

Corso di Programmazione Web e Mobile

A.A. 2020-2021

Info Valtellina

◀ Lorenzo Molinari, 976472 ▶

Autore:	Lorenzo Molinari
Ultima modifica:	27 giugno 2021 - versione 1.3
Prima modifica:	10 giugno 2021

Indice

1. Introduzione	3
2. Breve Analisi dei requisiti	3
3. Interfacce.....	6
4. Architettura	10
5. Codice	12
6. Nota bibliografica e sitografia	19

Info Valtellina

Website Progetto Programmazione Web e Mobile

1. *Introduzione*

Info Valtellina è un applicazione web che permette di consultare le ultime informazioni (News, Tweet, Meteo) riguardante la Valtellina, una regione geografica alpina, corrispondente al bacino idrico del fiume Adda a monte del lago di Como, nella regione Lombardia, o più in generale della provincia di Sondrio. Per realizzarla ho utilizzato cinque differenti API:

- <https://openweathermap.org/api> (Meteo)
- <https://newsapi.org/> (News)
- <https://developer.twitter.com/en/docs/twitter-api> (Tweet)
- <https://cloud.google.com/maps-platform/maps> (Mappa)
- <https://developers.google.com/fonts/> (Font)

2. *Breve Analisi dei requisiti*

2.1 Destinatari

Il progetto è pensato per essere utilizzato da qualsiasi persona, in particolare quei soggetti poco pratici delle tecnologie informatiche, in quanto si tratta di un sito web di facciata attraverso il quale è possibile interagire con l'utilizzo di bottoni o attraverso lo scorrimento.

La scelta dell'utilizzo di un'interfaccia semplice e non dispersiva e proprio data per la totale fruizione agli utenti (anche meno esperti) di tutte le funzionalità del sito web. Gli utenti non possono essere distinti in macro-campi in quanto si tratta di un'applicazione web adatta ad ogni persona che vuole aggiornarsi sugli avvenimenti riguardanti la provincia di Sondrio, anche per questo la lingua utilizzato è l'italiano in quanto si possa pensare che l'utenza sia limitata o maggiormente rappresentata da persone la cui lingua madre è l'italiano.

Il sito web è stato reso dinamico e responsivo in modo di potersi adattare a qualsiasi dispositivo su cui viene visualizzato come: smartphone, tablet, pc ma anche televisori o proiettori (in quanto potrebbe essere utilizzato da enti comunali per esporre le notizie locali in luoghi pubblici).

2.2 Modello di valore

Questa applicazione fa riferimento ad un servizio già ampiamente presente sul web, ma aggiunge la specifica locale per poter incentrare il focus a livello provinciale (e non regionale o nazionale come già presente su internet), il sito è quindi adattabile, qualora fosse richiesto, anche a macrogruppi locali al di fuori di quello specifico in essere (Provincia di Sondrio), per fare un' esempio potrebbe essere utilizzato per raccogliere le informazioni della Provincia di Bergamo, oppure per zone più piccole come la Val Seriana (una valle di medie dimensioni presente nel comprensorio Bergamasco) fino addirittura fruire per un comune come ad esempio Lovere (Bergamo). Questo è possibile attraverso l'utilizzo delle API che ci permettono di cambiare intuitivamente i parametri in base alle nostre esigenze

La scelta di una tecnologia Client-Server, inoltre, ci permette di poter allargare il bacino di utenza in base alle esigenze e in futuro potrebbe anche essere possibile eseguire modifiche lato server senza dover influenzare il corretto funzionamento dell'applicazione lato client.

Per poter adottare però l'applicazione su una scala di utenti più ampia saranno necessarie costi di gestione, sia per il corretto funzionamento e mantenimento dell'applicazione, sia per l'utilizzo delle API che, superato un numero limitato di richieste, non permettono più di ottenere il file JSON contenente le informazioni che poi saranno visualizzate sullo schermo del client.

2.3 Flusso di dati

Come già specificato nel paragrafo precedente, i dati visualizzati lato client sono presi e successivamente elaborati da delle API (Application Programming Interface) che attraverso l'inserimento di un URL specifico ci restituiscono un file JSON contenente le informazioni da noi richieste.

Le API utilizzate in questa applicazione web sono: OpenWeatherAPI, newsAPI, googleMapsAPI e TwitterAPI:

- Le OpenWeather API sono utilizzate in diversi modi all'interno del sito web: nella lateral bar per poter visualizzare il meteo delle città più importanti della provincia di Sondrio, all'interno della mappa per poter avere una visione generale del meteo della provincia e dintorni e sempre nella lateral bar è possibile visualizzare il meteo in base alla propria geolocalizzazione del luogo.
- Le API di newsAPI sono utilizzate per poter estrapolare le notizie locali (in base ad una ricerca che passa come parametro "q=Valtellina") su un largo bacino di agenzie di stampa che tra gli altri comprendono: Ansa, AdnKronos, Corriere, OASport e Mediaset.

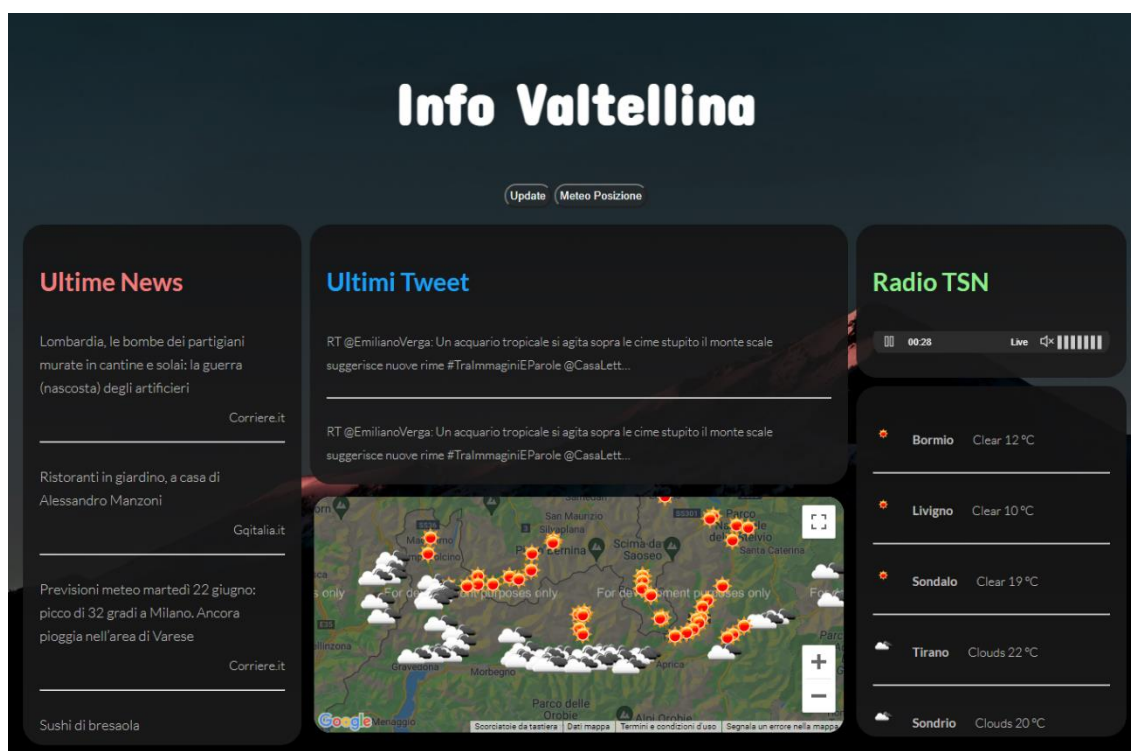
- Attraverso GoogleMapsAPI abbiamo la possibilità di visualizzare una mappa centrata appositamente al centro della Provincia di Sondrio.
- Infine, con le TwitterAPI e attraverso il modulo node twit è possibile estrapolare dal database di twitter tutti i tweet che presentano al loro interno la parola Valtellina (la scelta di utilizzare twitter rispetto ad altri social riguarda la ormai nota caratteristica di avere utenti più attivi a livello di avvenimenti anche locali)

2.4 Aspetti tecnologici

La gestione del codice avviene tramite il modello Model View Controller. La logica di presentazione del contenuto è affidata al modulo node ejs che si occupa di gestire ciò che l'utente deve visualizzare una volta caricata la pagina nel proprio browser. Le tecnologie richieste sono state inserite, con l'aggiunta dei moduli twit e fs per il corretto funzionamento di node. Queste scelte (in particolare quella di twit) sono state fatte per una maggiore comprensione ed elaborazione dei moduli scelti.

3. Interfacce

3.1 Home Layout



Layout di Base, quando si accede all'applicazione (oppure quando si premerà il pulsante update) la pagina verrà visualizzata in questo modo.

- Nella barra laterale a sinistra troviamo le ultime News della Valtellina;
- Nella body centrale troviamo due div che dividono superiormente gli ultimi dieci tweet in cui era presente la parola Valtellina ed inferiormente la mappa del meteo focalizzata su Sondrio;
- Nella barra laterale a destra troviamo superiormente il collegamento a Radio Tele Sondrio News (TSN) radio locale della provincia, mentre inferiormente il meteo delle città più importanti della Valtellina.

3.2 Layout Geolocation



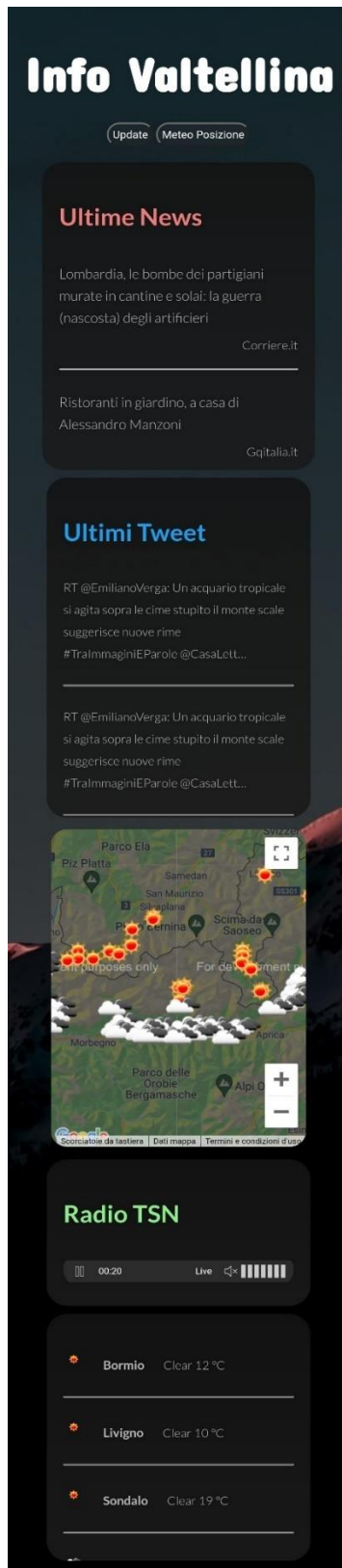
Se viene premuto il pulsante Meteo Posizione il client, dopo aver preso la posizione attraverso le nuove API di HTML5, le invia al server node che posizionerà al posto del meteo delle città più importanti il meteo della posizione passata dal browser con: temperatura, temperatura percepita, meteo, velocità del vento e umidità.

3.3 Layout News



Se invece viene premuto su una qualsiasi notizia verrà posto in sostituzione degli ultimi tweet una div contenente: il provider della notizia, il titolo della notizia, il testo vero e proprio della notizia, l'immagine collegata all'articolo e infine il link per raggiungere la notizia sul provider che l'ha pubblicata.

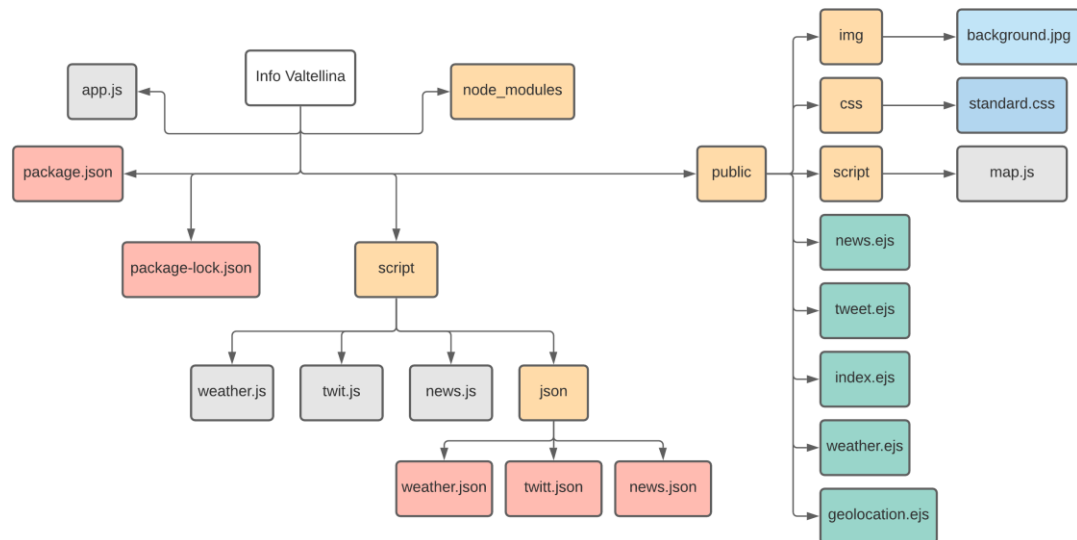
3.4 Layout Mobile



Esempio di schermata rispetto come viene visualizzato il sito web da uno smartphone con display di medie dimensioni (6.41 pollici) [screenshot allungato].

4. Architettura

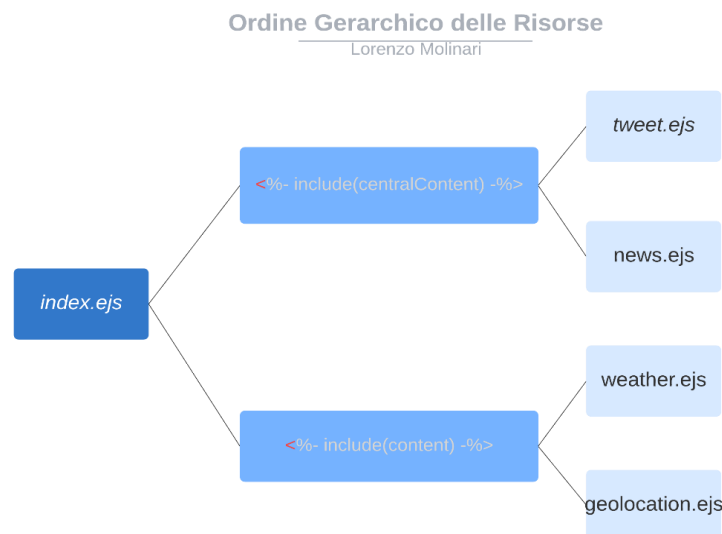
4.1 Diagramma delle directory



Legenda:

- **Arancione:** Cartelle
- **Grigio:** Javascript
- **Rosa:** File JSON
- **Verde:** File EJS
- **Blu:** File CSS
- **Azzurro:** Immagini

4.2 Diagramma dell'ordine gerarchico delle risorse



La pagina non è suddivisa in più file che vengono caricate in base alla richiesta, bensì all'interno della pagina viene adattato il contenuto in base alla richiesta tramite URL. Questa scelta ricalca sempre la facilità di utilizzo per gli utenti non avvezzi all'utilizzo della tecnologia.

4.3 Descrizione delle risorse

Lo scambio di risorse mediante l'interazione dei componenti del MVC è ricorrente all'interno di ogni pagina e avviene nello stesso modo per tutto l'arco di funzionamento dell'applicazione web, ovvero, a seguito dell'interazione dell'utente riguardo una parte dell'interfaccia grafica avverrà uno scambio di dati (in particolare in JSON) per potere offrire al client il risultato da lui richiesto.

5. Codice

Frammenti del codice più significativo

5.1 HTML

```
<!-- HomePage -->
<div id="homepage">
  <div id="news" class="flex-width45">
    <h1>Ultime News</h1>
    <div class="post-news">
      <% if(articles !=null) { %>
        <% articles.forEach(function(article, index) { %>
          <a href="/article/<%- article.title %%" class="news__card">
            <p class="normal-text">
              <%- article.title %>
            </p>
            <p class="news_source">
              <%- article.source.name %>
            </p>
          </a>
          <hr />
        <% } %>
      <% } else { %>
        No posts found.
      <% } %>
    </div>
  </div>

  <div id="centralBody" class="flex-width70">
    <div class="tweetAndNews">
      <%- include(centralContent) -%>
    </div>
    <div class="meteoMap">
      <div id="map"></div>
    </div>

    <div id="lateralBar" class="flex-width60">
      <div class="radioPlayer">
        <a href="https://www.radiotsn.tv/" target="_blank"><h2>Radio TSN</h
2></a>
        <iframe id="ascolta" src="https://player.streamshow.it/embed/tsnrad
io.php?" autoplay="0" frameborder="0"
          marginwidth="0" marginheight="0" width="100%" height="45"></ifr
ame>
      </div>
      <div class="others">
        <%- include(content) -%>
      </div>
    </div>
  </div>
```

Questa parte di codice presa dal file index.ejs riguarda la parte centrale del sito web. Qui possiamo notare come in due situazioni: nella div “tweetAndNews” e nella div “others” sia presente codice javascript (<%- include... %>), questo perché attraverso node ed express è possibile gestire e caricare internamente alle pagine html ulteriori sezioni o piccole pagine, rendendo il sito web (seppur visualmente statico) dinamico e che cambia in base alle esigenze.

5.2 CSS

```
/* Stile */

body {
  font-family: 'Lato', sans-serif;
  font-size: 15px;
  background: linear-
gradient(0deg, rgba(0, 0, 0, .2), rgba(0, 0, 0, 0.3)), url('/img/nathan-anderson-
5xZMSUEcze0-unsplash.jpg') no-repeat center center;
  background-size: cover;
}

a {
  text-decoration: none !important;
  color: rgb(190, 190, 190);
}

a:hover {
  color: #00acee !important;
}

h1, h2, h3, h4, h5, p {
  color: white;
  padding-top: 10px;
  padding-bottom: 10px;
}

p {
  line-height: 28px;
  color: rgb(190, 190, 190);
  font-weight: 300;
}

em {
  font-style: italic;
}

.big-text {
  font-size: 70px;
  font-weight: 900;
}

.large-text {
  font-size: 50px;
  font-weight: 700;
}

.med-text {
  font-size: 40px;
}
```

Parte iniziale del file standard.css, riguarda le principali specifiche grafiche che devono intercorrere all'interno della pagina, in questo specifico riguarda gli elementi più utilizzati nella costruzione di pagine web.

5.3 API (JSON)

```
{
  "cnt":8,
  "list":[{"
    "coord":{"
      "lon":10.3721,
      "lat":46.4684
    },
    "sys":{"
      "country":"IT",
      "timezone":7200,
      "sunrise":1624678033,
      "sunset":1624734921
    },
    "weather":[{"
      "id":801,
      "main":"Clouds",
      "description":"poche nuvole",
      "icon":"02d"
    }],
    "main":{"
      "temp":17.91,
      "feels_like":17.01,
      "temp_min":16,
      "temp_max":20.56,
      "pressure":1022,
      "humidity":48
    },
    "visibility":10000,
    "wind":{"
      "speed":4.63,
      "deg":220
    },
    "clouds":{"all":20},
    "dt":1624702172,
    "id":3181730,
    "name":"Bormio"
  }
}
```

Sezione della risposta che riceve il server dall'api OpenWeather per il meteo delle città più importanti.

```

{
  "status": "ok",
  "totalResults": 43,
  "articles": [
    {
      "source": {
        "id": null,
        "name": "Gqitalia.it"
      },
      "author": "Olga Winderling",
      "title": "Ristoranti in giardino, a casa di Alessandro Manzoni",
      "description": "Tra i migliori ristoranti all'aperto di Milano, Vòce in Giardino rinsalda il proprio legame con il territorio e la sua storia: siamo nel centralissimo Giardino di Alessandro - che collega le Gallerie e d'Italia, Piazza Scala e il museo di Intesa Sanpaolo alla C...",
      "url": "https://www.gqitalia.it/lifestyle/article/ristoranti-allaperto-con-giardino-milano-voce-aimo-nadia-alessandro-manzoni",
      "urlToImage": "https://media.gqitalia.it/photos/60d5a41a8edbbaf00f7e133d/16:9/w_1920,c_limit/Voce%20In%20Giardino_VoceAimoeNadia_ph%20Marco%20Tabasso%20(8).jpg",
      "publishedAt": "2021-06-26T04:30:00Z",
      "content": "Tra i migliori ristoranti all'aperto di Milano, Vòce in Giardino rinsalda il proprio legame con il territorio e la sua storia: siamo nel centralissimo Giardino di Alessandro - che collega le Gallerie... [+4526 chars]"
    },
    {
      "source": {
        "id": null,
        "name": "Corriere.it"
      },
      "author": "Redazione Milano",
      "title": "Previsioni meteo martedì 22 giugno: picco di 32 gradi a Milano. Ancora pioggia nell'area di Varese",
      "description": "Clima estivo e a tratti afoso a Milano nella giornata di martedì 22 giugno. Tempo ancora instabile nella zona di Varese, con qualche possibile temporale residuo, da mercoledì in estensione sui rilievi di Valtellina, Bergamo e Brescia",
      "url": "https://meteo.corriere.it/news/21_giugno_21/previsioni-meteo-martedi-22-giugno-milano-torna-sole-ancora-pioggia-varesotto-lombardia-orientale-c89c63d8-d28f-11eb-9207-8df97caf9553.shtml",
      "urlToImage": "https://images2-milano.corriereobjects.it/methode_image/socialshare/2021/06/21/dd7d18d0-d292-11eb-9207-8df97caf9553.jpg",
      "publishedAt": "2021-06-21T13:36:47Z",
      "content": "Chiusa la breve (ma intensa) parentesi quasi autunnale di pioggia e vento forte che nelle scorse ore ha colpito la Lombardia, e in particolare la zona di Varese, dove le forti raffiche di vento hanno... [+1139 chars]"
    }
  ],

```

Sezione della risposta che riceve il server dall'api newsAPI per le notizie riguardanti la Valtellina.

5.4 Node.js

```

const update = async () => {
  weather.getWeather()
  news.getNews()
  twit.getTweet()
  try {
    twitAPI = JSON.parse(await fs.readFile(__dirname + '/script/json/twitt.json', 'utf8', (error, data) => {
      if (err) {
        console.error(err)
      }

      return data
    })))
    weatherAPI = JSON.parse(await fs.readFile(__dirname + '/script/json/weather.json', 'utf8', (err, data) => {
      if (err) {
        console.error(err)
      }

      return data
    })))
    newsAPI = JSON.parse(await fs.readFile(__dirname + '/script/json/news.json', 'utf8', (err, data) => {
      if (err) {
        console.error(err)
      }

      return data
    })))
  } catch (err) {
    console.error('Error: ', err)
  }
}

const renderHome = async (req, res) => {

  update()

  try {
    res.render('index', { twitt: twitAPI.statuses, weather: weatherAPI.list, articles: newsAPI.articles, centralContent: 'tweet', content: 'weather' })
  } catch (err) {
    res.render('index', { twitt: null, weather: null, articles: null, centralContent: 'tweet', content: 'weather' })
    console.error('Error: ', err)
  }
}

//gestisco index.ejs
app.get('/', renderHome)

```

In questa sezione del codice javascript eseguito da node si trova la gestione e la renderizzazione della pagina home (principale) quando viene richiesta (“https://localhost:port”).

5.5 Javascript (interno ad HTML)

```
function geolocation() {  
    if (navigator.geolocation) {  
        try {  
            const position = navigator.geolocation.getCurrentPosition(succ  
ss, error)  
  
            function success(pos) {  
                const lat = pos.coords.latitude  
                const lon = pos.coords.longitude  
  
                window.location.replace("/geolocation?lat=" + lat + "&lon=" + lon);  
            }  
  
            function error(error) {  
                console.log(error)  
                alert("Errore, impossibile recuperare la posizione");  
            }  
        } catch (error) {  
            console.log(error)  
            alert("Errore, impossibile recuperare la posizione");  
        }  
    } else {  
        console.log("Geolocation is not supported by this browser.");  
        alert("Geolocation is not supported by this browser.");  
    }  
};  
</script>
```

In questo frammento di codice viene gestito il prelievo della posizione tramite API di HTML5 e il successivo invio al server node della richiesta con le coordinate per la geolocalizzazione che verranno prelevate attraverso req.query.lon e req.query.lat.

5.6 Validator CSS e HTML

Nu Html Checker

This tool is an ongoing experiment in better HTML checking, and its behavior remains subject to change

Showing results for contents of text-input area

Checker Input

Show ☒ source ☐ outline ☐ image report [Options...](#)

Check by ☒ text input ☐ CSS

```
<!DOCTYPE html>
<html lang="it-IT">

<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>ValtellinaNews - Progetto Esame PwK</title>

  <!-- Stylesheet -->
  <link rel="stylesheet" href="/css/standard.css">

  <!-- Google Maps -->
  <script src="https://polyfill.io/v3/polyfill.min.js?features=default"></script>
  <script src="https://maps.googleapis.com/maps/api/js"></script>
</html>
```

[Check](#)

Press the Message Filtering button to collapse the filtering options and error/warning/info counts.

Message Filtering 10 messages hidden by filtering

Errors (10) · [Hide all errors](#) · [Show all errors](#)

- ☐ Bad character `%` after `<`. Probable cause: Unescaped `<`. Try escaping it as `<`. (9)
- ☐ Bad value `/article/<%- article.title %>` for attribute `href` on element `a`: Illegal character in path segment: `<` is not allowed.

Document checking completed. No errors or warnings to show.


Servizio di validazione del W3C per i fogli di stile a cascata (CSS)
Risultati della validazione W3C dei Fogli di Stile a Cascata (CSS) per TextArea (CSS versione 3 + SVG)

Passa a: [Avvisi \(3\)](#) [Foglio CSS validato](#)

Risultati della validazione W3C dei Fogli di Stile a Cascata (CSS) per TextArea (CSS versione 3 + SVG)

Congratulazioni! Nessun errore trovato.

Questo documento è valido rispetto alla specifica [CSS versione 3 + SVG](#)!

Per rendere visibile ai tuoi lettori che hai curato lo sviluppo per creare una pagina Web interoperabile, puoi pubblicare questa icona in qualsiasi pagina che rispetta la validazione. Di seguito il codice XHTML che puoi utilizzare per aggiungere questa icona alla tua pagina Web:



```
<p>
  <a href="http://jigsaw.w3.org/css-validator/check/referer">
    
  </a>
</p>
```



```
<p>
  <a href="http://jigsaw.w3.org/css-validator/check/referer">
    
  </a>
</p>
```

(Se si utilizza XHTML, chiudere l'elemento img con `</>`. Se si utilizza HTML <= 4.01, chiudere l'elemento img tag con `>` invece di `</>`)



Interested in understanding what new technologies are coming out of W3C? Follow [@w3cdevs on Twitter](#) to keep track of what the future looks like!

[Donate](#) and help us build better tools for a better web.

Se lo desideri, puoi scaricare una copia di questa immagine per salvarla all'interno del tuo sito Web, modificando il codice XHTML sopra riportato riferendolo all'immagine locale anziché a quella residente in questo server.

Se desideri creare un collegamento a questa pagina (per esempio, al risultato di validazione) per rendere più facile una futura riveduta di questa pagina o per consentire ad altre persone di verificare la conformità della tua pagina, l'URI è:

```
http://jigsaw.w3.org/css-validator/validator?link
or
http://jigsaw.w3.org/css-validator/check/referer (solo per documenti HTML/XML)
```

(Oppure puoi solamente aggiungere la pagina corrente ai tuoi collegamenti preferiti.)

Avvisi (3)

URI : TextArea	
95	<code>::-webkit-scrollbar</code> is an unknown vendor extended pseudo-element
312	<code>:-webkit-box-sizing</code> is an unknown vendor extension
313	<code>:-moz-box-sizing</code> is an unknown vendor extension

6. *Nota bibliografica e sitografia*

Dove si elencano le risorse bibliografiche e sitografiche usate come riferimento e documentazione del lavoro.

//Consultazione API

- (1) <https://newsapi.org/>
- (2) https://developers.google.com/maps/documentation/javascript/overview?hl=it#maps_map_simple-html
- (3) <https://openweathermap.org/>
- (4) <https://developer.mozilla.org/en-US/docs/Web/API/Geolocation/getCurrentPosition>
- (5) <https://twitframe.com/>
- (6) <https://developer.twitter.com/en/portal/projects/1408021421092134918/apps/21232654/keys>

//Consultazione porzioni di Codice

- (7) <https://raddy.co.uk/blog/build-news-website-with-node-js-express-ejs-wp-rest-api/>
- (8) <https://stackoverflow.com/questions/39947655/call-node-function-from-ejs-form>