

OpenFridge

Progetto per la progettazione e sviluppo di applicazioni web per dispositivi mobili di Valentina Giannini e Deniz Tümer.

Introduzione:

Il progetto sviluppato è una web application per la gestione di un eCommerce simulato, creato utilizzando il framework Vue.js. L'obiettivo principale del progetto è quello di fornire un'interfaccia utente interattiva che consenta agli utenti di navigare tra i prodotti disponibili, aggiungerli al carrello, salvarli nella lista dei preferiti e visualizzare i dettagli di ogni prodotto. Inoltre, è stata implementata una funzionalità di ricerca e la possibilità di aggiungere nuovi prodotti, il tutto gestito tramite lo stato globale dell'applicazione con Vuex, simulando un database.

Componenti principali:

L'applicazione è composta dai seguenti componenti principali:

- HomeView.vue: La vista principale che mostra la lista dei prodotti disponibili. Gli utenti possono visualizzare i dettagli di ciascun prodotto, aggiungerlo al carrello o alla lista dei preferiti.
- ProductDetail.vue: Una vista dedicata per mostrare i dettagli di un singolo prodotto quando l'utente clicca su "Vedi Dettagli" dalla Home. Mostra informazioni aggiuntive sul prodotto.
- CarrelloView.vue: Una pagina che visualizza i prodotti aggiunti al carrello, il prezzo totale e offre la possibilità di rimuovere prodotti dal carrello.
- WishList.vue: Una pagina che visualizza tutti i prodotti aggiunti ai preferiti, permettendo di visualizzare e gestire i prodotti desiderati dall'utente.

Gestione del Carrello

Gli utenti possono navigare tra i prodotti e aggiungerli al carrello cliccando sul pulsante "Aggiungi al carrello". Il carrello viene gestito tramite Vuex, che tiene traccia dei prodotti e aggiorna dinamicamente il prezzo totale. Nella pagina del carrello, gli utenti possono visualizzare la lista dei prodotti selezionati, il loro prezzo, e possono rimuovere prodotti dal carrello. La struttura di gestione del carrello è stata realizzata utilizzando le seguenti funzionalità di Vuex:

- `add_cart`: Aggiunge un prodotto al carrello, aggiornando anche il prezzo totale.
- `remove_cart`: Rimuove un prodotto dal carrello e aggiorna il prezzo totale.

Gestione della Lista dei Preferiti

Similmente al carrello, gli utenti possono aggiungere prodotti alla lista dei preferiti cliccando sul pulsante "Aggiungi ai preferiti". Questi prodotti sono visualizzati in una pagina dedicata chiamata "WishList". La lista dei

preferiti è anch'essa gestita tramite Vuex, con la seguente mutazione:

- *add_wishlist*: Aggiunge un prodotto alla lista dei preferiti.

Visualizzazione del Dettaglio Prodotto

Ogni prodotto nella lista ha un pulsante "Vedi Dettagli" che reindirizza l'utente a una pagina separata per visualizzare informazioni più dettagliate sul prodotto. Questa funzionalità è stata implementata usando il routing di Vue Router, con una route dinamica per ogni prodotto basata sul suo ID.

Ricerca Prodotti

È stato implementato un form di ricerca nella HomeView che permette agli utenti di cercare prodotti in base al nome. La ricerca è dinamica e i risultati vengono aggiornati in tempo reale man mano che l'utente digita la query. Questa funzionalità è gestita tramite un getter di Vuex, che filtra i prodotti in base alla query di ricerca:

- *searchProducts*: Filtra i prodotti in base alla stringa di ricerca.

Aggiunta di Nuovi Prodotti

L'applicazione permette agli utenti di aggiungere nuovi prodotti tramite un form nella Home. Questo simula l'aggiunta di contenuti a un database, e i prodotti vengono immediatamente visualizzati nella lista prodotti dopo l'invio del form. La funzionalità è implementata tramite la seguente mutazione di Vuex:

- *addProduct*: Aggiunge un nuovo prodotto alla lista di prodotti simulando l'inserimento in un database.

Gestione dello Stato Globale con Vuex

Tutte le funzionalità di gestione dei prodotti, del carrello, e dei preferiti sono centralizzate nello stato globale dell'applicazione grazie a Vuex.

- **Stato (state)**: Memorizza i prodotti disponibili, i prodotti aggiunti al carrello, i preferiti, e il prezzo totale del carrello.
- **Mutazioni (mutations)**: Aggiornano lo stato dell'applicazione in risposta a determinati eventi, come l'aggiunta o la rimozione di prodotti dal carrello e dai preferiti.
- **Getters (getters)**: Forniscono funzioni utili per ottenere i dati necessari, come il filtro dei prodotti in base alla ricerca.

Uso di Vue Router

Vue Router è stato utilizzato per gestire la navigazione tra le diverse pagine dell'applicazione:

- **Route dinamiche**: Per la visualizzazione del dettaglio del prodotto, è stata utilizzata una route dinamica `/product/:id`, che carica i dettagli del prodotto corrispondente basandosi sull'ID passato come parametro.

L'installazione

Per questo progetto sono stati installati questi strumenti:

- **Node.js** scaricato da <https://nodejs.org>.
- **Vue CLI**: Strumento per la gestione dei progetti Vue, installato tramite npm.

Dopo aver installato Node.js è stato installato Vue CLI utilizzando il comando `npm install -g @vue/cli`. Per creare il progetto è stato prima fatto una *vue create*, successivamente è stata fatta una selezione tra le opzioni di configurazione manualmente in cui è stato scelto di usare Vuex e come formattazione del codice Prettier. Infine sono state installate le dipendenze necessarie con `npm install` e avviata in modalità di sviluppo con il seguente comando `npm run serve`.

Il progetto sviluppato offre un'applicazione eCommerce, simulando un database di prodotti e offrendo funzionalità chiave come la gestione del carrello, la gestione dei preferiti, la ricerca di prodotti e l'aggiunta di nuovi prodotti. L'integrazione di Vuex per la gestione dello stato globale permette di mantenere una struttura organizzata e reattiva, facilitando l'espansione futura dell'applicazione. Il codice è ben strutturato e facilmente scalabile, permettendo di aggiungere ulteriori funzionalità come un sistema di pagamento o la gestione degli utenti.