

Come geolocalizzare le nostre applicazioni con HTML5?

In questo articolo vedremo come utilizzare le API javascript per la geolocalizzazione messe a disposizione nativamente da HTML 5.

Grazie a queste API possiamo ricavare la posizione geografica del dispositivo ed implementare delle funzionalità molto interessanti.

Sebbene tutto questo funzioni anche sui browser dei nostri PC di casa, trovo che abbia pochissimo senso (bene che vada verrà restituita la posizione del provider), mentre per i dispositivi mobili dotati di GPS si aprono innumerevoli possibilità. Ad esempio, per la versione mobile di un sito, possiamo sostituire il classico “dove siamo” con un “portami a”, ovvero utilizzare le API Google Maps (delle quali ho scritto un [articolo introduttivo](#)) per tracciare il percorso a partire dalla posizione del dispositivo. Puoi verificare se la richiesta arriva da un dispositivo mobile utilizzando la [classe mobileDetect](#). Ma iniziamo.

Warm up

Crea un nuovo file geolocalizzazione.html ed intestalo con il doctype HTML5

Verificare che il browser supporti la geolocalizzazione

Per verificare che il browser supporti questa caratteristica, è sufficiente verificare l'esistenza dell'oggetto geolocation, in questo modo:

Come rilevare la posizione

Per recuperare i dati sulla posizione del dispositivo, utilizzeremo il metodo `getCurrentPosition()` che passa tre argomenti:

- Il nome della funzione da invocare in caso di successo
- Il nome della funzione da invocare in caso di errori (facoltativo)
- I parametri di configurazione (facoltativo)

L'utilizzo del metodo `getCurrentPosition()` farà scattare la richiesta di autorizzazione a rilevare la posizione.

Iniziamo a scrivere la funzione da eseguire in caso di successo che chiameremo `get_position()`

La funzione `get_position()`

A questo punto abbiamo le coordinate, proviamo a passarle a Google Maps tramite [lo script che ho presentato in questo articolo](#) (e che ovviamente non starò a spiegare)

Come prima cosa richiamiamo le API di Google Maps

In seguito inseriamo nel body un contenitore per la mappa

Ed infine diamo una dimensione a questo contenitore

Ed ora siamo pronti per implementare il metodo `getCurrentPosition()` che sostituiremo al alert che ci avvertiva che il browser supporta la geolocalizzazione.

Ecco come dovrebbe apparire la pagina `geolocation.html` ora, con la funzione `get_position()` completamente implementata.

A questo punto lo script funziona, lo puoi provare in questa [pagina di esempio](#).

Una volta autorizzata la geolocalizzazione, dopo alcuni secondi apparirà la mappa. Se stai facendo la richiesta da un dispositivo mobile, sulla mappa verrà indicata la tua esatta posizione.

Passiamo ora agli argomenti facoltativi di `getCurrentPosition()`

Gestire gli errori

Crea una nuova funzione e chiamala `get_error()`.

In questa funzione disponiamo di tre codici di errore principali (in forma numerica) e di una proprietà (`message`):

- Codice 1: Restituito quando l'utente non ha autorizzato la geolocalizzazione
- Codice 2: Quando non è stato possibile stabilire una posizione
- Codice 3: Quando il timeout (che è configurabile, vedremo più avanti) è scaduto

Negli altri casi viene valorizzata la proprietà `message` (che andrebbe utilizzata solo in fase di debug)

quindi la nostra funzione per gli errori potrebbe essere così:

Passiamo quindi il nome della nostra funzione come secondo argomento del metodo `getCurrentPosition()` in questo modo:

Parametri di configurazione

Vediamo ora i parametri di configurazione:

- `enableHighAccuracy`: da impostare su `true` per abilitare il massimo livello di precisione
- `timeout`: il tempo massimo della richiesta espresso in ms.
- `maximumAge`: il tempo trascorso il quale sarà necessaria una nuova localizzazione

Completiamo dunque gli argomenti della funzione `getCurrentPosition()` ad esempio in questo modo:

Se vuoi approfondire ulteriormente, dai uno sguardo alla [pagina delle specifiche ufficiali](#).

Conclusione

Come hai potuto vedere, rilevare la posizione di un dispositivo è poco più che un gioco da ragazzi. E pensare che fino a pochi anni fa era del tutto impensabile. Nel nostro esempio abbiamo semplicemente indicato la posizione su di una mappa. Ma come detto, le API di Google Maps ci permettono anche di tracciare un percorso. Oppure, pensa di dover realizzare il sito dell'azienda dei trasporti pubblici della tua città. Inserendo in un database le coordinate di tutte le fermate, potresti implementare la funzione "portami alla fermata più vicina". Possiamo veramente immaginare una lunga lista di servizi basati sulla geolocalizzazione in grado

di valorizzare i nostri lavori e le potenzialità dei dispositivi mobili.
Tu hai già utilizzato queste funzionalità? Lo farai?