat;
  background-image: url("/v/home/x/images/heroes/iphone-12/iphone_12_us__fo0stbby242m_large.jpg")
}

@media only screen and (-webkit-min-device-pixel-ratio: 1.5), only screen and (min-resolution: 1.5dppx), (-webkit-
min-device-pixel-ratio: 1.5), (min-resolution: 144dpi) {
  .main [data-module-template="heroes"] .unit-image-wrapper .unit-image.unit-image-iphone-12-iphone-12-us {
    background-image:url("/v/home/x/images/heroes/iphone-12/iphone_12_us__fo0stbby242m_large_2x.jpg")
  }
}

@media only screen and (max-width: 1068px) {
  .main [data-module-template="heroes"] .unit-image-wrapper .unit-image.unit-image-iphone-12-iphone-12-us {
    width:1068px;
    height: 617px;
    background-size: 1068px 617px;
    background-repeat: no-repeat;
    background-image: url("/v/home/x/images/heroes/iphone-12/iphone_12_us__fo0stbby242m_medium.jpg")
  }
}

@media only screen and (max-width: 1068px) and (-webkit-min-device-pixel-ratio: 1.5), only screen and (max-width:
1068px) and (min-resolution: 1.5dppx), only screen and (max-width: 1068px) and (min-resolution: 144dpi) {
  .main [data-module-template="heroes"] .unit-image-wrapper .unit-image.unit-image-iphone-12-iphone-12-us {
    background-image:url("/v/home/x/images/heroes/iphone-12/iphone_12_us__fo0stbby242m_medium_2x.jpg")
  }
}

@media only screen and (max-width: 734px) {
  .main [data-module-template="heroes"] .unit-image-wrapper .unit-image.unit-image-iphone-12-iphone-12-us {
    width:734px;
    height: 548px;
    background-size: 734px 548px;
    background-repeat: no-repeat;
    background-image: url("/v/home/x/images/heroes/iphone-12/iphone_12_us__fo0stbby242m_small.jpg")
  }
}
```

this is just a part
of the css code of 1 image

Responsive images - srcset

```
<img sizes="(max-width: 30em) 100vw, (max-width: 50em) 50vw"

     srcset="rowing-200.jpg 200w, rowing-400.jpg 400w,
             rowing-800.jpg 800w, rowing-1600.jpg 1600w"

     src="rowing-400.jpg"

     alt="Twilight Voyage">
```

One or more strings separated by commas, indicating possible image sources for the user agent to use. Each string is composed of:

1. *A URL to an image*
2. *Optionally, a whitespace followed by one of:*
   - *A width descriptor (a positive integer directly followed by w). The width descriptor is divided by the source size given in the sizes attribute to calculate the effective pixel density.*
   - *A pixel density descriptor (a positive floating-point number directly followed by x).*

The user agent selects any of the available sources at its discretion. This provides them with significant leeway to tailor their selection based on things like user preferences or bandwidth conditions.

Dev.Opera: Native Responsive Images

MDN: Responsive images

MDN: <img> The Image Embed element, srcset

Q&A

edu_hu@epam.com