

# Dal sito web all'applicazione mobile

Paolo Milazzo

Dipartimento di Informatica

Università di Pisa

Master in Turismo e ICT

A.A. 2015/2016

# Responsive web design

- Nell'ambito dello sviluppo di siti web il termine «responsive» indica la capacità di un sito di adattarsi alle caratteristiche del dispositivo su cui viene visualizzato
- In particolare, il sito deve adattarsi alla dimensione dello schermo del dispositivo
  - Importante per i dispositivi mobili

[https://it.wikipedia.org/wiki/Design\\_responsivo](https://it.wikipedia.org/wiki/Design_responsivo)

[http://www.w3schools.com/html/html\\_responsive.asp](http://www.w3schools.com/html/html_responsive.asp)

# Modi per realizzare un sito web responsivo (1)

- Utilizzando PHP (o simili) per riconoscere il tipo di dispositivo
- L'informazione è nell'header del messaggio HTTP ricevuto dal server
  - Nel campo «user-agent»
- Se si riconosce un dispositivo mobile si reindirige la richiesta a un sito ad-hoc (ad es. m.repubblica.it, versione mobile di repubblica.it)

<http://stackoverflow.com/questions/13109770/best-way-to-redirect-single-php-page-for-mobile-devices-with-php-javascript>

# Modi per realizzare un sito web responsivo (1)

- Utilizzando le «media queries» CSS
  - Non richiede di utilizzare PHP (o simili)
  - Non richiede di sviluppare un sito separato dedicato ai dispositivi mobili
  - Consente di definire regole CSS da applicare solo nel caso di dispositivi mobili

[http://www.w3schools.com/css/css\\_rwd\\_mediaqueries.asp](http://www.w3schools.com/css/css_rwd_mediaqueries.asp)

# Uno strumento per provare...

- Emulatore di smartphone
  - Consente di vedere un sito come se fosse visualizzato su uno smartphone
  - Non è sempre accurato, ma rende l'idea...

<http://www.mobilephoneemulator.com/>

# Pacchetti CSS «pronti» per realizzare siti responsivi

- Esistono diverse soluzioni già pronte per realizzare siti responsivi
  - Consistono di documenti CSS + codice JavaScript da includere nel proprio documento HTML
  - Basta scrivere l'HTML seguendo le indicazioni e il sito sarà responsivo!

- Due esempi notevoli:

- W3.css

<http://www.w3schools.com/w3css/default.asp>

- Bootstrap

<http://getbootstrap.com/>

<http://getbootstrap.com/getting-started/#examples>

# Realizzare single page applications

- Esistono soluzioni per realizzare siti web consistono di più pagine all'interno di un unico documento HTML
  - «single page applications»
- L'utente scarica un unico documento HTML e può navigare tra diverse pagine
- Utile per realizzare applicazioni per dispositivi mobili (tramite web-view... vedremo)
  - In una app non vogliamo aspettare del tempo nel passaggio tra una pagina e l'altra
- Anche in questo caso esistono soluzioni JavaScript+CSS già pronte (es. JQuery mobile) da richiamare nel proprio documento HTML)

[http://www.w3schools.com/jquerymobile/jquerymobile\\_intro.asp](http://www.w3schools.com/jquerymobile/jquerymobile_intro.asp)

# Siti WordPress responsivi

- Anche un sito realizzato con WordPress può essere reso responsivo
- Basta usare un tema responsivo!
- Il tema
  - farà un uso opportuno delle «media queries» di CCS
  - oppure, sarà basato su pacchetti quali Bootstrap
- Esempi di temi WordPress basati su Bootstrap  
<http://wptavern.com/18-free-wordpress-themes-built-with-bootstrap>

# E per realizzare una vera e propria app?

Le applicazioni per il mondo mobile possono essere classificate in tre categorie:

- **app native** sono le app scritte e compilate per una specifica piattaforma utilizzando uno dei linguaggi di programmazione supportati dal particolare sistema operativo
  - Java (oppure C) per Android
  - Objective C per iOS
- **web app**, sono pagine Web ottimizzate per dispositivi mobili sfruttando le tecnologie Web, in particolare HTML, JavaScript e CSS
- **app ibride** sono le app che cercano di sfruttare il meglio delle due categorie precedenti: sono scritte con tecnologie Web ma vengono eseguite localmente all'interno di un'applicazione nativa

# App native, web e ibride...



# App native

- Devono essere scritte in un linguaggio di programmazione
- Sono le più efficienti, perché vengono eseguite senza intermediari
- Possono accedere (se autorizzate dall'utente) a tutte le funzionalità del dispositivo (GPS, rubrica, fotocamera, ecc....)
- Funzionano solo sul sistema operativo per cui sono state scritte (Android, iOS, Windows mobile)
- Vengono normalmente distribuite tramite un «app store». Ogni aggiornamento dei contenuti richiede l'aggiornamento dell'applicazione.

Esempio: per sviluppare applicazioni Android si può usare l'ambiente Android Studio

<http://developer.android.com/sdk/index.html>

# Web app

- Sono scritte usando i linguaggi per il web (sono in realtà siti web...)
- Ogni volta che si lancia l'app, il dispositivo apre un browser, si collega al sito web e lo visualizza a tutto schermo (l'utente non si accorge che c'è il browser in esecuzione)
- Sono le meno efficienti, perché vengono eseguite tramite un browser (anche se l'utente non se ne accorge)
- Non possono accedere a tutte le funzionalità del dispositivo (GPS, rubrica, fotocamera, ecc....)
- Funzionano su tutti i sistemi operativi
- Sono normalmente distribuite tramite un «app store». L'aggiornamento dei contenuti però è automatico (basta aggiornare il sito web)

Esempio: anche in questo caso si può usare l'ambiente Android Studio (ma senza bisogno di programmare)

<http://developer.android.com/guide/webapps/index.html>

<http://www.chimerarevo.com/android/sviluppare-una-webapp-su-android-parte-1-116919/>

# App ibride

- Combinano i vantaggi dei due approcci
- Sono scritte come un sito web inserito in un'applicazione nativa «contenitore» standard
- Sono mediamente efficienti
- Possono accedere (se autorizzate dall'utente) a tutte le funzionalità del dispositivo (GPS, rubrica, fotocamera, ecc....) tramite JavaScript (che interloquisce con il «contenitore»)
- Funzionano su tutti i sistemi operativi (ci sarà un contenitore diverso per ogni sistema operativo)
- Sono normalmente distribuite tramite un «app store». L'aggiornamento dei contenuti può essere automatico (usando un sito web esterno) o manuale (integrando il sito web nell'applicazione)

Una tra le piattaforme per app ibride più utilizzate è Apache Cordova:

<https://cordova.apache.org/>

<https://cordova.apache.org/docs/en/latest/guide/overview/index.html>

<http://www.html.it/pag/42121/introduzione-ad-apache-cordova/>