

Verifica di Tecnologie : Google Instant

Si vuole simulare il funzionamento di **google instant** che, tramite chiamate Ajax, aggiorna in tempo reale i suggerimenti di ricerca sulla base del testo inserito dall'utente.

A tal fine si utilizza un database denominato **google** strutturato nelle seguenti tre tabelle:

utenti contiene semplicemente un elenco di utenti registrati (con username e password)

voci (**_id**, **voci**, **nClick**) Ogni record contiene:

- un **_id** numerico custom
- un elenco di stringhe correlate (voci) impostate sulla base delle ricerche già effettuate da altri utenti. Ad esempio:
["fossano", "castello", "acaja"]
["scuole", "superiori", "cuneo"]
- un campo numerico **nClick** che indica quante volte quelle voci sono state cliccate

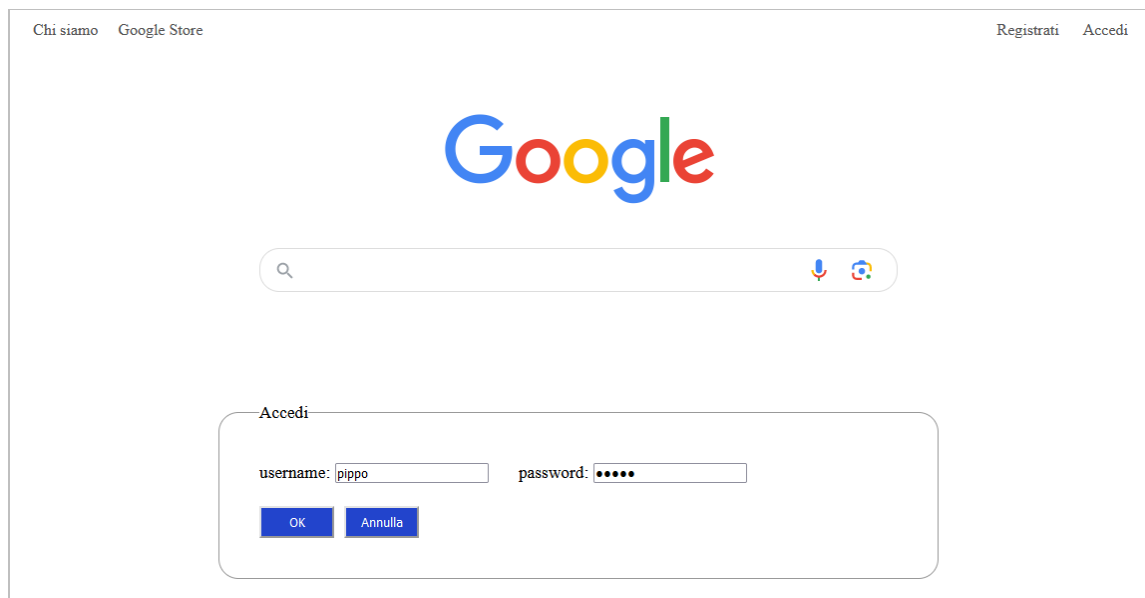
link (**codVoce**, **desc**, **link**) contiene un insieme di link riferiti alle voce precedenti.

Per ogni link è riportato il codice della voce a cui il link si riferisce

Parte 1 : Login

L'applicazione presenta all'avvio in alto a destra due pulsanti REGISTRATI e ACCEDI che consentono ad un utente registrato di accedere ai servizi google. In basso sono presenti due apposite sezioni **divAccedi** e **divRegistrazione** inizialmente nascoste.

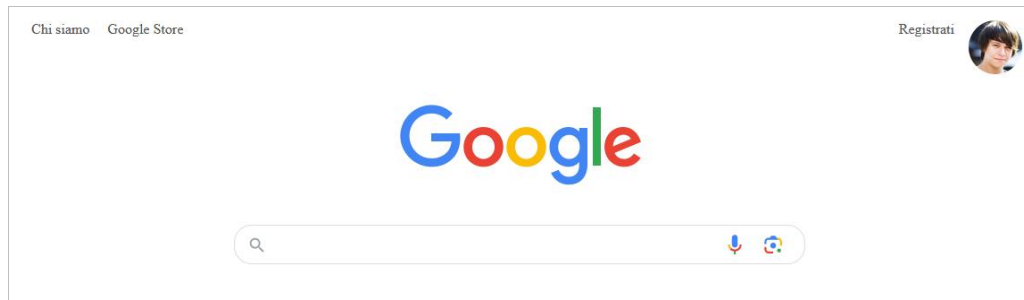
In corrispondenza del click sul pulsante **ACCEDI** viene visualizzata la sezione **divAccedi** in cui l'utente inserisce username e password.



In corrispondenza del click sul pulsante OK il client richiede al server di validare username e password

- In caso di NOK il server restituisce 401 **Unauthorized** ed il client visualizza una apposita alert
- In caso di OK il pulsante ACCEDI (in alto a destra) viene nascosto ed al suo posto viene visualizzata l'immagine di profilo dell'utente loggato. L'immagine ha lo stesso nome del campo username ed estensione .jpg. Al termine la sezione **divAccedi** viene nascosta.

Indirizzo **INFORMATICA** - Tecnologie



In corrispondenza del click sul pulsante **REGISTRATI** viene visualizzata la sezione **divRegistrali** in cui l'utente sceglie username, password e immagine di profilo.

In corrispondenza del click sul pulsante OK il client richiede al server di registrare il nuovo utente.

Il server controlla che lo username non si già esistente.

- In caso di username già esistente il server restituisce 422 **"username già esistente"** ed il client visualizza una apposita alert
- In caso di OK il server registra il nuovo utente, il client visualizza una apposita alert e chiude la sezione **divRegistrali**. L'utente dovrà quindi eseguire il login

In entrambi i casi il pulsante **ANNULLA** semplicemente nasconde la relativa finestra

Parte 2 : Google Instant

in corrispondenza della pressione di un tasto sul Text Box di ricerca (evento keyup), il client (**se il textbox non è vuoto**) invia una richiesta GET richiedendo al server **l'elenco delle voci contenenti tutte le parole fino ad ora digitate dal client**. Come parametro invia il contenuto del Text Box di ricerca.

il server in corrispondenza della richiesta esegue le seguenti operazioni:

- Esegue uno split delle parole ricevute, sulla base del carattere spazio
- Esegue una query che ricerca tutte le voci che contengono al loro interno **tutte** le parole ricevute come parametro. Per ogni parola occorre impostare una regex del tipo:

```
let regex = new RegExp("\\b" + voce + ".*", "i")
```

\b è il delimitatore di parola. Cioè se l'utente digita **s**, occorre restituire tutte le voci che contengono almeno una parola che inizia per s. Se l'utente digita **s f**, occorre restituire tutte le voci che contengono almeno una parola che inizia per s ed almeno una parola che inizia per f. Per maggiori dettagli si veda il filmato allegato
- In caso di corrispondenze, il server restituisce tutte le voci trovate **ordinate in ordine decrescente sulla base del campo nClick**.
- Nel caso in cui non ci siano corrispondenze, semplicemente restituisce un vettore vuoto

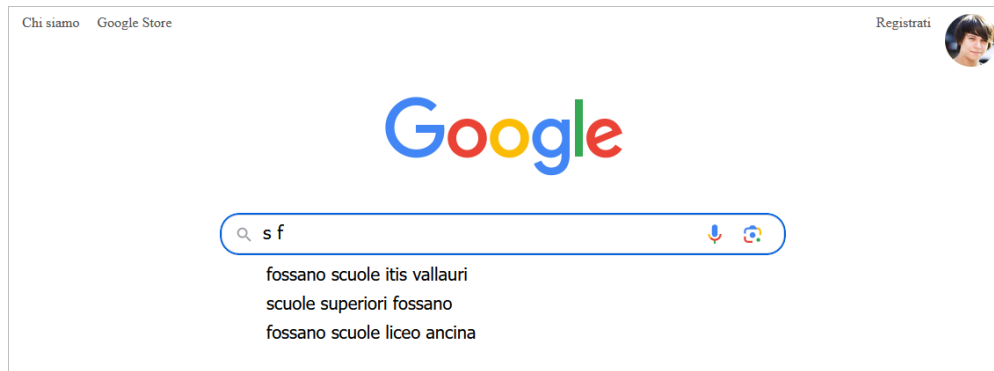
In corrispondenza della risposta, il client provvede a:

- Visualizzare il ListBox dei suggerimenti (inizialmente nascosto) impostando la proprietà html **size** (che rappresenta il numero di righe da visualizzare) al valore del numero di record ricevuti + 1.
- Eliminare dal ListBox eventuali vecchie voci
- Aggiungere nel ListBox le varie voci ricevute, ognuna opportunamente serializzata

ISTITUTO TECNICO INDUSTRIALE STATALE "G. VALLAURI"

▲ Via S. Michele, 68 – 12045 Fossano

Indirizzo **INFORMATICA** - Tecnologie



In corrispondenza **del click su una delle voci del List Box** il client deve:

- Copiare la voce selezionata all'interno del campo di ricerca
- nascondere il ListBox delle voci
- visualizzare l'apposito contenitore **divLink**.
- inviare una richiesta PATCH al server passandogli come parametro l'_id della voce selezionata

Il server deve :

- Incrementare di 1 il contatore dei click di quella voce
- Restituire al client **tutti** i link relativi alla voce selezionata.

Il client visualizzerà tutti i dati ricevuti all'interno dell'apposito contenitore **divLink** utilizzando semplicemente, per ogni riga, un **tag p** per la descrizione ed un **tag a** per il link.

