

E-commerce di articoli Sportivi

Lorenzo Nobili

November 2023

1 Descrizione

Per questo progetto di Tecnologie Internet, abbiamo sviluppato un E-commerce dedicato al mondo del tennis. Il nostro sito offre una vasta gamma di prodotti per gli appassionati di questo sport, tra cui racchette, palline, abbigliamento, accessori e altro ancora. Il nostro sito offre anche le seguenti funzionalità:

- Registrazione al sito come cliente o venditore, con possibilità di creare un profilo personale e gestire i propri dati.
- Ricerca di prodotti per categoria, marca, prezzo, recensioni o parole chiave, con possibilità di ordinare e filtrare i risultati.
- Aggiunta o rimozione di un prodotto al carrello, con possibilità di modificare la quantità, visualizzare il totale e procedere al checkout.
- Acquisto tramite PayPal, con possibilità di scegliere l'indirizzo di spedizione, il metodo di pagamento e confermare l'ordine.
- Creazione e annullamento di una spedizione, con possibilità di tracciare lo stato della consegna e ricevere notifiche via email.
- Creazione ed eliminazione della lista acquisti, con possibilità di salvare i prodotti preferiti e visualizzare lo storico degli ordini.
- Supporto al cliente, con possibilità di contattare il venditore o l'amministratore del sito per qualsiasi domanda o problema.

Inoltre, l'amministratore del sito ha il controllo totale sulle operazioni del sito e può:

- Creare o eliminare un prodotto, inserendo o modificando le informazioni relative al nome, alla descrizione, al prezzo, alla disponibilità, alle immagini e alle recensioni.
- Creare o eliminare un utente, assegnando o revocando il ruolo di cliente o venditore, e gestendo i dati personali, gli ordini e le spedizioni.

- Rendere un cliente un venditore, concedendo il permesso di inserire e vendere i propri prodotti sul sito, e ricevere una percentuale sulle vendite.
- Vendere prodotti a sua volta, scegliendo tra quelli disponibili sul sito o inserendo i propri, e ricevendo il pagamento tramite PayPal.

Le tecnologie usate per lo sviluppo del progetto sono:

- **React:** React è una libreria JavaScript ottimale per gli sviluppatori frontend di app web e mobile. Consente la costruzione di interfacce utente dinamiche, interattive e facilmente gestibili in progetti di complessità media e grande. La sua facilità di integrazione la rende adatta anche per progetti esistenti. La presenza di un Virtual DOM migliora l'efficienza dei cambiamenti nel DOM attraverso una renderizzazione selettiva dei sottoalberi.
- **Axios:** Axios è un client HTTP leggero basato su promesse, che consente l'interfacciamento con un'API REST. L'utilizzo di promesse agevola la scrittura di codice asincrono leggibile. Axios offre anche funzionalità di intercettazione e annullamento delle richieste, con integrazione della protezione lato client contro la falsificazione delle richieste tra siti.
- **Redux:** Redux è un'architettura per applicazioni web basate su React, progettata per rendere prevedibile il cambio di stato. Si basa su un singolo stato conservato in uno store, mutabile solo attraverso azioni che attivano funzioni pure chiamate reducer. I tre principi di Redux, "Single source of truth", "State is read-only" e "Changes are made with pure functions", garantiscono coerenza e prevedibilità nello sviluppo del software. Le componenti principali di Redux includono:
 - **Store:** Mantiene lo stato dell'applicazione.
 - **Action:** Funzioni che dispatchano azioni, con la possibilità di contenere un corpo (body) utilizzato nei reducer per la mutazione dello stato.
 - **Reducer:** Gestisce lo stato attraverso uno switch che varia lo stato in base all'azione e, eventualmente, al suo contenuto.
 - **Connect:** Funzione utilizzata per collegare un componente alle azioni e alla parte dello stato necessaria, consentendo il passaggio delle proprietà dello stato quando cambiano.
 - **Selectors:** Si può notare come la selezione dello stato soffra del problema dell'usare una stringa che può causare problemi nel refactoring. Per ovviare a questo si possono creare delle funzioni in un unico luogo e che vanno a compiere quell'operazione. Questi sono i selectors.
 - **Container:** il concetto può essere interpretato come il componente contenitore di altri componenti che vanno a formare quello che nelle

vecchie applicazione web chiameremmo pagina. Quindi i container sono le nostre pagine, ma sono anche le cornici in cui esse sono contenute. Immaginiamo la classica applicazione con header e footer. Questi saranno 2 componenti contenuti in un container. In questo container si alterneranno altri container che saranno le nostre pagine.

- **Middleware:** come interagisce il nostro sistema con le API del back-end? Come si crea un sistema di logging trasparente? A queste domande risponde il termine middleware. Il middleware è una higher-order function che riceve una funzione di dispatch e ne restituisce un'altra potenziata.
- **Node:** Node.js è una piattaforma realizzata su V8, il motore Javascript di Chrome, che permette di realizzare applicazioni web veloci e scalabili. Node usa un modello ad eventi e un sistema di I/O non bloccante che lo rende leggero ed efficiente, perfetto per applicazioni real-time che elaborano dati in modo intensivo e che può essere distribuito su più sistemi.
- **Express:** Express è un framework per applicazioni web Node.js flessibile e leggero che fornisce una serie di funzioni avanzate per le applicazioni web e per dispositivi mobili.
- **Mongoose:** Mongoose è una libreria Object Data Modeling (ODM) per MongoDB e Node.js. Gestisce le relazioni tra i dati, fornisce la convalida dello schema e viene utilizzato per tradurre gli oggetti nel codice e la rappresentazione di tali oggetti in MongoDB. MongoDB è un database di documenti NoSQL senza schema. Significa che puoi archiviare documenti JSON al suo interno e la struttura di questi documenti può variare in quanto non viene applicata come i database SQL. Questo è uno dei vantaggi dell'utilizzo di NoSQL poiché accelera lo sviluppo delle applicazioni e riduce la complessità delle distribuzioni

2 Librerie Utilizzate

Abbiamo installato le seguenti librerie per lo sviluppo del progetto

- **npm install redux:** installa la libreria **redux**, che è un gestore di stato prevedibile per le applicazioni JavaScript. Redux aiuta a scrivere applicazioni che si comportano in modo coerente, girano in ambienti diversi (client, server e nativi) e sono facili da testare¹.
- **npm install react-router-dom:** installa la libreria **react-router-dom**, che è una versione di **react-router** specifica per il web. React Router è una raccolta di componenti di navigazione per le applicazioni React².

- `npm install react-responsive-carousel`: installa la libreria **react-responsive-carousel**, che è un componente React per creare caroselli reattivi e personalizzabili. Supporta immagini, video, miniature, indicatori, frecce, autoplay, swipe e molto altro³.
- `npm install react-redux`: installa la libreria **react-redux**, che è l'implementazione ufficiale di Redux per l'integrazione con React. React Redux permette di connettere i componenti React al negozio Redux, accedendo allo stato e alle azioni.
- `npm install react-paypal-button-v2`: installa la libreria **react-paypal-button-v2**, che è un componente React per integrare il pulsante PayPal v2 nelle applicazioni React. Supporta le opzioni di pagamento standard, sottoscrizione e ordine.
- `npm install react-google-charts`: installa la libreria **react-google-charts**, che è un componente React per utilizzare i grafici di Google nelle applicazioni React. Offre una varietà di tipi di grafici, come a barre, a torta, a linee, a dispersione, a bolle, a timeline, a calendario, a geo, a sankey e molti altri.
- `npm install react-dom`: installa la libreria **react-dom**, che è una parte del core di React. React DOM fornisce le funzioni per manipolare il DOM in modo efficiente e reattivo, usando gli elementi e i componenti React.
- `npm install react`: installa la libreria **react**, che è una libreria JavaScript per creare interfacce utente. React permette di creare componenti UI riutilizzabili e dinamici, gestendo lo stato e il ciclo di vita dei componenti.
- `npm install google-maps-react@2.0.6`: installa la libreria **google-maps-react**, alla versione 2.0.6, che è un componente React per integrare le mappe di Google nelle applicazioni React. Supporta le funzionalità di base delle mappe, come i marker, le info window, i poligoni, i cerchi, i rettangoli, le direzioni e i cluster.
- `npm install --save react-google-maps`: installa e salva la libreria **react-google-maps**, che è un'altra libreria per integrare le mappe di Google nelle applicazioni React. Offre una serie di componenti di alto livello, come GoogleMap, Marker, InfoWindow, Circle, e altri.
- `npm install axios`: installa la libreria **axios**, che è un client HTTP basato su promesse per il browser e Node.js. Axios permette di effettuare richieste HTTP asincrone, intercettare le richieste e le risposte, trasformare i dati delle richieste e delle risposte, annullare le richieste e gestire gli errori.
-
- `npm install react-scripts`: installa la libreria **react-scripts**, che è uno script e una configurazione predefinita per creare applicazioni React con **create-react-app**. React Scripts permette di avviare,

testare e compilare le applicazioni React senza dover configurare manualmente Webpack, Babel, ESLint e altri strumenti.

- `npm install jsonwebtoken`: installa la libreria **jsonwebtoken**, che è un'implementazione di JSON Web Token (JWT) per Node.js. JWT è uno standard per la creazione di token di accesso sicuri, che possono essere usati per l'autenticazione e l'autorizzazione.
- `npm install multer`: installa la libreria **multer**, che è un middleware per la gestione dei file multiforme in Express. Multer permette di accettare e salvare i file caricati dai client, usando il modulo **busboy**.
- `npm install bcryptjs`: installa la libreria **bcryptjs**, che è una versione puramente JavaScript di **bcrypt**. Bcrypt è una libreria per il hashing delle password, che usa l'algoritmo Blowfish e un sale casuale per rendere le password più sicure e resistenti agli attacchi.
- `npm install express-async-handler`: installa la libreria **express-async-handler**, che è un wrapper per i middleware e le route di Express che restituiscono una promessa. Express Async Handler permette di gestire gli errori asincroni in modo semplice e uniforme, usando il middleware di gestione degli errori di Express.
- `npm install dotenv`: installa la libreria **dotenv**, che è un modulo per caricare le variabili di ambiente da un file `.env` in Node.js. Dotenv permette di separare le configurazioni sensibili o specifiche dell'ambiente dal codice, rendendolo più portatile e sicuro.
- `npm install mongoose`: installa la libreria **mongoose**, che è un Object Data Modeling (ODM) per MongoDB e Node.js. Mongoose permette di definire schemi, validare i dati, creare relazioni, eseguire query e aggiornamenti e molto altro, usando una sintassi elegante e semplice.
- `npm install express`: installa la libreria **express**, che è un framework web minimalista e flessibile per Node.js. Express permette di creare applicazioni web e API robuste e scalabili, usando una serie di funzionalità, come il routing, il middleware, la gestione degli errori, i template e altro.
- `npm install -g nodemon`: installa globalmente la libreria **nodemon**, che è uno strumento per lo sviluppo di applicazioni Node.js. Nodemon monitora i cambiamenti nei file e riavvia automaticamente il server, rendendo lo sviluppo più rapido e produttivo.
- `npm install redux-thunk`: installa la libreria **redux-thunk**, che è un middleware per Redux. Redux Thunk permette di scrivere azioni asincrone che restituiscono una funzione invece di un oggetto. Questa funzione può poi eseguire operazioni asincrone, come chiamate API, e disacciare altre azioni quando necessario.

3 Struttura del progetto

Per il nostro sito abbiamo deciso di utilizzare un Flex layout, il Flex layout è composto da flex-container (il contenitore principale) e flex-elementi (elementi in esso incorporati). Gli elementi figli possono essere organizzati in una riga o colonna, e lo spazio rimanente viene distribuito tra loro in vari modi. Grazie a flexbox elementi si comportano in modo prevedibile su tutti i tipi di dispositivi e schermi di dimensioni diverse (il nostro sito è infatti utilizzabile anche da dispositivi mobile). Il modello di layout flessibile presenta tanti vantaggi rispetto alla struttura del blocco a causa della assenza d'elementi galleggianti e l'assenza del sovrapposizione del margine margin esterna. I Flex-elementi figli possono essere posizionati in qualsiasi direzione e nello stesso tempo hanno dimensioni flessibili, il cio permette a loro di adattarsi alle dimensioni dello schermo.