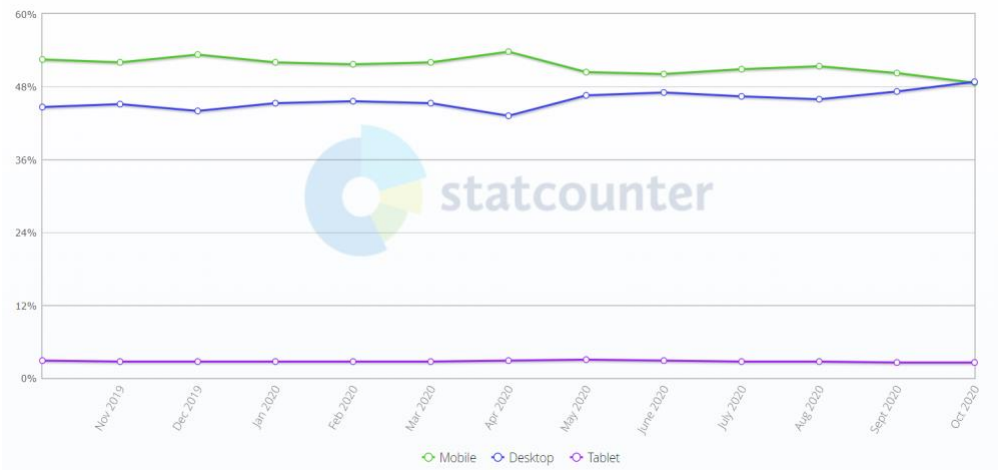
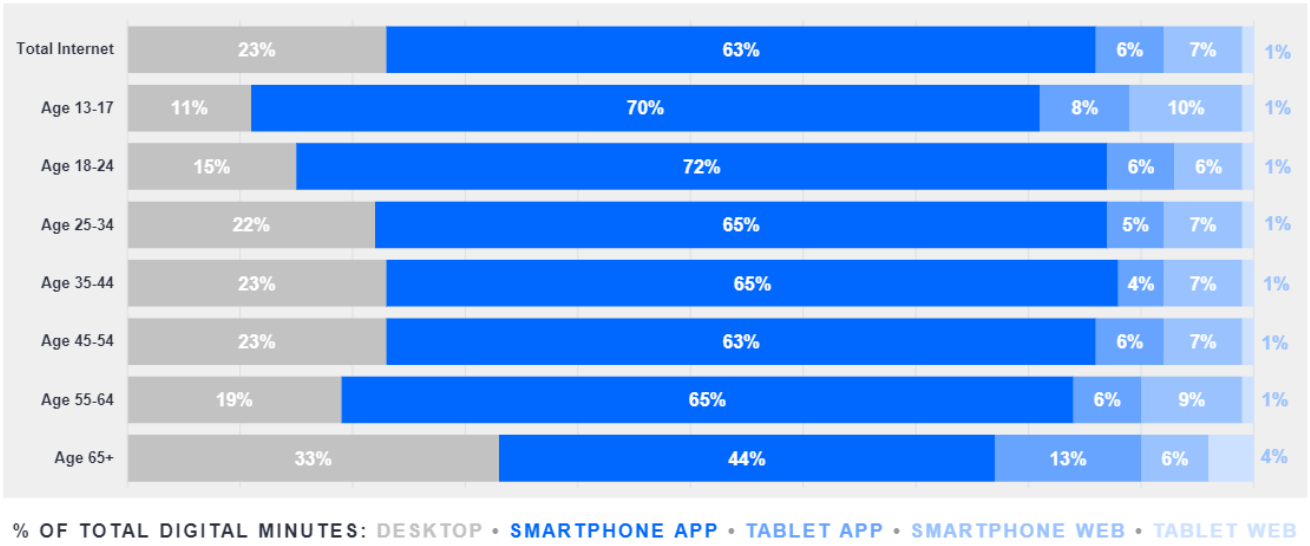
Why Mobile Web Development?

The question can also be: Why should I develop a web site for mobile devices?



The image on the right was taken in Nov/2020 from the gs.statcounter.com website. Check the image below:



Digital platform usage by age - young people spend 72% of their time in apps.

If we think about our own lives, most probably our third screen is a mobile (after TV and desktop/laptop). According to the bar graphic presented above (from Simon Hearne – Web Performance Consultant), young people spend 72% of their time in apps, meaning, in their mobile devices (smartphones)!

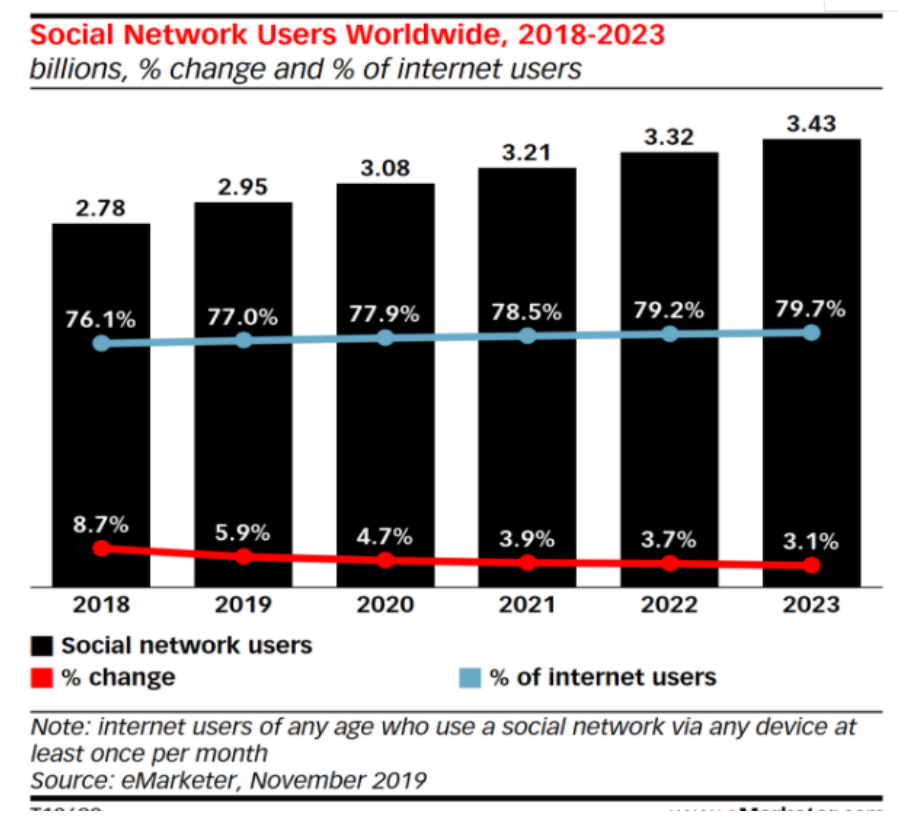
At the end of 2014, studies were showing that between 2008 and 2012 the internet adoption had increased a lot at a global level. In 2012, the number of users reached 2.4 billion while in 2007 that number was 1.3 billion (very close to 100% growth). According to a study from Cisco, mobile data traffic grew 81% which translates to nearly 18 times the size of the entire global Internet in 2000!

These two pictures above show some important reasons that web developers should be targeting mobile devices even before they think about desktop/laptop.

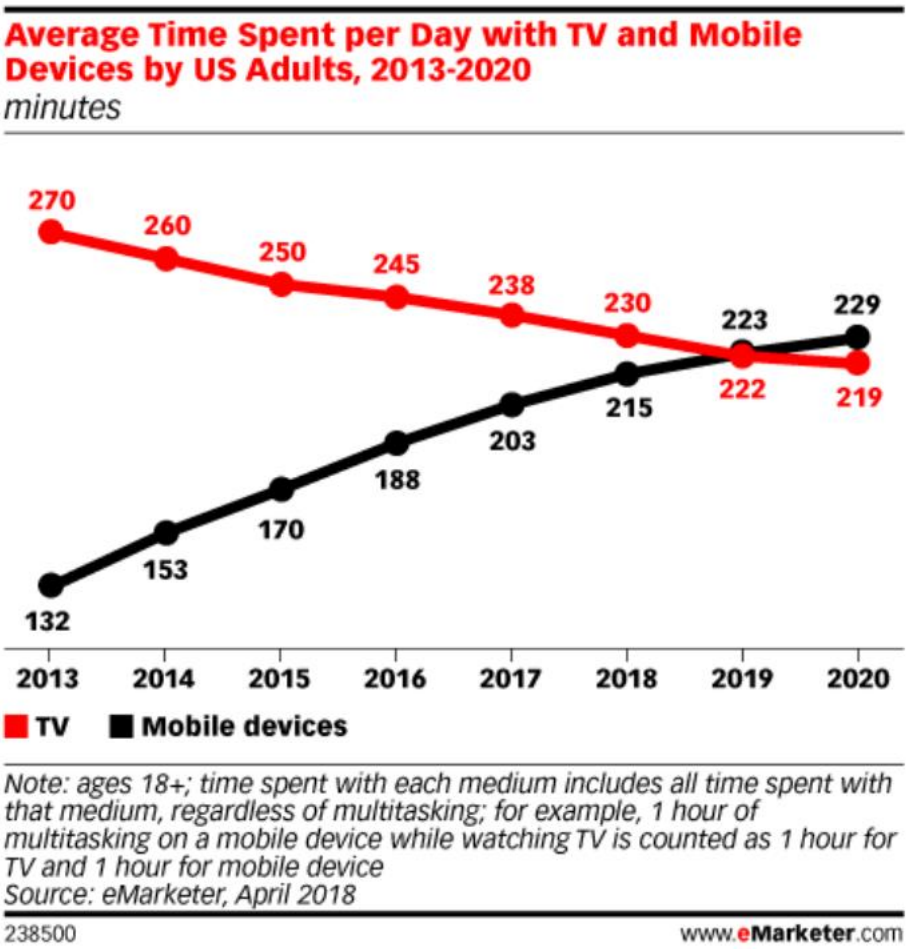
But besides the numbers shown in the graphic above, it's easy to verify that it's not possible to have 100% of the same desktop site presented in mobile devices due to incompatibilities, bandwidth, different browser (lots!!!).

Do you need other numbers? Here we go… The numbers below come from many different studies!

- 91% of people own a mobile phone
- 56% of people own a smartphone
- 50% of mobile phone users use their devices as their primary internet source (most of the time to activate or use an app)
- 72% of people claim that their smartphone is in their immediate vicinity most of the time
- Apple device owners spent more than $20 billion in the App Store in 2015
- Candy Crush Saga players spent over $1.3 billion on in-app purchases in 2014
- 44% of cell phone owners have slept with the phone next to their bed as they do not want to miss any calls, text messages, or even updates during the night
- 35% use tablet in the bathroom
- About 1.5 million Android devices activated every single day
- In the first 3 months of 2015, Apple sold 61 million iPhones in UK
- The total number of mobile app downloads in 2017 – 197 billion – and the app category people spent time the most was social networking (around 26%)
- Mobile social networking keeps growing – see numbers in the picture below



The mobile device growth happened because of the apps developed for them at a point that currently users are spending more time using apps then watching TV – see the image below:



Web developers need to always pay attention to the evolution in the market and new trends. There are many articles that are released every year discussing about new UI/UX and/or design trends in web development, such as the one from Blue Compass – Web Design Trends to Watch for.

We want to create web sites that users feel comfortable navigating no matter what device they are using and it's important to know that about 85% of users think that a company's mobile site should be as good or better than the desktop/laptop version – you can read this article with some other Eye-opening website statistics.

Some of us have just studied and/or become more experienced using HTML, CSS, JavaScript and then, suddenly, we start listening people talking about other technologies such as Java (coming back…), Objective C, Swift, etc. That's when we may get confused and start thinking about updating and/or switching the "plugs of our brain" towards learning those new technologies thinking that the ones we have been using for so long are "dead".

NO!!! They are not! For sure, if we think about versatility, it's always good to know about these new technologies and even study them so you will be more prepared to even suggest directions at your place of work and to your own studies as well.

When you hear the term **native apps**, do not think they are the only solution available for mobile devices with only advantages. Mobile users tend to prefer native apps (they can download from the app store) and the native apps generally define the look-and-feel of your device (the icons, what you choose to be in the home screen, etc.). Besides that, they can work very well with the many different hardware capabilities of the devices they are designed to. They tend to be faster than web apps.

The disadvantage of native apps is that they are developed toward a single platform (OS-device), using different programming language which means that if your company decides to develop a native app, it will need to have basically two different groups of programmers – one for iOS and one for Android (not even considering here the other different platforms such as: Windows, Blackberry, etc.). The learning curve is steep as the developer needs to learn a new programming language. Once the native app is developed, the company needs to get the approval to publish it in the app store and this process may be somewhat long (Apple is a good example of a long and sometimes frustrating process).

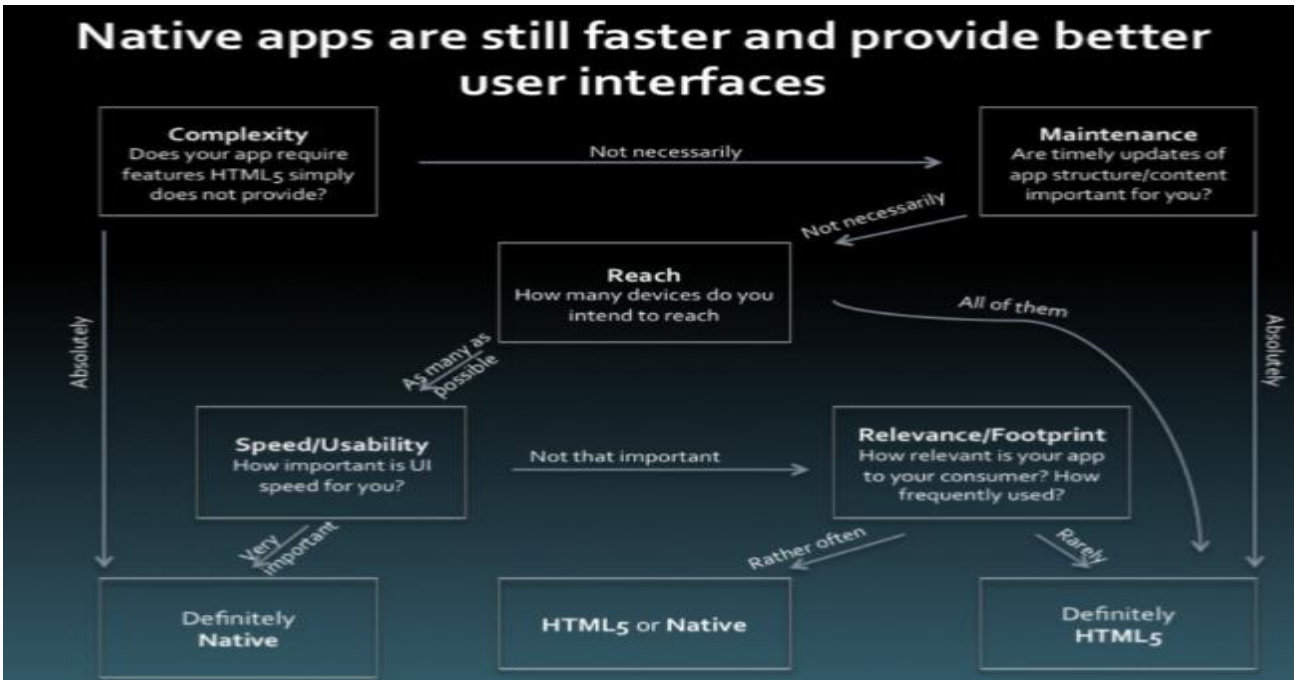
Of course, there will be instances when developing a native app will make more sense than developing a web app – one great article about this is from Apiko – PWA vs Native Apps: How to Choose your Technology.

After HTML5 came into this arena, the term **web apps** started to grow and come as another possible solution of having customers accessing your data, your site, in their mobile devices. Web apps need the browser to run but their main advantage is exactly that they are apps that can look very similar to their native "friends" developed using well-known technologies such as HTML5, CSS, and JavaScript. This brings another advantage which is the fact that the company is developing a single app, with one code, for all different platforms. One other good aspect of web apps is that they do not need store approval to be "published".

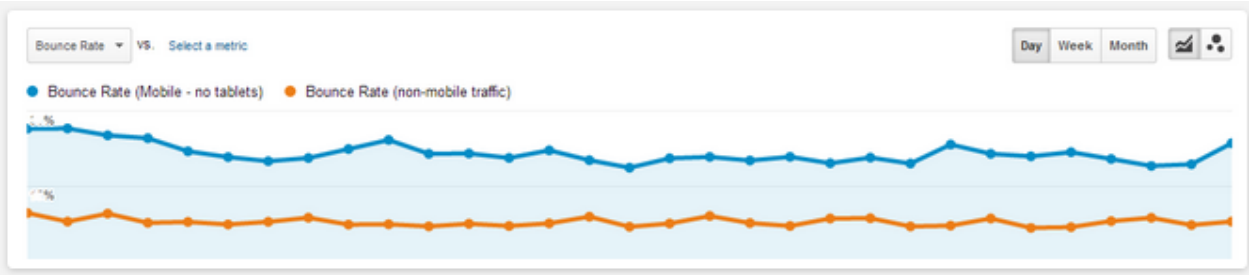
Of course, nothing comes with only the good parts! HTML5 faces its own fragmentation problem (too many different browsers in the mobile world – if it was already a headache to develop websites for desktops/laptops to work well with Chrome, Safari, Firefox, IE, imagine with so many different browsers available for mobile devices). The offline feature is still in its infancy (getting better as you read this). HTML5 still does not access many of the device features and, last, but not least, it is hard to monetize from web apps

as there is not central app store for them (sometimes developers use the hybrid app approach such as PhoneGap).

Here is a good graphic to help you decide which way to go:



If even with this graphic, you are still in doubt and your question is: I am not sure how many customers are accessing my site from a mobile device; you can then think about using some tools to help you get those numbers during a period. One of those tools is Google Analytics – you need to create an account and insert the Google Analytics code in your web pages and let it run for some time to collect meaningful data. Google Analytics provides very good data differentiating mobile devices traffic from desktop/laptop traffic. For example, you can get a graphic like the one shown below:



This graphic shows clearly that the site has a higher bounce rate for mobile devices – is the layout of the site appropriate for mobile? Do the users need to zoom/pinch to read the content?

One other fact to help convince you about the importance of developing for mobile first – on April 21st, 2015 Google search engine had its algorithm changed to rank pages based also on the ability of being or not mobile-friendly – here is the link they provided so people could test their websites – Mobile Friendly Test.

If you also need to know what is the browser mostly used by your target audience, you can have some idea in Statcounter website that keeps, up to the previous month, the statistics of desktop/laptop browsers, mobile browsers and other statistics.

What is mobile? You might think: I have read a lot of things and now this question? Well, you need to be aware that what you consider mobile might not be exactly what some authors consider to be mobile. For most of the authors, mobile means portable, personal, a companion, with easy usage, and with some type of network connection. Some authors consider tablets to be in-between mobile devices and desktop/laptops. Some authors are more specific and divide the mobile devices in categories such as:

- Mobile phones – calls and SMS
- Low-end mobile devices – basic browsing capability
- Mid-end mobile devices – camera, 3G, browser
- High-end mobile devices – good camera, Bluetooth, better internet range, 4G up
- Smartphones – GPS, video camera, touch accelerometer, 4G up, etc.
- Non-phone devices – iPad, iPod Touch, tablets in general

No matter what you consider to be a mobile device, there are some points that you need to have in mind while developing a web site or a web app for these devices:

- Size of the screen
- What content the user will really need when accessing the data from a mobile device
- You need to optimize your HTML, CSS, and JavaScript code
- You will face a very fragmented market: different OS, browsers, screen sizes, accessories and possibilities – basically a new universe with its own rule

That last point is quite interesting as you might think about having a separate domain just for the mobile world. Well, this is one of the solutions we will be talking about during this course.

But, for sure, there are certain things you need to avoid while writing the HTML for your app:

- Tables – especially tables for layout
- Pop-up windows – this can make the site unusable as some mobile browsers do not support and some open them in unexpected ways
- Graphics for layout
- Frames and image maps (iframe is ok)

In terms of browsers in the mobile world, it is always important to know where we came from to imagine where we might be going to. So, here is short history of mobile browsers.

WAP (Wireless Application Protocol) started with version 1.0 that was never used and version 1.1 was released in 1998. This version brought WML as the standard and sites were called WAP sites – an example of a WAP site is the Yahoo WAP site. The version 2.0 was released in 2002 and introduced the XHTML-MP which put WML in a deprecated status.

The “DotMobi” era (.mobi) was approved by ICANN in 2005 and released to the public only in 2006.

Due to some limitations of WAP 2.0, a new tentative called ODP (On-Device Portals) was implemented and tested by Yahoo. But it was discontinued in 2010 because a different ODP site needed to be created for every different platform.

Currently mobile devices come with pre-installed browsers such as: NetFront, Myriad, IE Mobile, Safari, Nokia, Sony Ericson, Obigo, Motorola Mobile, Symbian, Android, WebOS, Blackberry, Samsung WebKit, etc. Users can also install other browsers such as: Opera Mobile, Opera Mini, Firefox for Mobile, SkyFire, Bolt, Chromium, etc. Remember that although you see the same laptop/desktop icons for browsers when you install them in mobile devices, they are not the same!

Regarding markup language, in 1996 came HDML (Handheld Markup Language) that was very similar to HTML. It was never released but gave way to WML. The WML was never standardized by W3C but by the WAP Forum (Open Mobile Alliance = OMA) – the language was formed by a "deck" of cards tags and each card tag looks like the body tag of HTML creating the possibility of having more than one web page in the same document (great feature to speed the rendering of sites in early 2000's). This markup language did not support GIF, JPEG, PNG images.

After WML, the OMA released a new standard called XHTML-MP. This came as a subset of XHTML and the validator can be found at W3 Mobile Validator. It does not support: nested tables, thead, tbody, rowspan, colspan, input type="image", input type="files", image maps, frames, iframes, font, dir, menu, strike, u, center. If you open our workfiles, you will find the **pmwtemplate.html** that was written using XHTML-MP – observe the DOCTYPE tag.

CSS came way later – for those of you already involved in web development, you might remember that XHTML Basic 1.0 did not support CSS and, although version 1.1 added support to it, it was just for embedded and external CSS, not the inline (using the style attribute). WCSS, also called WAP CSS came as a subset of CSS 2.0. Open in your text/web editor the file **pmwmarquee.html**. The properties display:none; in .offer are for browsers that do not understand the marquee element.

The first standard adopted by W3C was XHTML Basic.

The guide from Google, for Mobile Web Development has made it clear that users expect responsive and visually engaging websites no matter the devices they are using. A web application (web app) should have the layout and styling that is responsive to the current display with intuitive functionality. When building a web app, you will be asked to use HTML, CSS, and JavaScript and the responsive layout and style should include:

- DOM elements that can be accessed and manipulated using basically JavaScript without too much overhead of libraries or frameworks (for example: jQuery)
- Appropriate document type declaration – the DOCTYPE tag!
- A responsive grid-based layout using CSS
- Media queries that will provide fluid breakpoints for different screen sizes
- The multimedia tags that will be used to display video or play audio – the <video> and <audio> tags
- Images that are responsive and will adjust for the dimensions and resolution of any device
- Touch and mouse events that contain large hit targets regardless of the platform the web app is being used on

Now, I am pretty sure you are ready to start navigating in this great and turbulent world of options available for mobile devices!