

Vinili & vinili

Fadi Marco Alrazem - 298913

Summary

- ▶ Il progetto, basato su node.js e react, è stato suddiviso in due cartelle, rispettivamente backend e frontend.
- ▶ Nella prima troviamo tutti i file relativi alle API, il db gestito da MongoDB, express, mongoose, data.js, bcryptjs, jsonwebtoken
- ▶ Nella seconda invece c'è tutta la parte che riguarda l'applicativo in sé, in cui ho inserito tutti i .js per le varie funzioni del sito web e inserito tutte le pagine che verranno visualizzate in esso.
- ▶ In app invece ho restituito tutta la parte riguardante il frontend, con axios e le vari implementazioni di react. La parte grafica è invece tutta compresa in index.css

App.js

In app ho importato tutte le pagine screens precedenti, toastify e axios per il corretto funzionamento e connessione anche col backend.

Si Presenteranno tutti i pulsanti, i prodotti e le funzionalità di ricerca presentate prima

Comunicherà tramite slug per stringhe che presentano un codice di tipo UNICODE e con le api tramite Axios.

Frontend - components

- ▶ Le sottocartelle di src comprendono le varie funzioni dell'applicativo:
 1. Checkoutsteps si riferisce a tutta la procedura per completare l'ordine
 2. Loadingbox è una semplice animazione di caricamento
 3. MessageBox è un alert
 4. Product è l'oggetto prodotto in cui è presente il dialogo con l'api del backend e la componente link di react
 5. Rating è una semplice funzione per restituire le reviews dei prodotti
 6. SearchBox è il bottone di ricerca per il sito web

Frontend - screens

Sono le pagine presenti in tutto il sito web:

1. CartScreen è la pagina subito prima il checkout
2. HomeScreen è la homepage del sito web
3. OrderHistoryScreen ti permette di visualizzare lo storico degli ordini
4. OrderScreen per la visualizzazione corretta di tutti i passaggi per il completamento di un ordine
5. PaymentMethodScreen per i metodi di pagamento con possibilità di salvarli
6. PlaceOrder è la pagina finale dopo il completamento di un ordine
7. ProductScreen è la pagina che si presenta dopo aver cliccato su un prodotto
8. ProfileScreen per la modifica dei dati di un utente
9. SearchScreen per gestire la ricerca di un prodotto tramite vari parametri
10. ShippingAddress è una parte del checkout che riguarda l'indirizzo di spedizione
11. SignIn ti chiederà i parametri per il login dell'utente
12. SignUp ti farà registrare e questi dati verranno inseriti nel DB
13. Store è il negozio con le sue varie funzionalità

Backend

Il backend comunicherà tramite server all'indirizzo ip 127.0.0.1 ed avrà una propria porta, che in questo caso ho scelto essere :5000.

Ho importato tutti i vari componenti che poi andrò a spiegare e tramite mongoose mi connetterò al database creato su MongoDB.

In utils invece ho fatto in modo che per ogni utente verrà generato il relativo token con scadenza di 30gg, per far sì che ogni utente sia protetto.

Le password saranno criptate tramite bcryptjs, che creerà una stringa random diversa per la password di ogni utente.

Backend - models

Sono i modelli creati nella backend

1. orderModel in cui viene presentato com'è impostato un ordine
2. productModel è relativo al prodotto
3. userModel all'utente che può essere admin o meno

Si utilizza mongoose per definire uno SCHEMA da utilizzare poi per la creazione di un modello da esportare

Backend - routes

Include tutte le funzioni router, che serviranno per le varie feature del backend:

1. `orderRoutes` per la gestione degli handler riguardanti un ordine (es. se ci sono errori oppure no)
2. `productRoutes` per la gestione di quelli basati sui prodotti, con possibilità di filtraggio in base alla categoria ecc..
3. `seedRoutes` per la comunicazione e creazione di nuovi utenti o prodotti con il server
4. `userRoutes` per tutto quello che riguarda gli user (sign-in sign-up e profilo) in cui sono presenti anche elementi di crittografia come token e bcrypt.

THE END
grazie per l'attenzione!

