MOBA1

MOBILE WEB:

BASICS

OVERVIEW

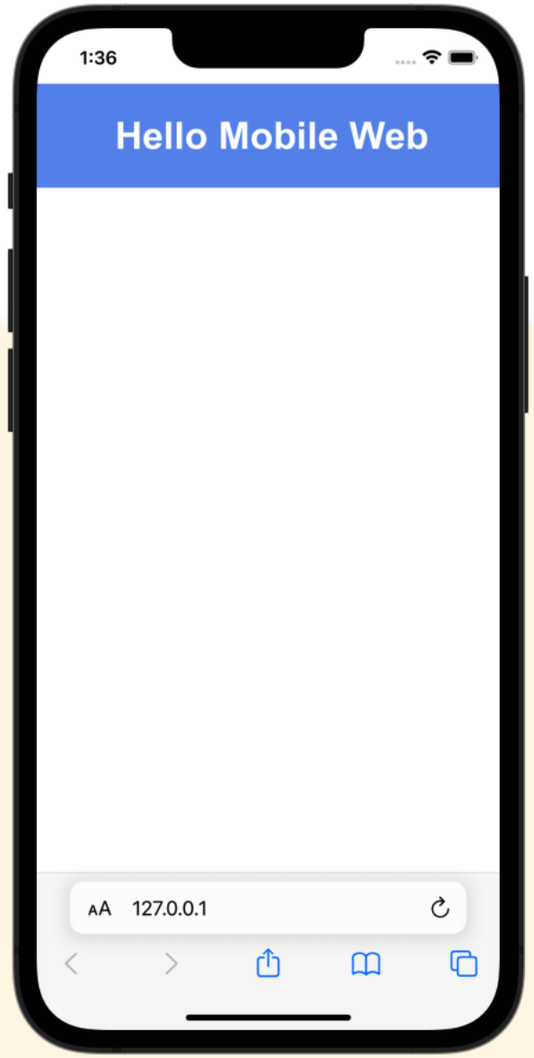
- Websites and Web Apps on Mobile Devices
- Browsers and Rendering Engines
- Mobile-Friendly Web Content
- Forms on Mobile Devices
- Mobile Web Device APIs (Part 1)

OVERVIEW

- Websites and Web Apps on Mobile Devices
- Browsers and Rendering Engines
- Mobile-Friendly Web Content
- Forms on Mobile Devices
- Mobile Web Device APIs (Part 1)

MOBILE WEB

```html
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport"
        content="width=device-width, initial-scale=1.0">
  <title>Mobile Web Demo</title>
</head>
<body>
  <header>
    <h1>Hello Mobile Web</h1>
  </header>
</body>
</html>
```



MOBILE WEB

- Every smartphone has a web browser in it
- Mobile web usage exceeds desktop web usage
- The Web is the only true cross-platform technology

Mobile web has its own challenges, but there is no other technology that allows you to create content and apps that reach every platform.

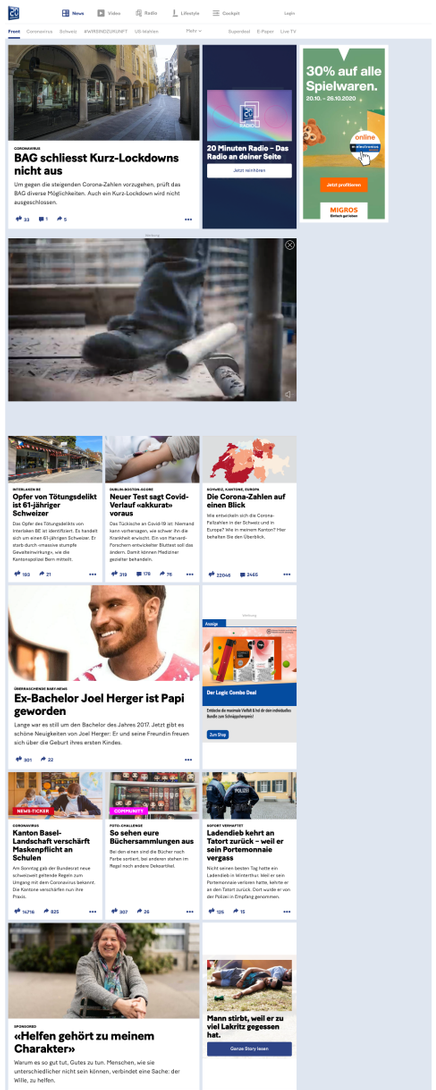
Since April 2015 mobile usability of a website is one of Google’s ranking factors. Starting July 1, 2019, mobile-first indexing is enabled by default for all new websites.

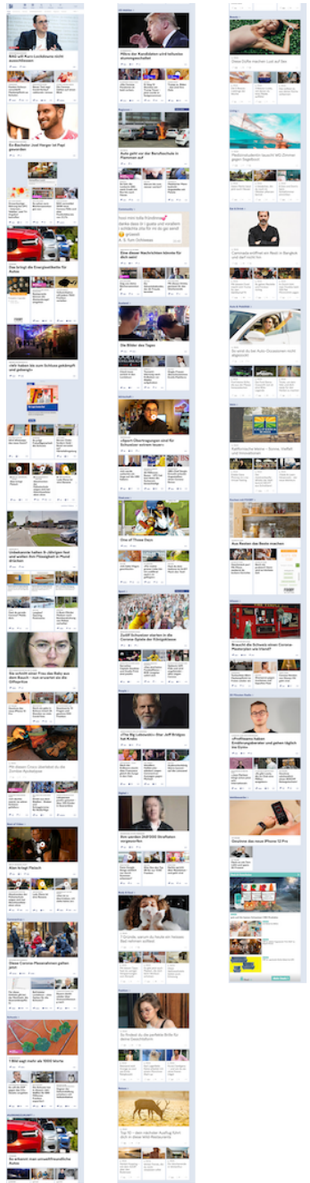
POSSIBLE PROBLEMS

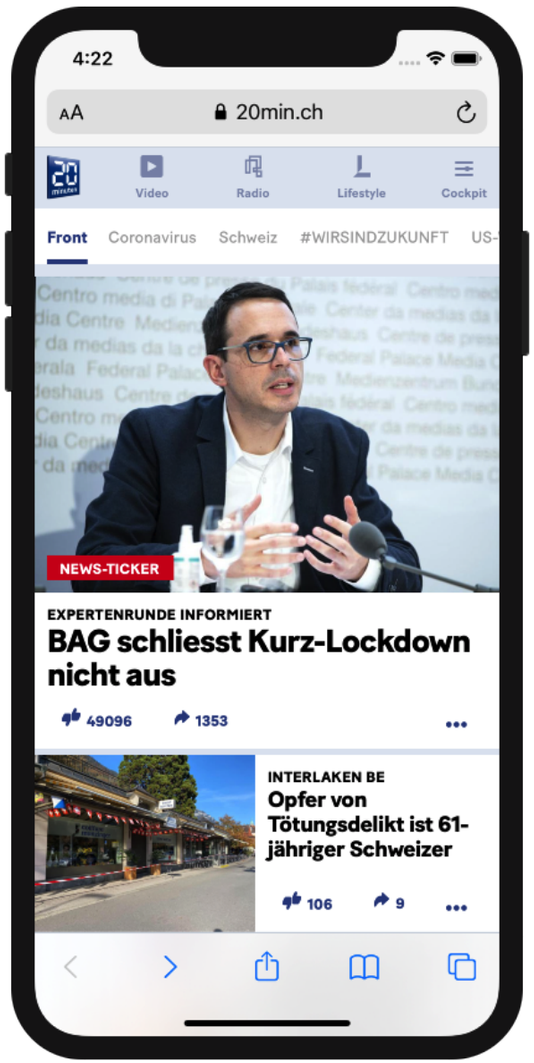
- Many different mobile web browsers
- Support for web technologies varies
- Mobile devices are smaller and slower
- Mobile interfaces require us to rethink our sites

POSSIBLE PROBLEMS

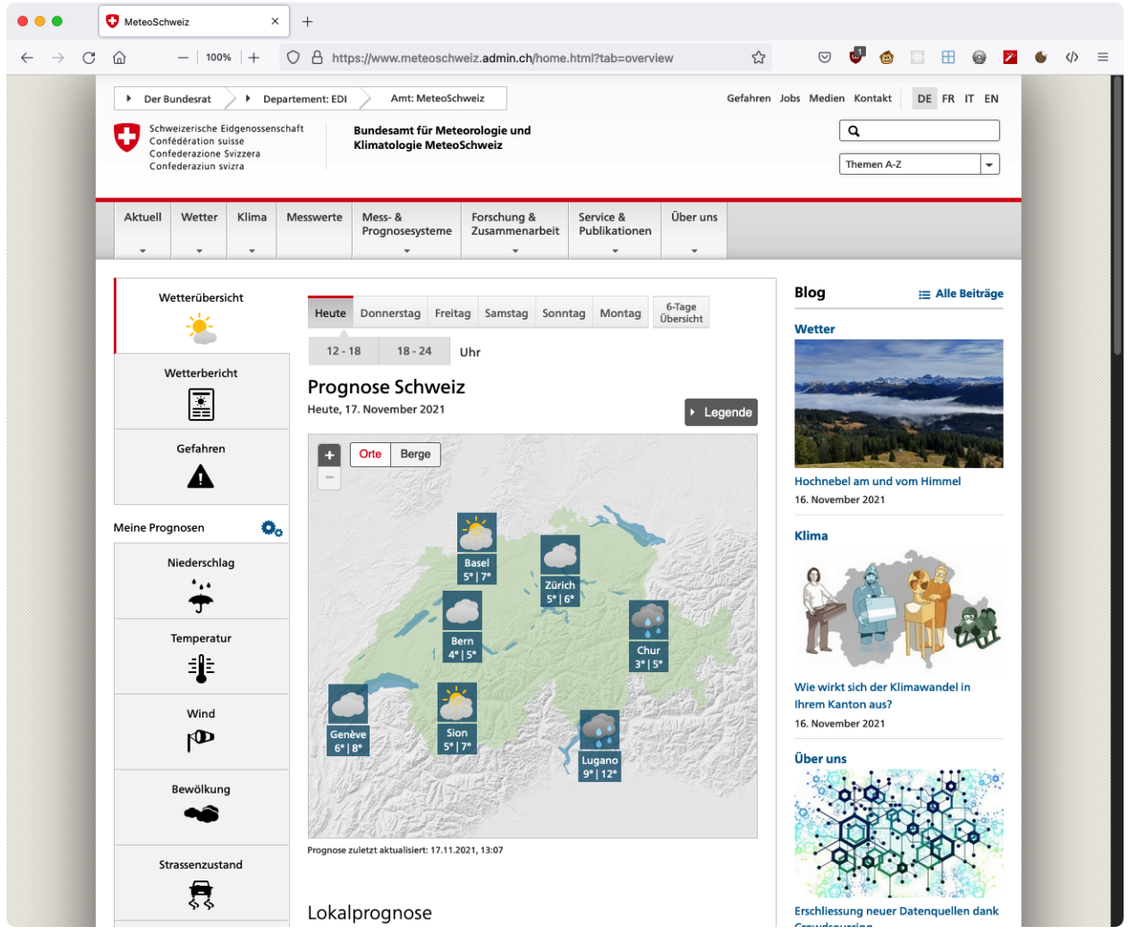
LESS INFORMATION (NEWS PORTAL)



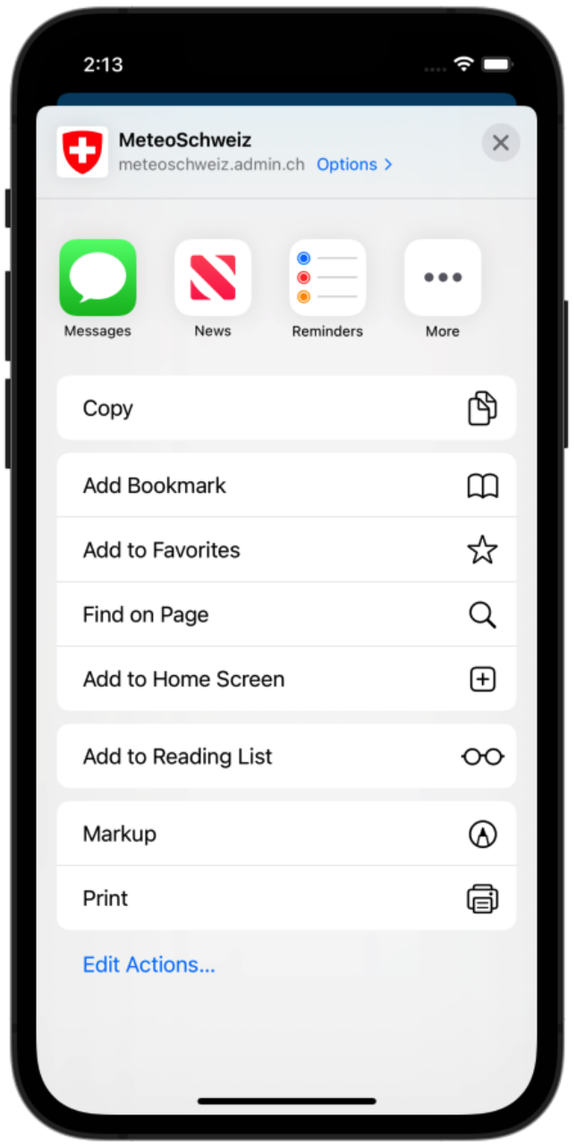




LESS INFORMATION



ADD TO HOME SCREEN



MOBILE APPS AND WEBSITES

- Apps
  - Native
  - Web-based
  - Hybrid
- Websites
  - Tailored to mobile browsers
  - Responsive

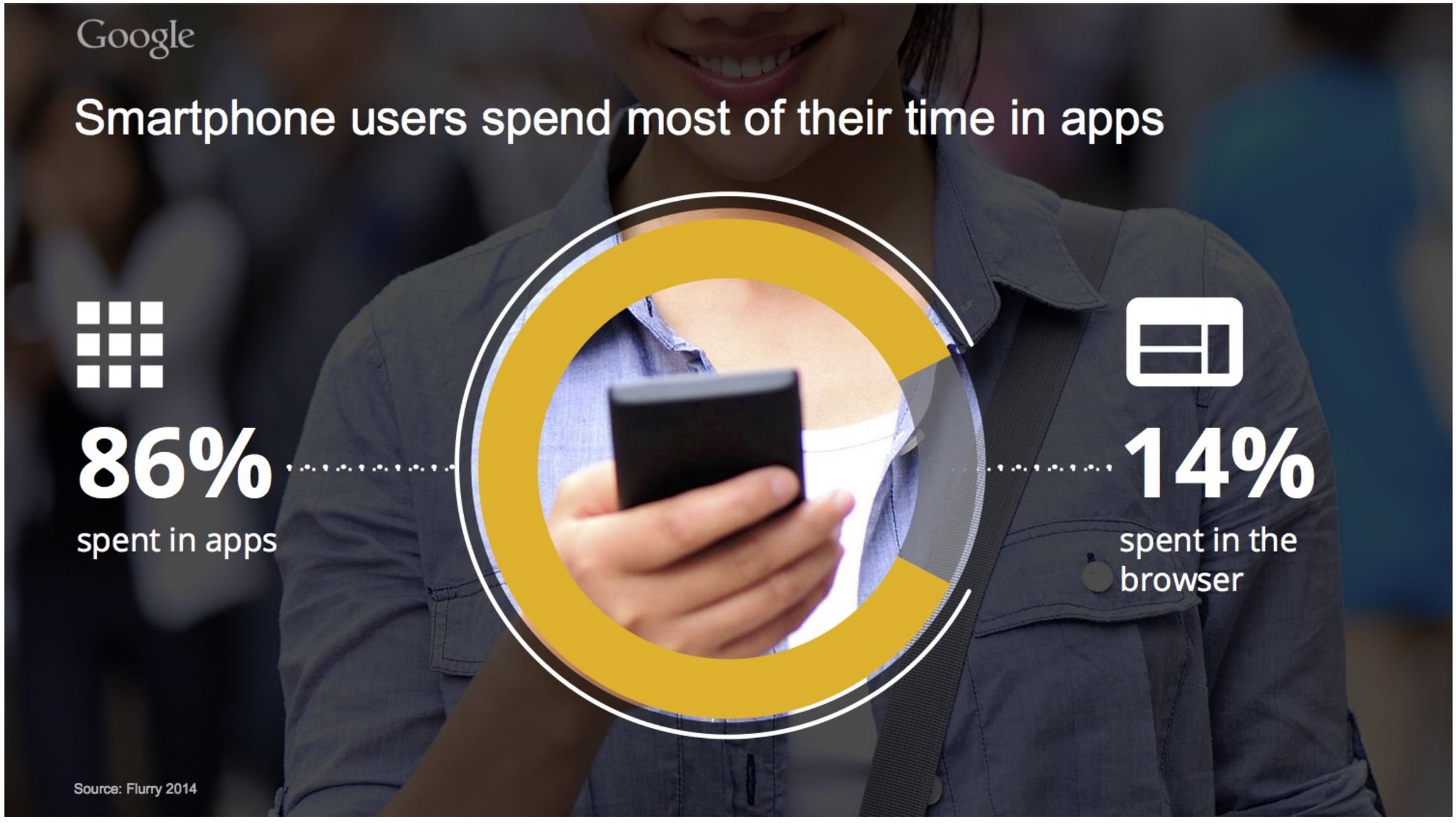
Fundamental change in use

PC internet

- Shared, or used at work and locked-down
- Semi-portable at best
- Web and web search

Mobile internet

- Personal
- Taken everywhere
- Web, web search, apps, social, location, service integration, prediction, APIs, image recognition, local wireless…



TIME SPENT IN APPS

That looks pretty bad for the Web

Until you realize that 60% of the time in native is spent in just four apps (social and gaming)

MOBILE WEB ADVANTAGES

- Easiest route into mobile development
- HTML, CSS and JavaScript
- Make content accessible on almost any platform
- Save development time and cost
- Lower maintenance costs
- Independence of app stores

MOBILE WEB CHALLENGES

- Platform integration of native apps cannot be reached
- More or less dependent on connectivity
- Lack of developer tools compared to native app development
- Monetization of mobile sites can prove tricky

If monetization is one of the key requirements, a hybrid or web app strategy could prove to be a good compromise.

HTML, CSS, JAVASCRIPT

- Progress in web standards
- Look-and-feel *close* to that of apps possible
- Single code base for a number of popular devices
- Access hardware such as the camera and microphone
- Local data storage for offline availability
- Problem: Do wee need to factor in technology fragmentation across mobile browsers?

OVERVIEW

- Websites and Web Apps on Mobile Devices
- Browsers and Rendering Engines
- Mobile-Friendly Web Content
- Forms on Mobile Devices
- Mobile Web Device APIs (Part 1)

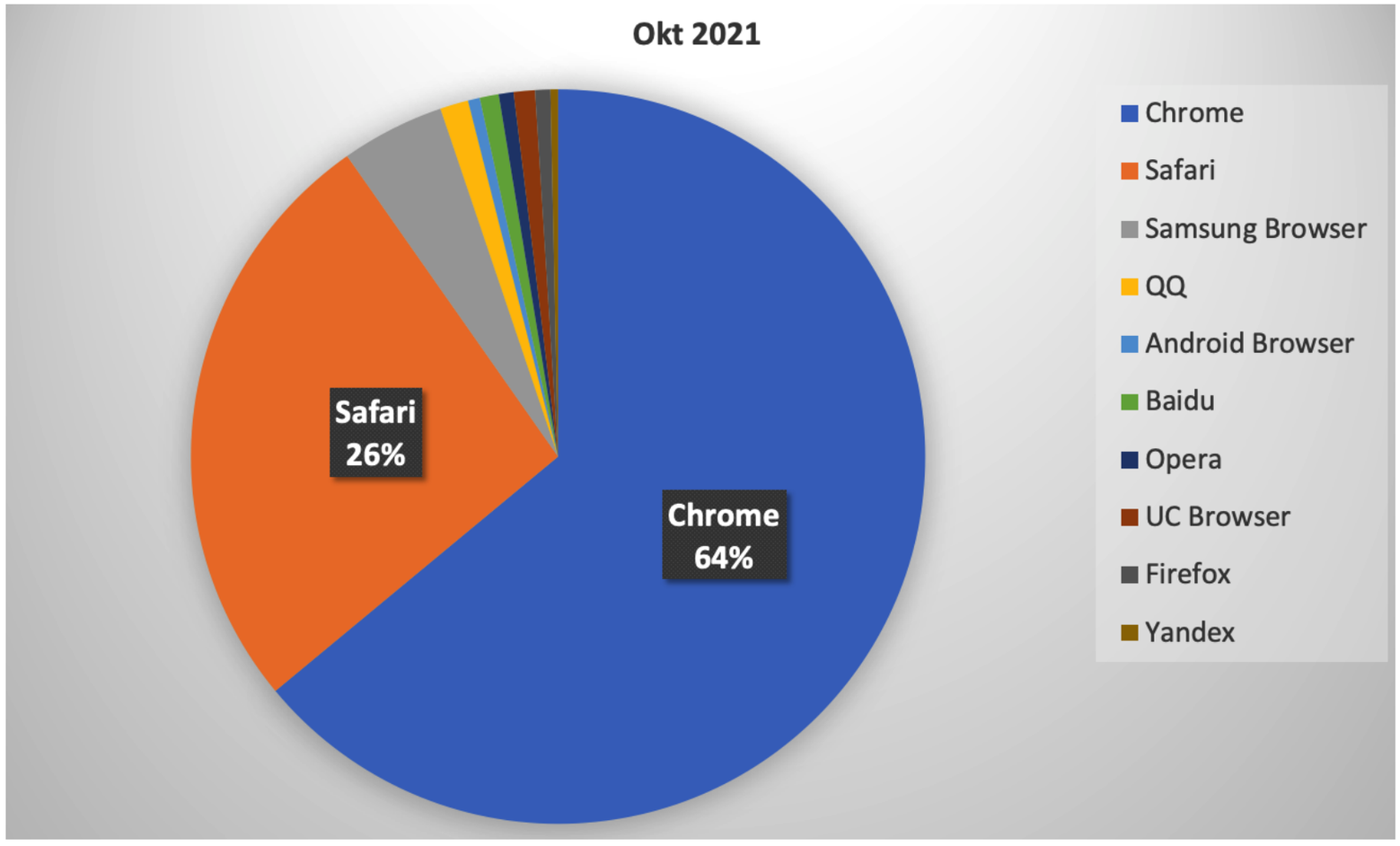
DOMINANT MOBILE RENDERING ENGINES

- WebKit (Safari)
- Blink (Chromium)

Support of Web specs

- html5test.com/results/mobile.html
- rng.io (Facebook)

MOBILE BROWSER MARKET SHARE



Source: http://www.netmarketshare.com

MOBILE BROWSER MARKET SHARE (OCT-21)

CHROME ON ANDROID

- Now the Android default browser
- Device vendor can use Google’s version
- Or they can build their own based on Chromium
- HTC Chromium, Samsung Chromium, ...

WEBVIEWS

- Embedded browser that native apps can use
- Android 4.3 and lower: AndroidWebKit
- Android 4.4: Chromium 30; fixed version
- Android 4.4.3: Chromium 33; fixed version
- Android 5: Chromium 42; updated with Chrome
- ...

GOOGLE SERVICES

- Package of crucial Google apps
- Maps, YouTube, Play ... and Chrome
- Device vendors may opt in to or out of the *entire* package
- But: Not mandatory to make Chrome the default browser
- Some vendors opted out: Amazon, Xiaomi, Huawei, ZTE, ...

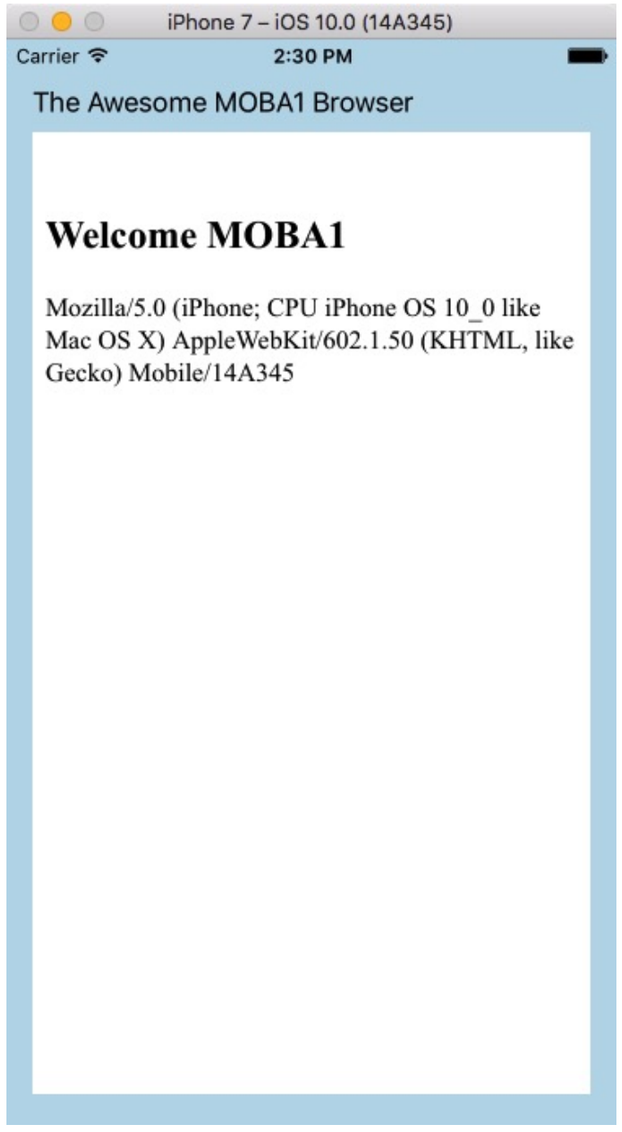
BROWSERS ON IOS

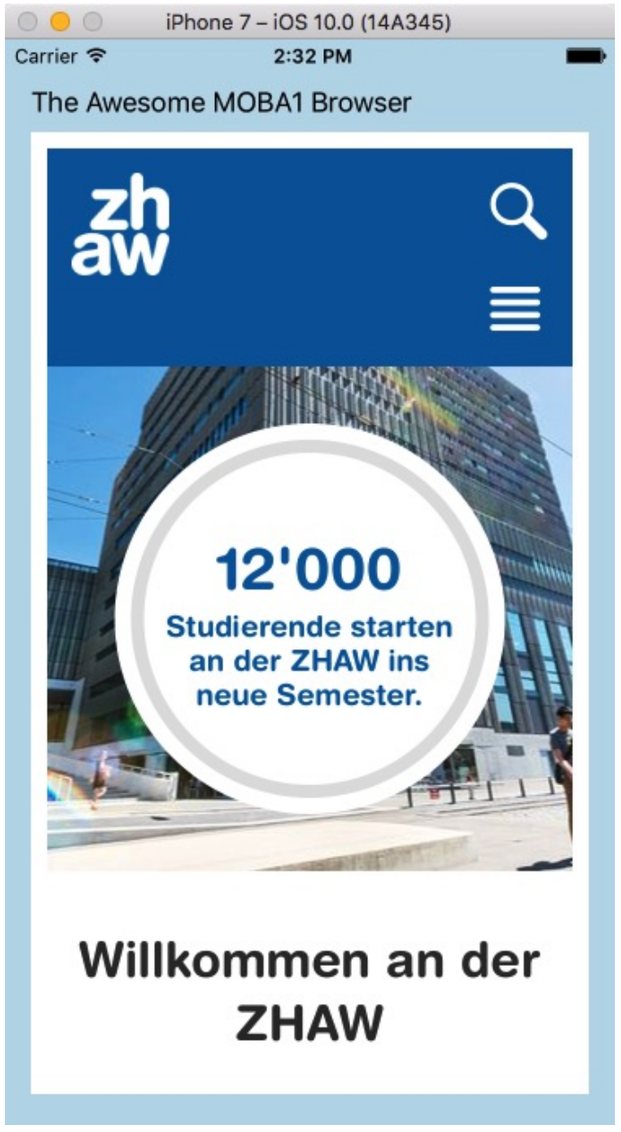
- Must use Apple’s rendering engine WebKit
- Chrome on iOS is not Chrome
- Testing with Chrome on iOS is in fact an Apple WebView test
- Same for Firefox and Edge

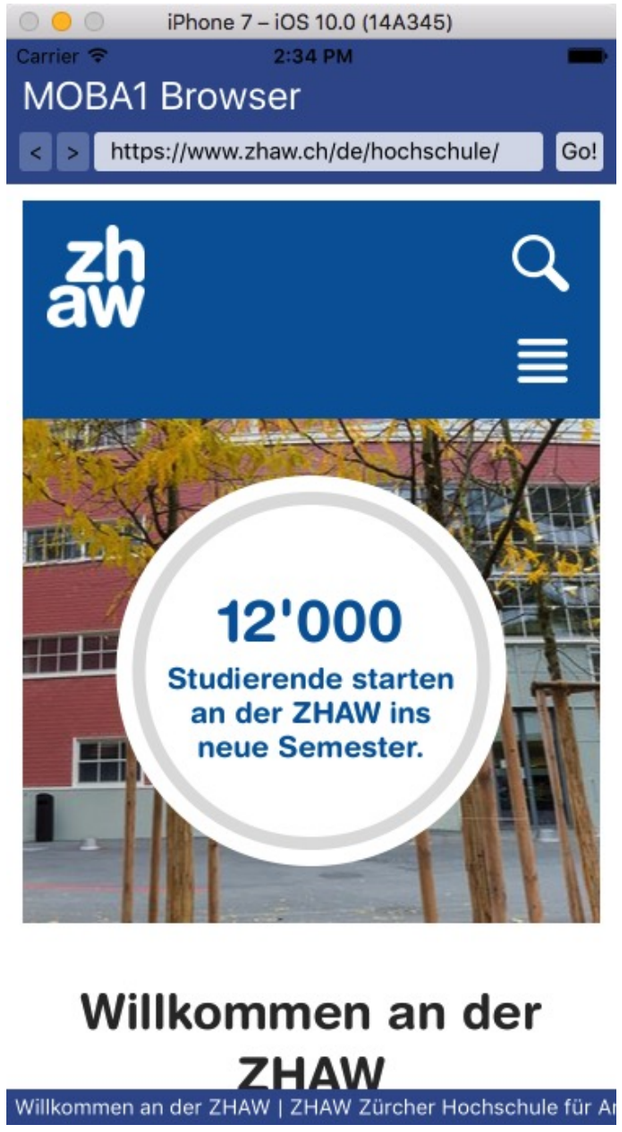
2017: Announcing Microsoft Edge for iOS and Android, Microsoft Launcher

2020: Change the default web browser or email app on iOS

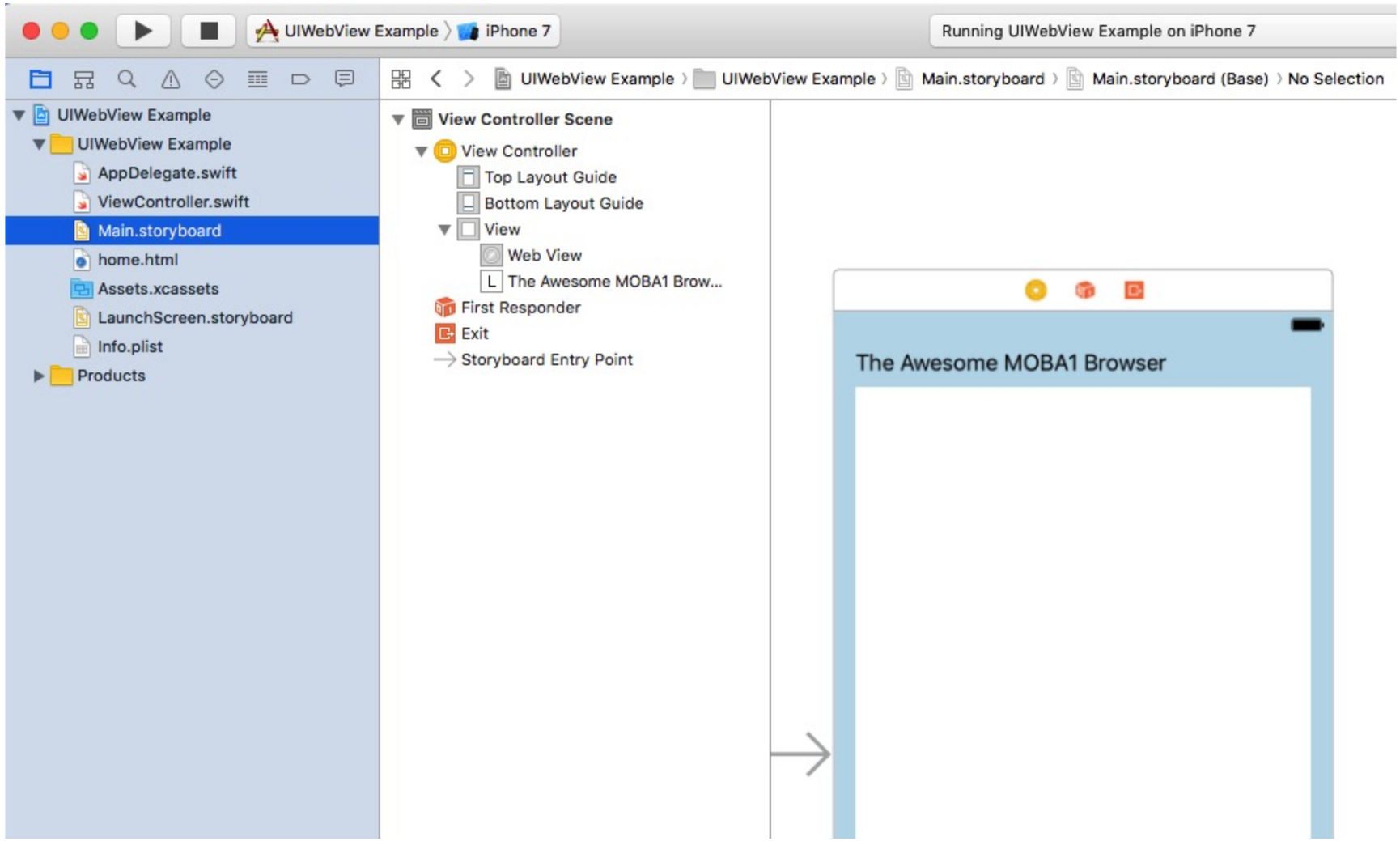
DEMO: WEBVIEW







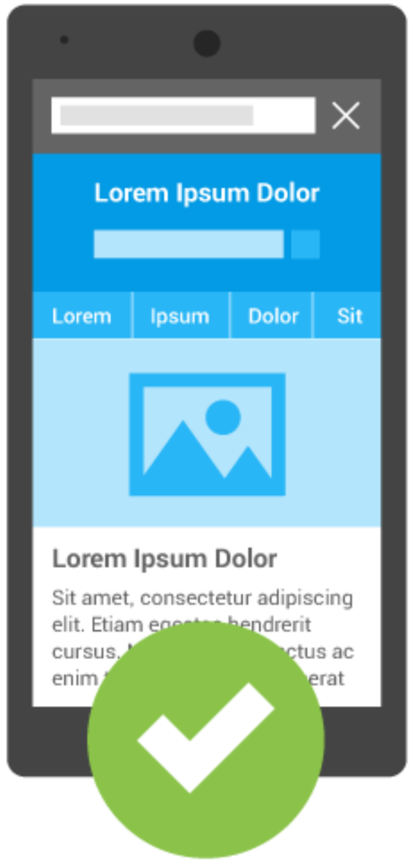
DEMO: WEBVIEW



OVERVIEW

- Websites and Web Apps on Mobile Devices
- Browsers and Rendering Engines
- Mobile-Friendly Web Content
- Forms on Mobile Devices
- Mobile Web Device APIs (Part 1)

MOBILE-FRIENDLY WEBSITES



COMMON MISTAKES

- Unplayable Content
- Interstitials
- Faulty Redirects
- Mobile-Only 404s
- Wrong viewport settings
- Small font size
- Touch elements too close

https://developers.google.com/search/mobile-sites/mobile-seo/common-mistakes

MULTI-DEVICE CONTENT

- People don’t read web pages, they scan
- On average, people only read 20–28% of web page content
- Write for mobile
  - Focus on the subject at hand and tell the story upfront
  - Get your main points across at the start (in around 70 words)
  - Use plain language, shorter words and simple sentence structures

THE NEXT BILLION USERS

- Most of these users coming online will have cheap devices
- Take low-cost phones with small viewports and data budget into account
- Users may not be reading in their first language

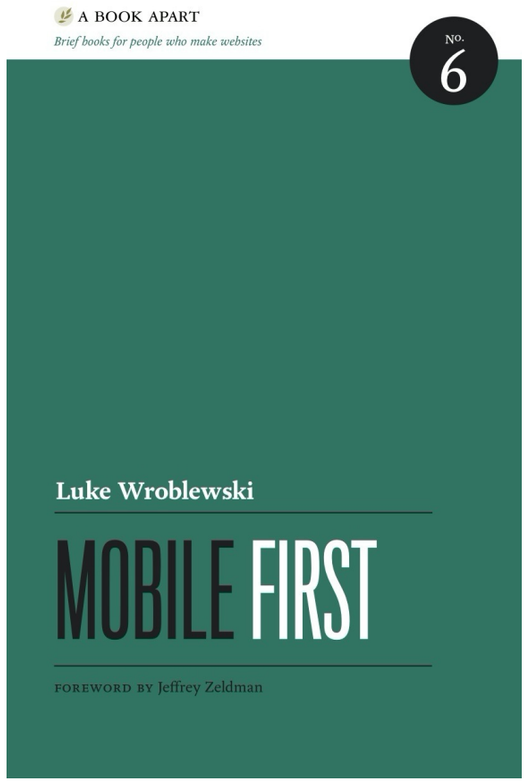
MOBILE-FRIENDLY CONTENT

- Eliminate unnecessary content
- Remove redundant images
- Consider different viewport sizes
- Design content for mobile
- Test content
- Understand data cost

Keep it simple – reduce clutter – get to the point

MOBILE FIRST

- Traditional Approach:
  - Start with desktop version
  - Then develop reduced mobile version
- Alternative:
  - Reduce to the essential from the beginning
  - Develop mobile first
  - Then bother with desktop version



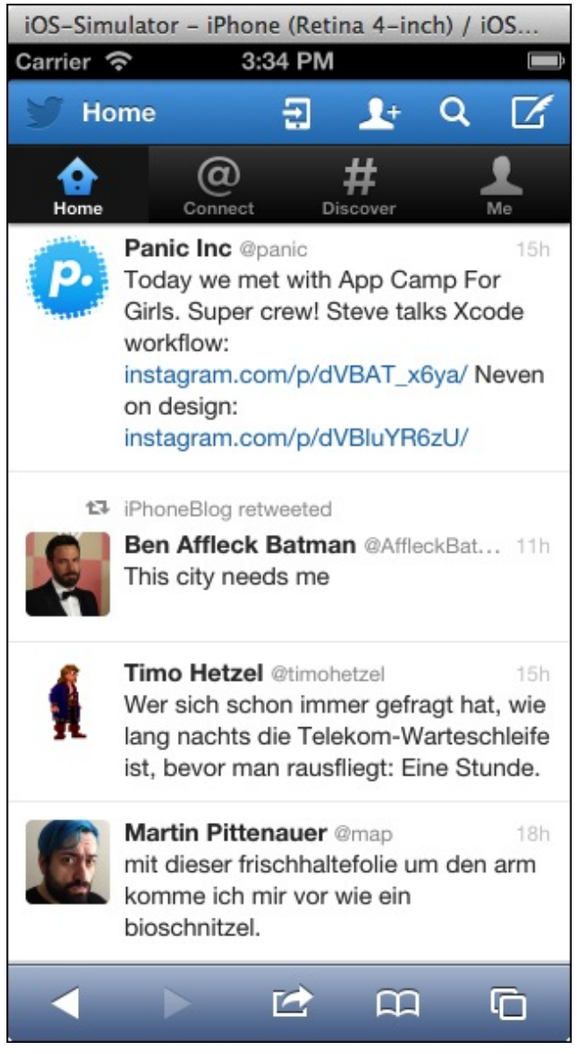
Luke Wroblewski: Mobile First

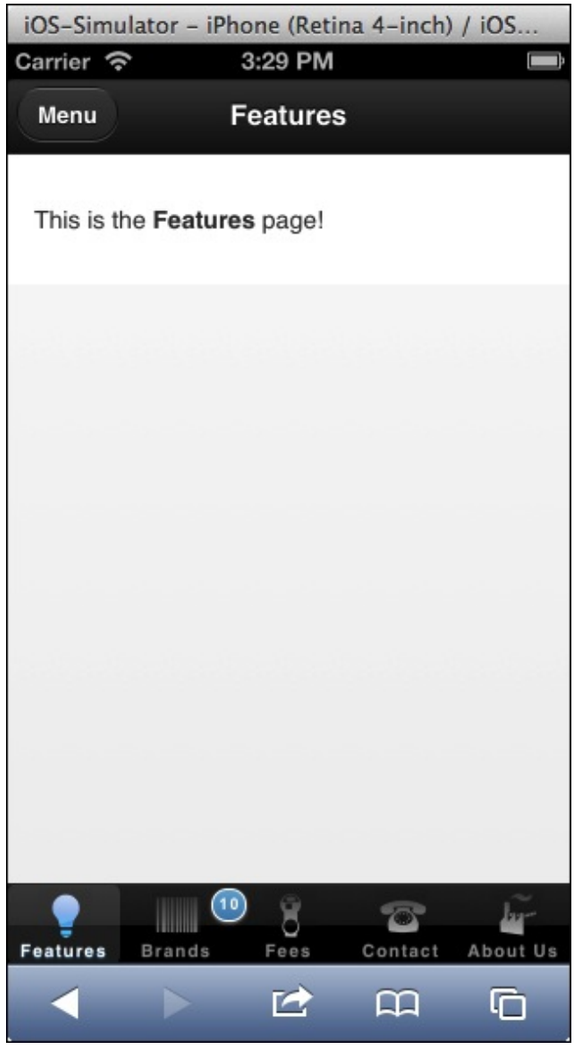
CONTENT FIRST

CONTENT FIRST

CONTENT FIRST

MOBILE WEB – NAVIGATION





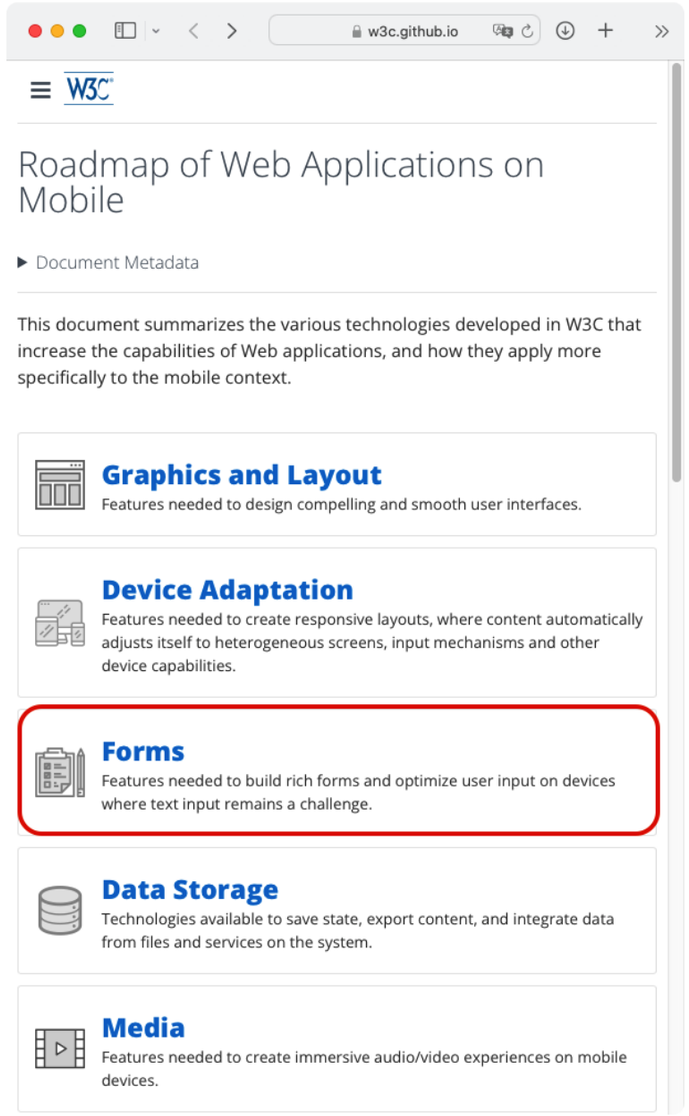
MOBILE WEB – TOUCH CONTROL

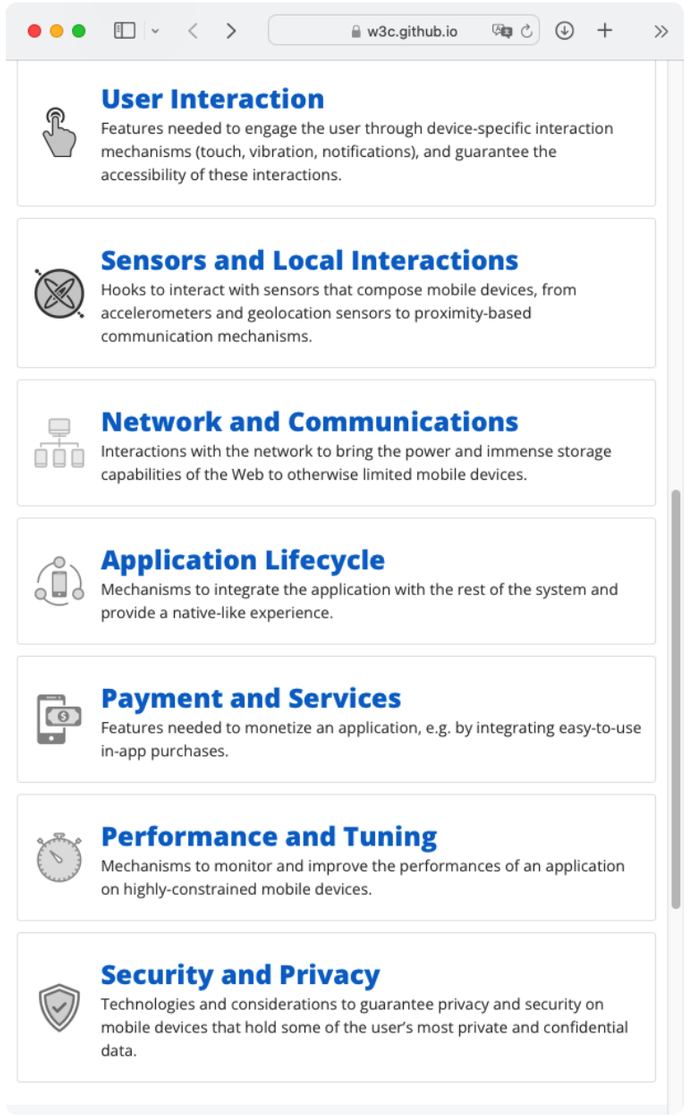
- Larger controls needed
- Enough space between controls
- Not all regions on the screen equally reachable

OVERVIEW

- Websites and Web Apps on Mobile Devices
- Browsers and Rendering Engines
- Mobile-Friendly Web Content
- Forms on Mobile Devices
- Mobile Web Device APIs (Part 1)

MOBILE WEB – ROADMAP





https://w3c.github.io/web-roadmaps/mobile/

MOBILE WEB – FORMS

- Little space for labels
- Often useful: `placeholder` attribute
- Important: appropriate `type` for the `input` element
- Date picker with `<input type="date">`
- Extensive testing on various devices necessary

MOBILE WEB – FORMS

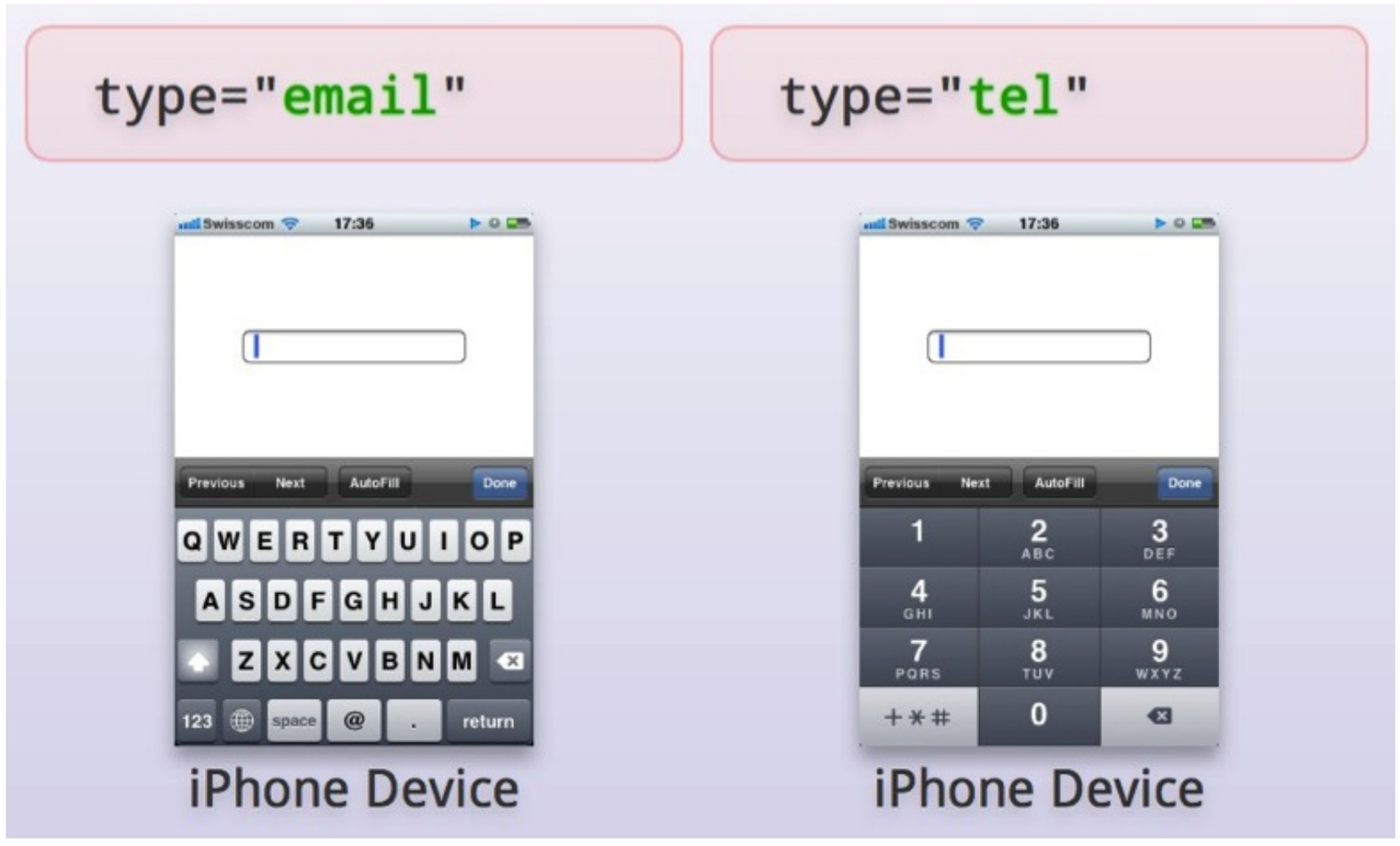
sbb.ch
2015 vs 2020

MOBILE WEB – FORMS

bahn.de
2015 vs 2020

MOBILE WEB – FORMS

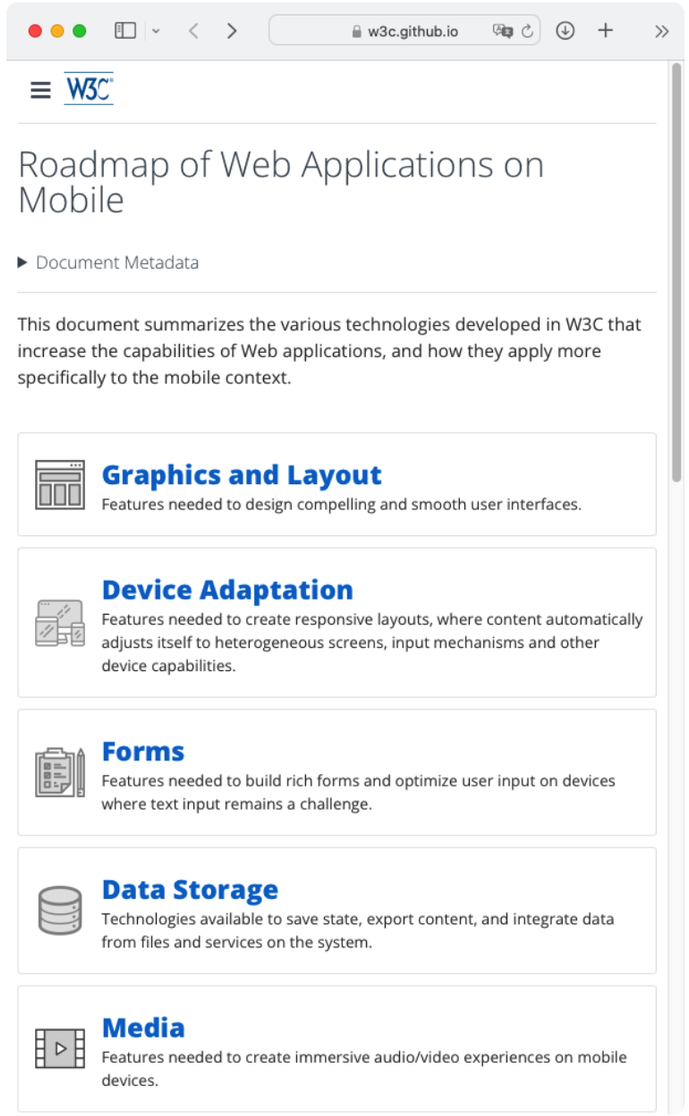
MOBILE WEB – FORMS

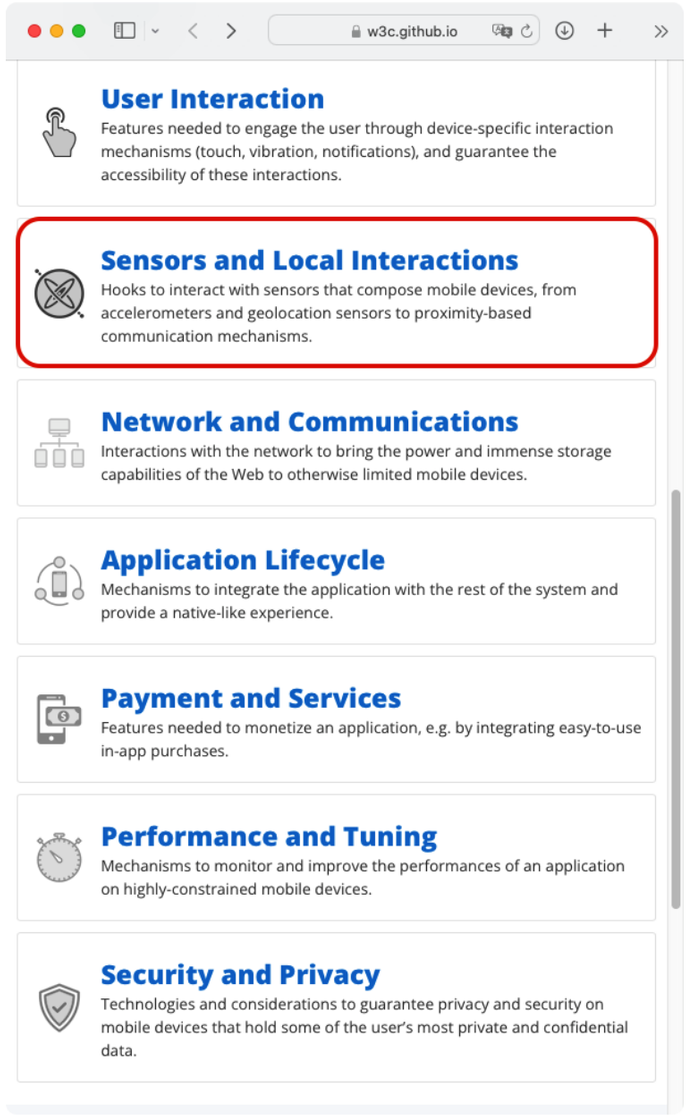


OVERVIEW

- Websites and Web Apps on Mobile Devices
- Browsers and Rendering Engines
- Mobile-Friendly Web Content
- Forms on Mobile Devices
- Mobile Web Device APIs (Part 1)

MOBILE WEB – ROADMAP





https://w3c.github.io/web-roadmaps/mobile/

URLS BEYOND THE WEB

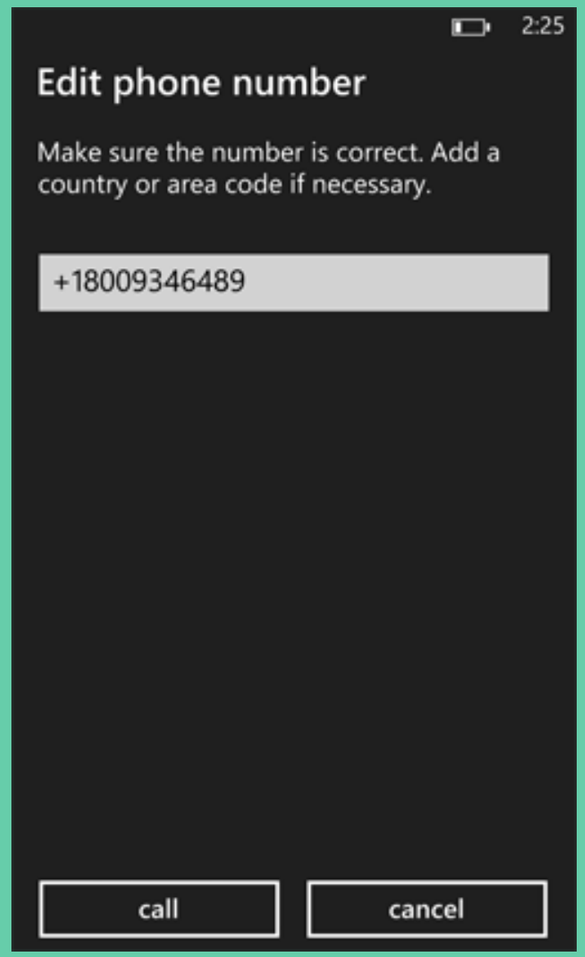
- Phone call and text message links
- Deep Linking into Apps

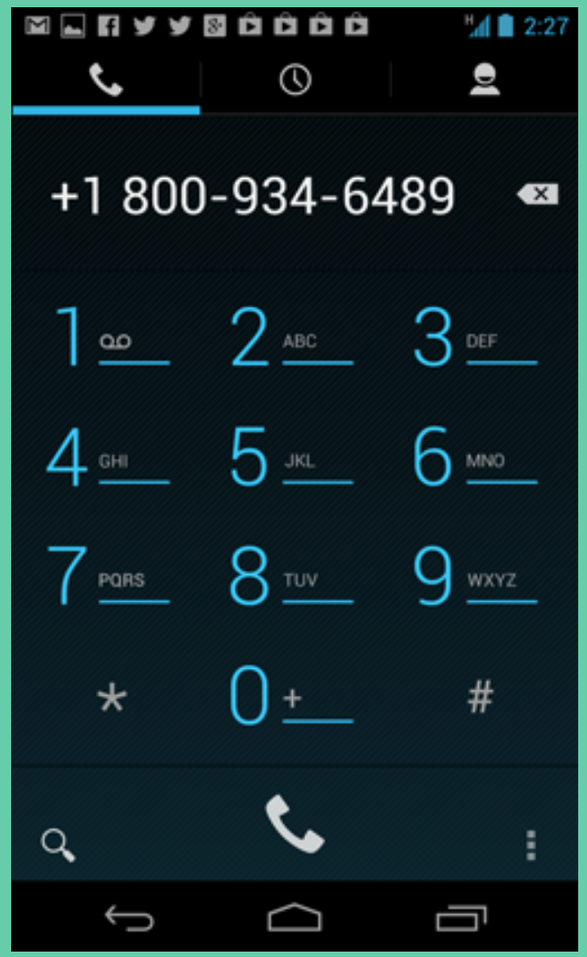
CLICK TO CALL

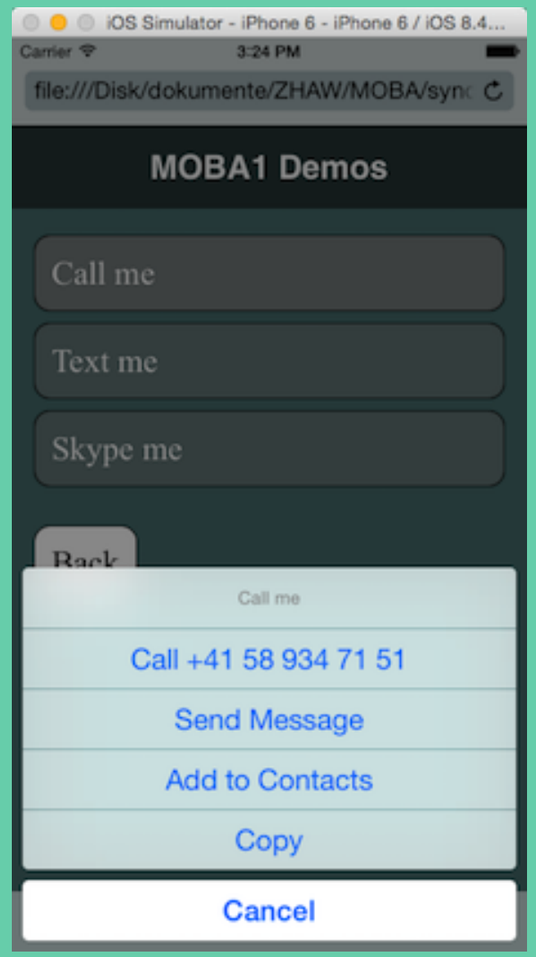
```
Order Pizza Now:
<a href="tel:+41-58-9347151">+41-58-9347151</a>

<a href="sms:+41589347151?body=hello%20there">Text me!</a>
```

- Directly connect by simply tapping a phone number
- Wrap all phone numbers in hyperlinks with the tel: schema
- Always use the international dialing format
- User receives a confirmation before the number is dialed

CLICK TO CALL







CLICK TO CALL

URI scheme rather than a Web API

- The tel URI for Telephone Numbers: RFC3966
- URI Scheme for [...] Short Message Service (SMS): RFC5724
- Platform or app specific schemes, e.g. Skype, FaceTime

Wikipedia: Uniform Resource Identifier
IANA: Uniform Resource Identifier (URI) Schemes

FORMAT DETECTION

- Mobile Safari automatically detects certain formats in texts
- Phone numbers for example are converted to links
- Chrome on Android also detects phone numbers
- To prevent Mobile Safari from automatically creating links:

```
<meta name="format-detection" content="telephone=no">
<meta name="format-detection" content="date=no">
<meta name="format-detection" content="address=no">
<meta name="format-detection" content="email=no">
```

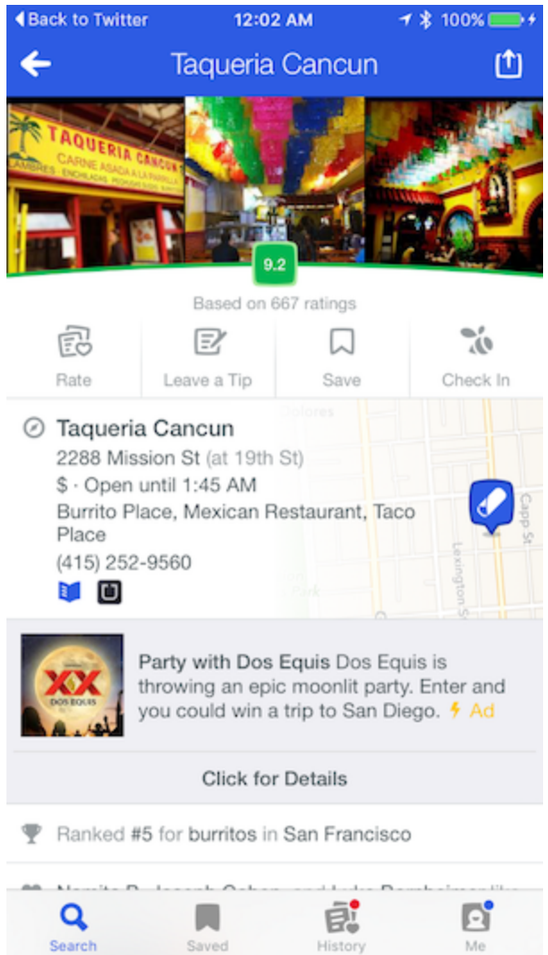
DEEP LINKING

- Usually it’s a web browser that responds to URLs
- Native apps can, too

iOS: Support Universal Links

Android: Deep linking and Android App Links

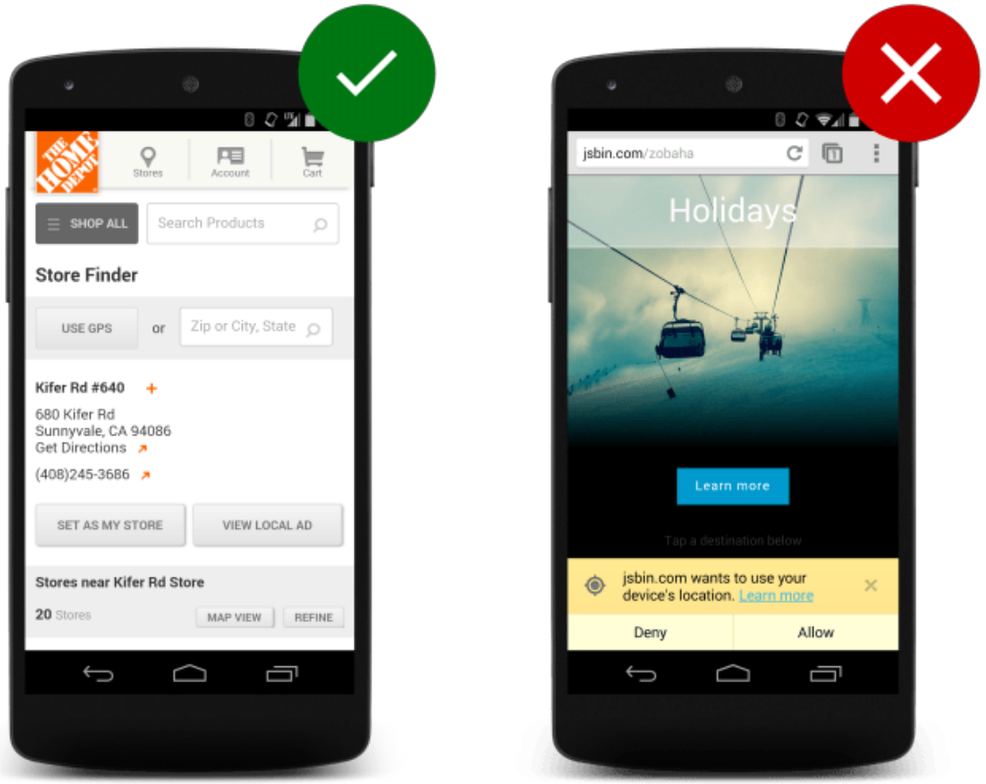
HOW DEEP LINKING WORKS IN IOS



- Clicked on a link in the Twitter app
- Foursquare app opens with linked page
- Or: web browser if app isn’t there
- Link back to the Twitter app

Apps more and more as part of the Web?

GEOLOCATION API



- Discover, with the user's consent, the user's location
- This creates a lot of interesting use cases
- Always request access to location on a user gesture

Users are distrustful of sites that prompt the user to give away their position on page load: Be clear and explicit about your need for the location.

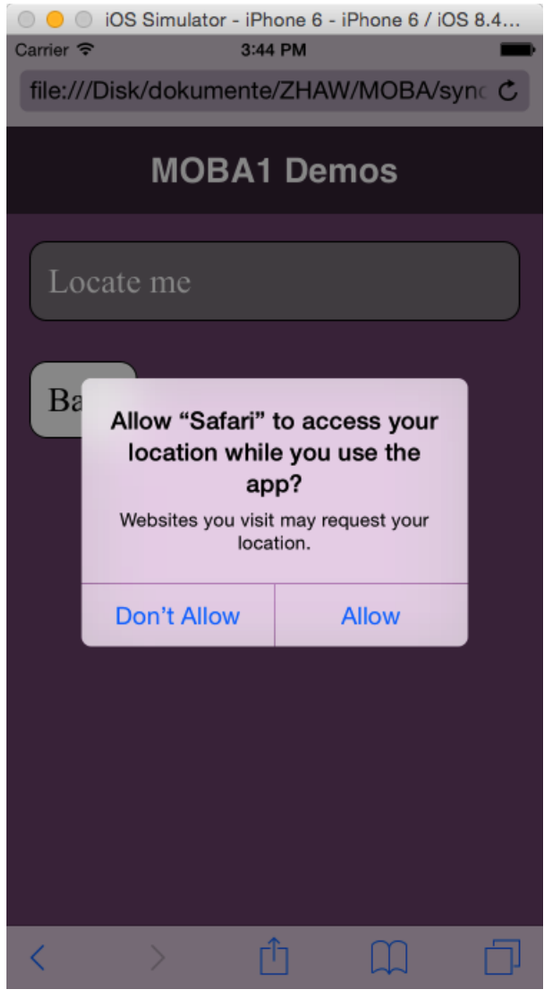
GEOLOCATION API

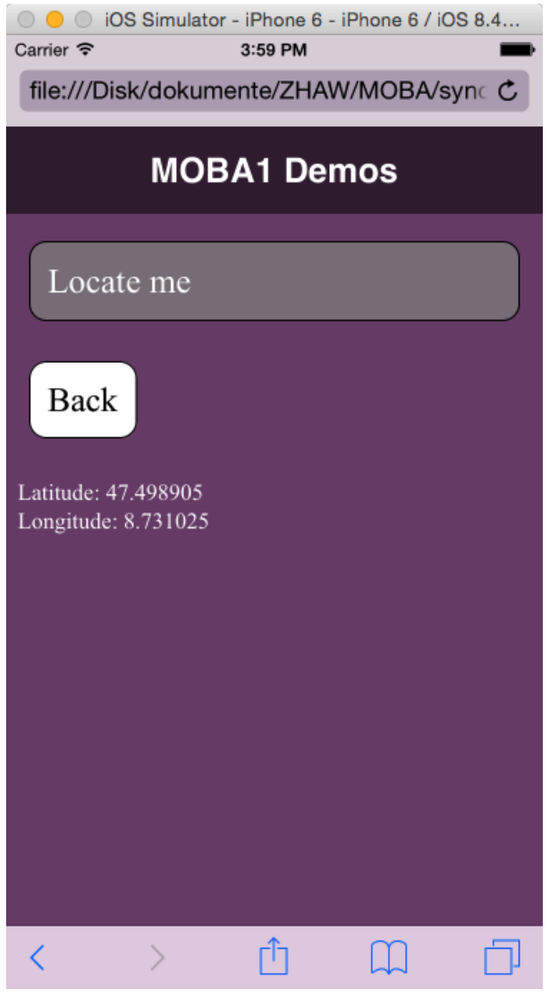
```
if (navigator.geolocation) {
  navigator.geolocation.getCurrentPosition(success, fail)
}

function success(position) {
  alert('Latitude: '+ position.coords.latitude +
  ', Longitude: '+ position.coords.longitude)
}
```

https://www.w3.org/TR/geolocation-API/

Depending on the location device your browser is using, the position object might actually contain a lot more than just latitude and longitude.

GEOLOCATION API: ACCESS PERMISSION





WATCHING THE USER'S LOCATION

- Use the Geolocation API method `watchPosition()`
- Fires as a more accurate lock on the user is available
- Fires when the user's position is changing
- Beware: this is not a free operation
- Call `clearWatch()` when finished

```
var watchId = navigator.geolocation.watchPosition(function(position) {
  ...
})
```

Now Obsolete: Geofencing API

GEOLOCATION API: OPTIONS

```
var geoOptions = {
  /* reduce the need to start geolocation hardware */
  maximumAge: 5 * 60 * 1000,

  /* don't keep the user waiting, set a timeout */
  timeout: 10 * 1000,

  /* default: coarse location, if fine-grained location is needed,
     this option can be used, but: it's slower to resolve and uses
     more battery */
  enableHighAccuracy: true,
}

navigator.geolocation.getCurrentPosition(geoSuccess, geoError, geoOptions)
```

GEOLOCATION

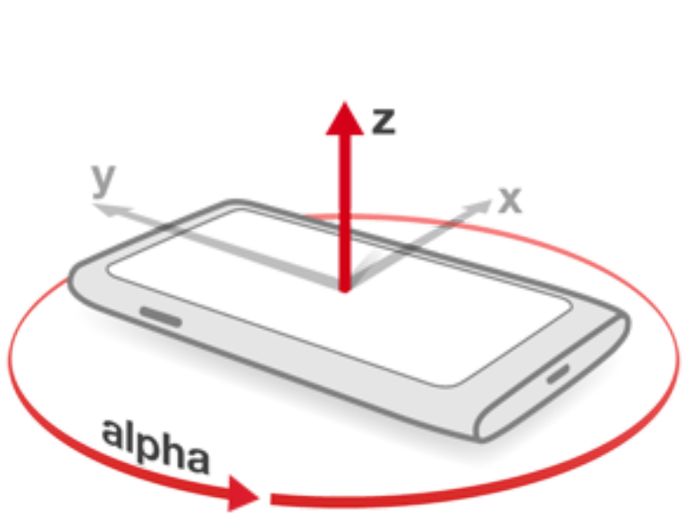
	ACCURACY	POSITIONING TIME	BATTERY LIFE
GPS	10m	2-10 minutes (only indoors)	5-6 hours on most phones
WiFi	50m (improves with density)	Almost instant (server connect and lookup)	No additional effect
Cell tower triangulation	100-1400m (based on density)	Almost instant (server connect and lookup)	Negligible
Single cell tower	500-2500m (based on density)	Almost instant (server connect and lookup)	Negligible
IP	Country: 99% City: 46% US, 53% International Zip: 0%	Almost instant (server connect and lookup)	Negligible

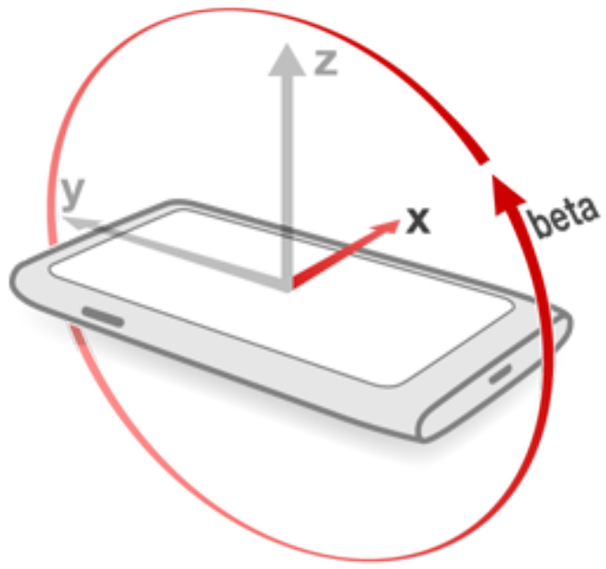
TABLE 3.1: An overview of the different ways a modern mobile device can detect your location. Smartphones make hybrid use of GPS, WiFi, and cell tower triangulation; laptops and desktops use WiFi, IP, and only rarely GPS.

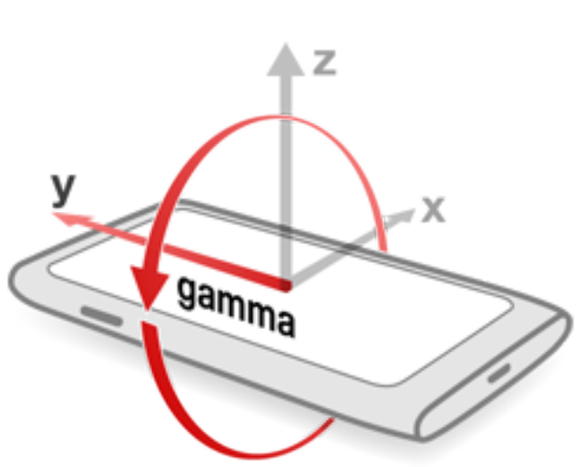
DEVICE ORIENTATION

Physical orientation and movement of the device

Gyroscope, Accelerometer, Compass







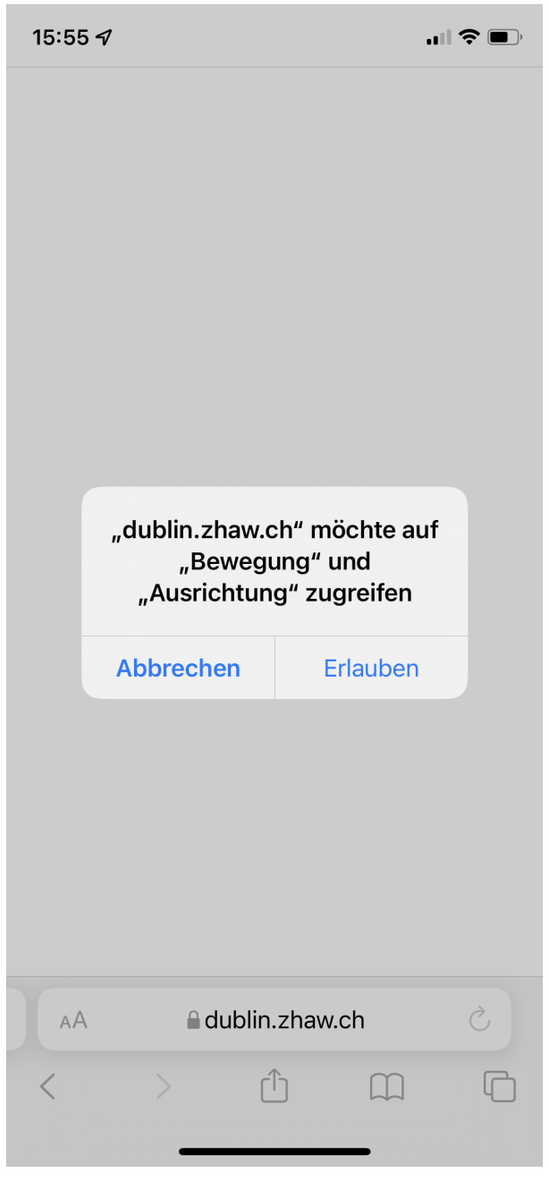
DEVICE ORIENTATION

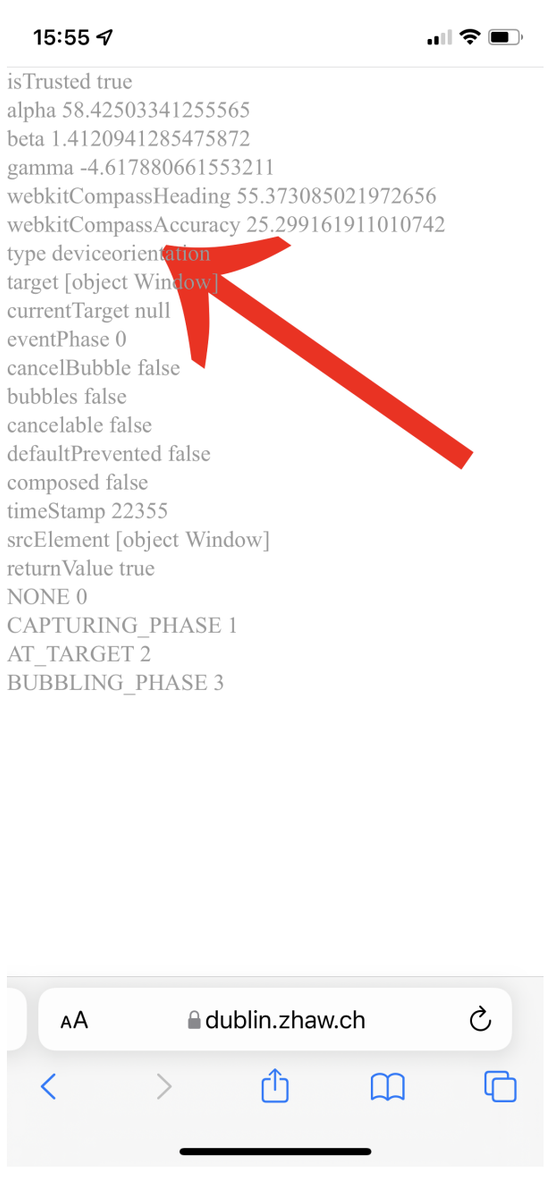
- Device orientation event returns rotation data
- Fires when the device moves or changes orientation
- Returns three properties: `alpha`, `beta`, and `gamma`
- On Mobile Safari also `webkitCompassHeading`
- Use sparingly and test for support
- Don’t update the UI on every orientation event
- Instead, sync to `requestAnimationFrame`

EVENT: DEVICEORIENTATION

```
if (window.DeviceOrientationEvent) {
  window.addEventListener('deviceorientation', function (e) {
    a = Math.floor(e.alpha)
    b = Math.floor(e.beta)
    c = Math.floor(e.gamma)
    el.style.transform
      = 'rotateZ('+a+'deg) rotateX('+b+'deg) rotateY('+c+'deg)'
    ...
  }, true)
}
```

EVENT: DEVICEORIENTATION





DEVICE MOTION

Event *devicemotion* fires when the device accelerates

- `acceleration` (excludes the effects of gravity)
- `accelerationIncludingGravity`
- `rotationRate`
- `interval`

DEVICE MOTION

Example:

What's the maximum acceleration of a person jumping?

```
if (evt.acceleration.x > jumpMax.x) {
  jumpMax.x = evt.acceleration.x;
}
if (evt.acceleration.y > jumpMax.y) {
  jumpMax.y = evt.acceleration.y;
}
if (evt.acceleration.z > jumpMax.z) {
  jumpMax.z = evt.acceleration.z;
}
```

https://developers.google.com/web/fundamentals/native-hardware/device-orientation

SPECIFICATIONS

- W3C Draft – DeviceOrientation Event Specification
- W3C Note – DeviceOrientation Event Specification
- Things got messy over time...
- Devices and Sensors Working Group
- Generic Sensor API

GENERIC SENSOR API

- Expose sensor data to the Web platform in a consistent way
- Blueprint for writing specifications of concrete sensors
- Abstract Sensor interface for different sensor types
- A number of sensor APIs are being built on top of this API: Accelerometer, Gyroscope, RelativeOrientationSensor, ...
- Currently on Chromium based browsers
- Polyfills: https://github.com/kenchris/sensor-polyfills

READING MATERIAL, SOURCES

READING MATERIAL

- Google Developers: Optimierung von Websites für Mobilgeräte
  https://developers.google.com/webmasters/mobile-sites/

SOURCES

- Slides and other material from courses WEB1, WBE
- Mobile Developer's Guide To The Galaxy, Open XChange https://www.open-xchange.com/resources/mobile-developers-guide-to-the-galaxy/
- Organizing Mobile (Luke Wroblewski) http://alistapart.com/article/organizing-mobile
- Mobile First (Luke Wroblewski, A Book Apart) http://mobile-first.abookapart.com
- The plural of Chromium is Chromia, Peter-Paul Koch, http://quirksmode.org/presentations/Spring2015/chromia_bt.pdf

SOURCES

- Google: Web Fundamentals
  https://developers.google.com/web/fundamentals
- Google: Be Mobile-Friendly
  https://developers.google.com/search/mobile-sites/get-started
- Building better with the mobile web, GDG Keynote, Paul Kinlan, Google,
  https://docs.google.com/presentation/d/1WLi-_iHgFitgxaQLSDoC5eVDABR_9yCWWgpMfZQM8Mw/edit#slide=id.g518e34c0
- Mobile is eating the world, Benedict Evans, 2013,
  http://de.slideshare.net/bge20/2013-11-mobile-eating-the-world