

Relazione Progetto Tecnologie Internet

Per la realizzazione del progetto di Tecnologie Internet, abbiamo deciso di realizzare un e-Commerce di moto e accessori con le tecnologie viste durante il corso:

- HTML
- JAVASCRIPT (JSX)
- CSS
- REACT
- MySQL per il database

-PRESENTAZIONE

L'utente, si trova di fronte alla seguente schermata :

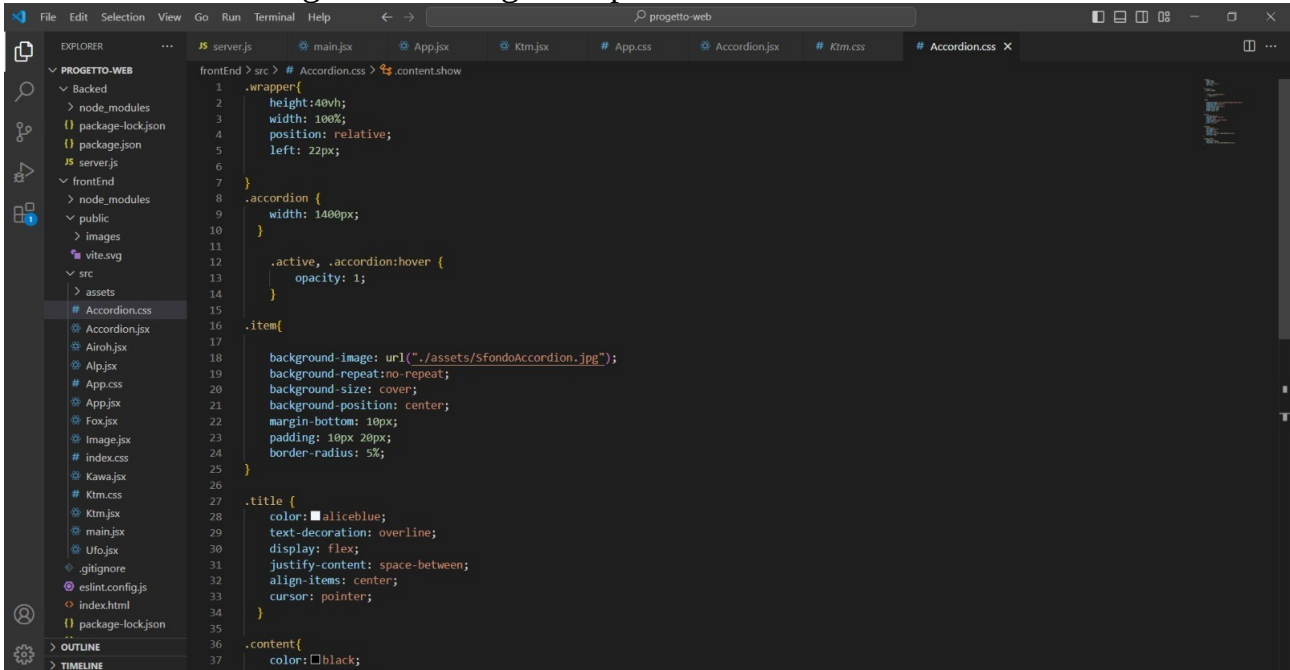


Nel menù a tendina che trova, l'utente può selezionare:

- News : per le ultime news sul mondo delle due ruote
- Contact : con tutte le info per contattarci
- About us : per la storia del nostro e-commerce

Questi sono stati realizzati usando HTML e CSS per la parte grafica e JSX per il contenuto dei tre elementi:

-Con CSS abbiamo gestito il design e la presentazione

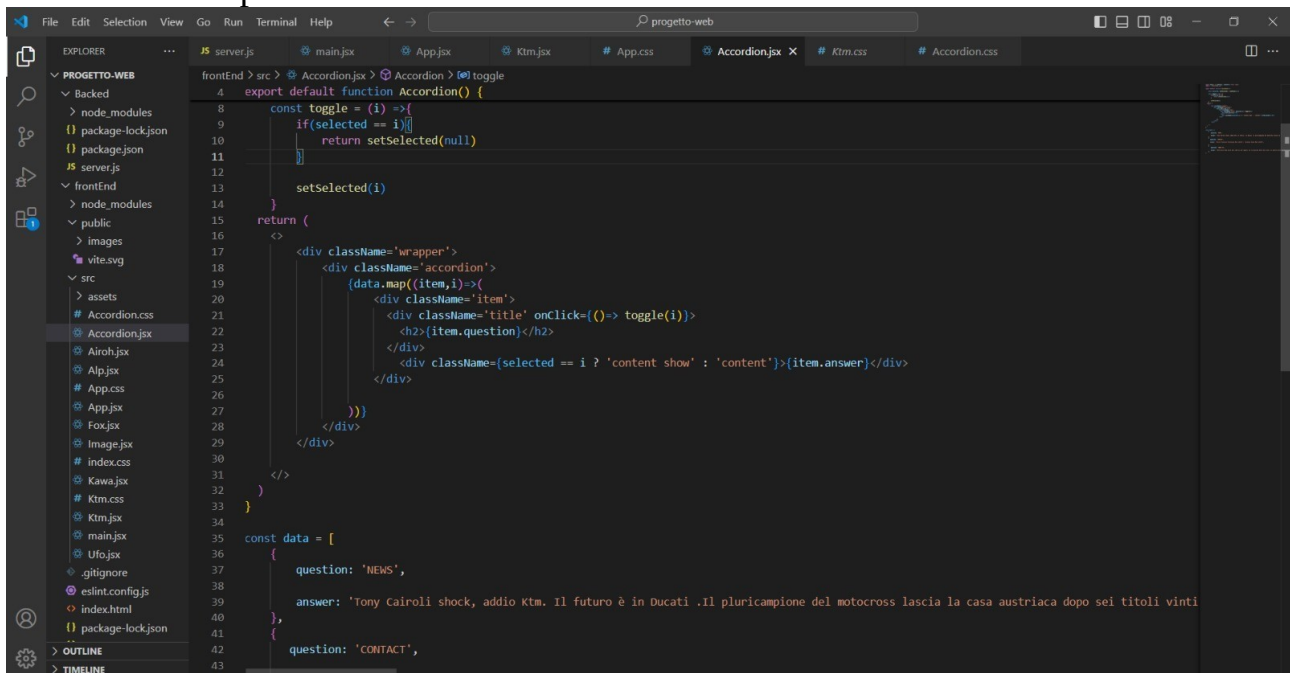


Con la classe **.item** abbiamo settato lo sfondo del menù a tendina:

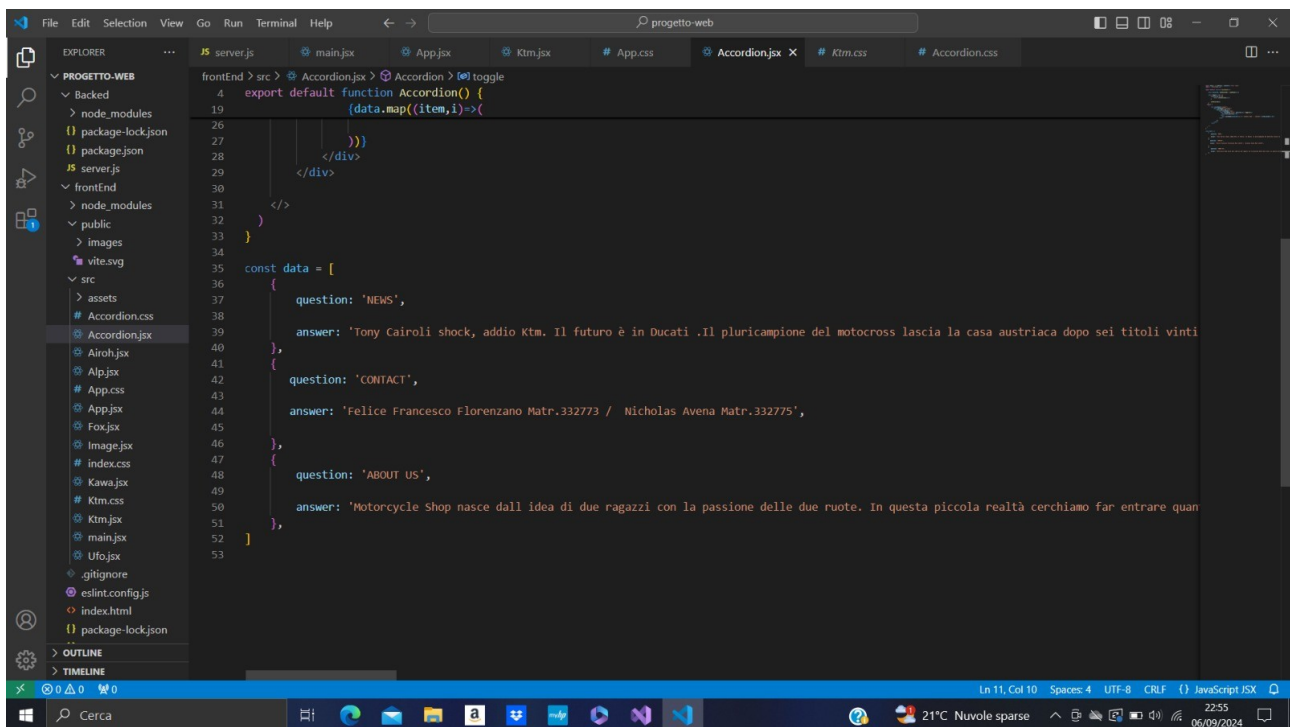
- background-image per selezionare l'immagine di sfondo
- background-repeat per ripetere o meno l'immagine di sfondo
- background-size per impostare le dimensioni dell' immagine usate come sfondo
- background position per impostare la posizione dell'immagine nell' elemento

Con la classe **.title** e **.content** per settare lo stile di scrittura del titolo e del contenuto dei tre elementi

-JAVASCRIPT per scrivere il contenuto



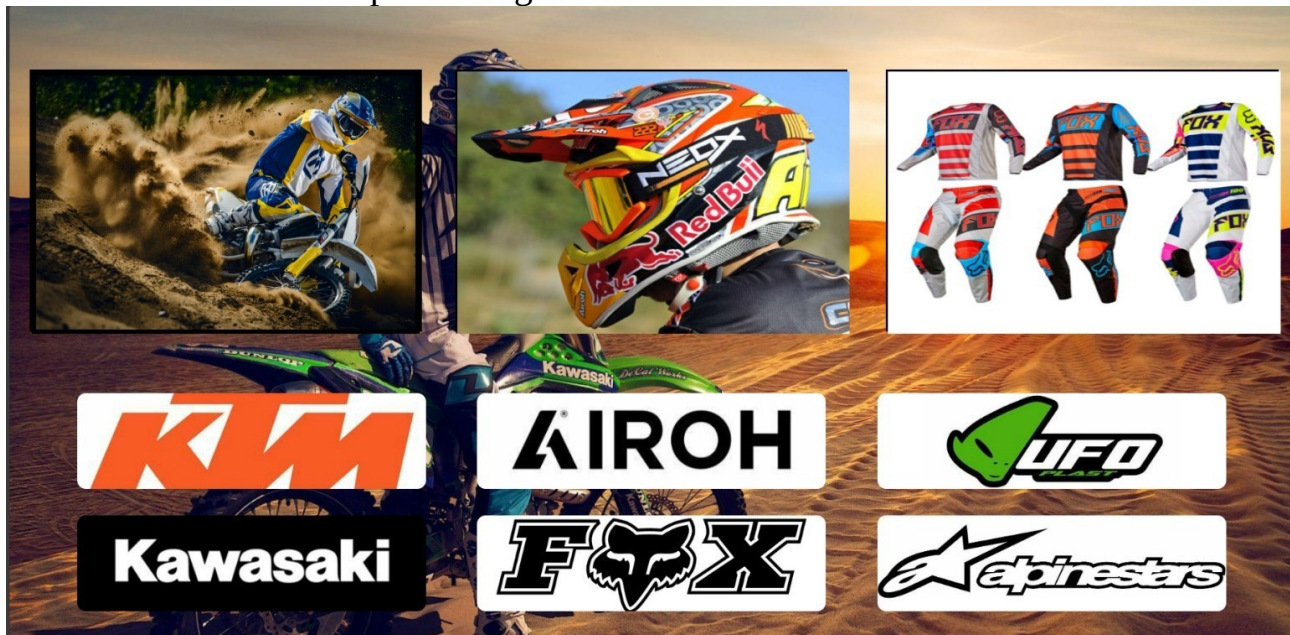
```
frontEnd > src > Accordion.jsx > Accordion > toggle
4 export default function Accordion() {
8   const toggle = (i) =>{
9     if(selected == i){
10       return setSelected(null)
11     }
12   }
13   setSelected(i)
14   return (
15     <>
16       <div className='wrapper'>
17         <div className='accordion'>
18           (data.map((item,i)=>{
19             <div className='item'>
20               <div className='title' onClick={()=> toggle(i)}>
21                 <h2>{item.question}</h2>
22               </div>
23               <div className={selected == i ? 'content show' : 'content'}>{item.answer}</div>
24             </div>
25           )))
26         </div>
27       </div>
28     </div>
29   )
30 }
31
32 const data = [
33   {
34     question: 'NEWS',
35     answer: 'Tony Cairoli shock, addio Ktm. Il futuro è in Ducati .Il pluricampione del motocross lascia la casa austriaca dopo sei titoli vinti
36   },
37   {
38     question: 'CONTACT',
39   }
40 ]
41
42
43
```



```
frontEnd > src > Accordion.jsx > Accordion > toggle
4 export default function Accordion() {
19   (data.map((item,i)=>{
26     </div>
27   )))
28   </div>
29 }
30
31 </div>
32
33 const data = [
34   {
35     question: 'NEWS',
36     answer: 'Tony Cairoli shock, addio Ktm. Il futuro è in Ducati .Il pluricampione del motocross lascia la casa austriaca dopo sei titoli vinti
37   },
38   {
39     question: 'CONTACT',
40     answer: 'Felice Francesco Florenzano Matr.332773 / Nicholas Avena Matr.332775',
41   },
42   {
43     question: 'ABOUT US',
44     answer: 'Motorcycle Shop nasce dall idea di due ragazzi con la passione delle due ruote. In questa piccola realtà cerchiamo far entrare quan
45   },
46 ]
47
48
49
50
51
52
53
```

Scorrendo in basso , l'utente trova sei pulsanti con i marchi disponibili presso il nostro e-commerce :

- KTM e KAWASAKI per le moto
- AIROH e FOX per i caschi
- UFO e ALPINESTAR per l'abbigliamento



Con CSS è stata creata una griglia con 6 elementi, in ognuno dei quali è stata inserita una foto relativa ai marchi presenti nel nostro e-commerce.

I 6 elementi sono dei pulsanti, che l'utente può cliccare.

Non appena l'utente ne preme uno, si genera un link e il server interrogherà il database con la relativa query.

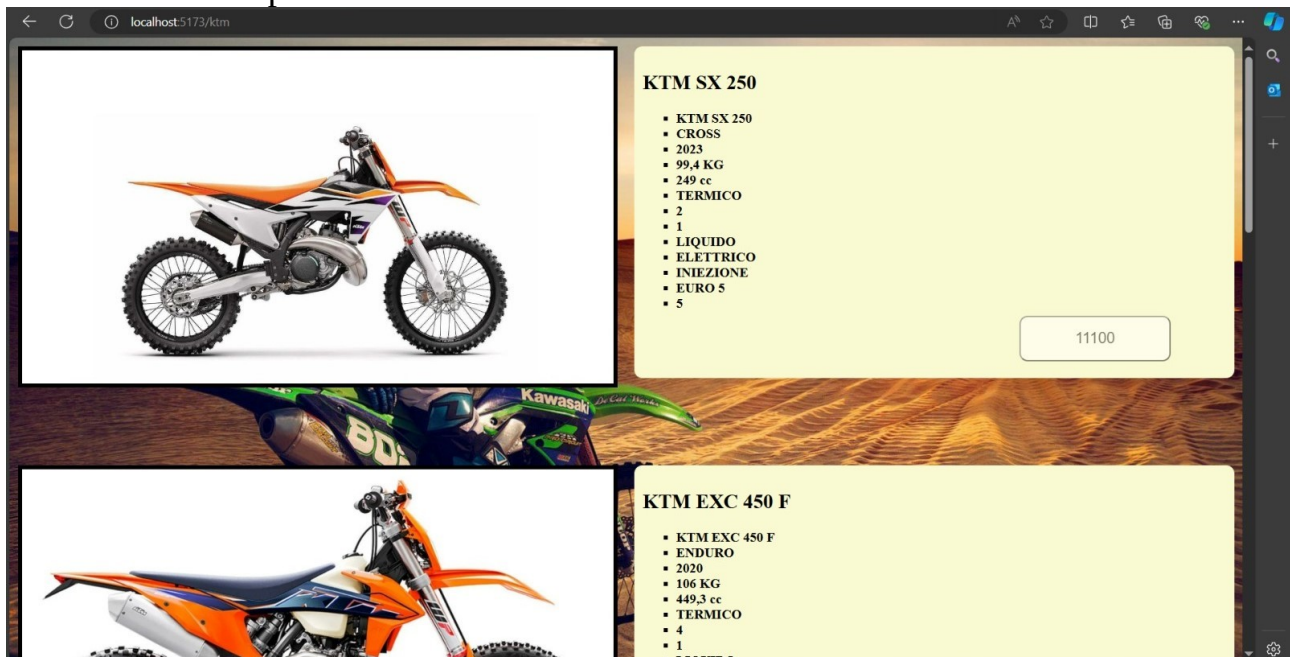
```
File Edit Selection View Go Run Terminal Help
progetto-web

EXPLORER
PROGETTO-WEB
  Backed
  > node_modules
  {} package-lock.json
  {} package.json
  JS server.js
  > frontEnd
  > node_modules
  > public
  > images
  vite.svg
  > src
  > assets
  # Accordion.css
  # Accordion.jsx
  # Airoh.jsx
  # App.jsx
  # App.css
  # App.jsx
  # Fox.jsx
  # Image.jsx
  # index.css
  # Kawa.jsx
  # Ktm.css
  # Ktm.jsx
  # main.jsx
  # Ufo.jsx
  .gitignore
  eslint.config.js
  index.html
  {} package-lock.json
  > OUTLINE
  > TIMELINE

server.js main.jsx App.jsx Ktm.jsx App.css Accordion.jsx Ktm.css Accordion.css

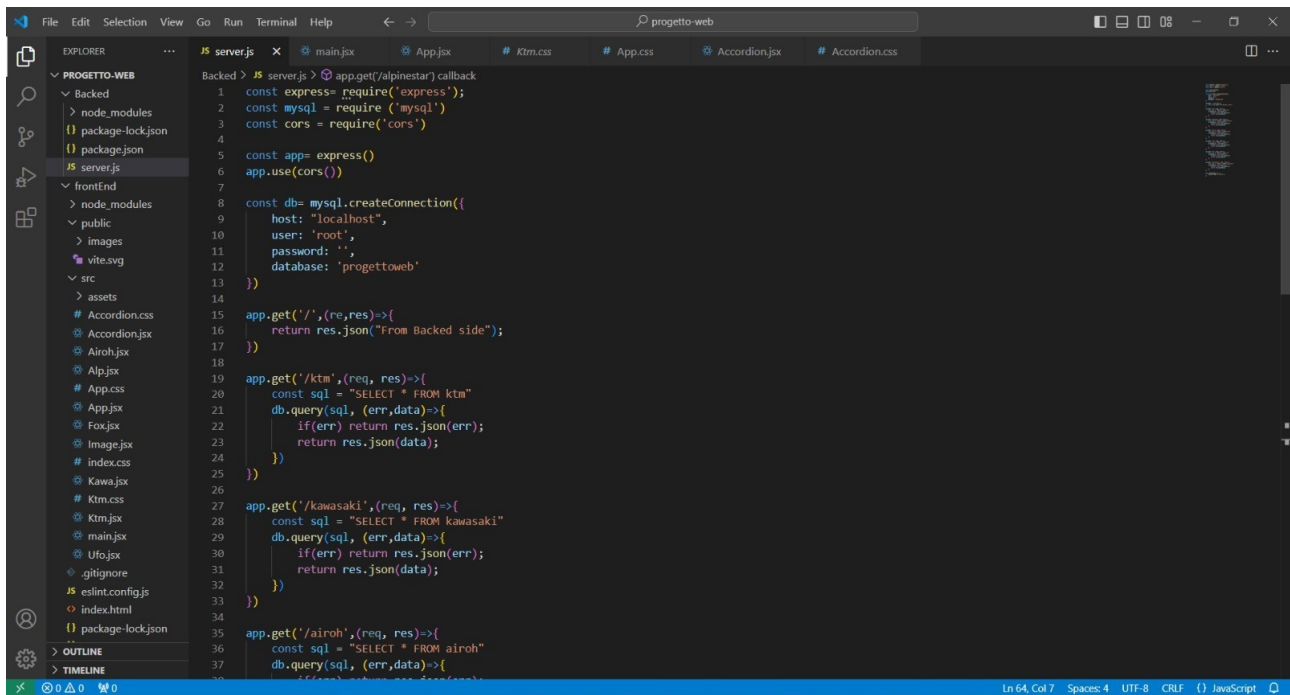
frontEnd > src > App.jsx > App
11 export default function App() {
12   ,[]
13 }
14
15 return (
16   <
17     <html lang="en">
18     <body>
19       <h1 className="Title">Motorcycle Shop</h1>
20       <Accordion></Accordion>
21       <Image></Image>
22       <div className="grid-container">
23         <div className="item1">
24           <link to={"/ktm"}><button className="buttonKTM"></button></Link>
25         </div>
26         <div className="item4">
27           <link to={"/kawasaki"}><button className="buttonKAWA"></button></Link>
28         </div>
29         <div className="item2">
30           <link to={"/airoh"}><button className="buttonAIROH"></button></Link>
31         </div>
32         <div className="item5">
33           <link to={"/fox"}><button className="buttonFOX"></button></Link>
34         </div>
35         <div className="item3">
36           <link to={"/ufo"}><button className="buttonUFO"></button></Link>
37         </div>
38         <div className="item6">
39           <link to={"/alpinestar"}><button className="buttonALP"></button></Link>
40         </div>
41       </div>
42     </body>
43     </html>
44   </>
45 )
46
47
48
49
50
51
52
53
```


Cliccando, ad esempio, sul pulsante con il logo **KTM**, si aprirà una nuova schermata con tutte le moto presenti nel nostro e-commerce.



Stessa cosa accade cliccando il pulsante con il logo **KAWASAKI**.

Questo si ottiene interrogando il database e in particolare la tabella KTM contenente tutte le moto:



```
Backed > JS server.js > app.get('/elpinestar') callback
1 const express = require('express');
2 const mysql = require('mysql');
3 const cors = require('cors');
4
5 const app = express();
6 app.use(cors());
7
8 const db = mysql.createConnection({
9   host: 'localhost',
10  user: 'root',
11  password: '',
12  database: 'progettoweb'
13 });
14
15 app.get('/', (req, res) => {
16   return res.json("from Backed side");
17 });
18
19 app.get('/ktm', (req, res) => {
20   const sql = "SELECT * FROM ktm";
21   db.query(sql, (err, data) => {
22     if (err) return res.json(err);
23     return res.json(data);
24   });
25 });
26
27 app.get('/kawasaki', (req, res) => {
28   const sql = "SELECT * FROM kawasaki";
29   db.query(sql, (err, data) => {
30     if (err) return res.json(err);
31     return res.json(data);
32   });
33 });
34
35 app.get('/airoh', (req, res) => {
36   const sql = "SELECT * FROM airoh";
37   db.query(sql, (err, data) => {
38     if (err) return res.json(err);
39     return res.json(data);
40   });
41 });
```

In particolare, ogni moto del nostro e-commerce è descritta da:

- NOME MOTO
- CATEGORIA
- ANNO DI PRODUZIONE
- PESO A SECCO
- CILINDRATA
- MOTORE
- TEMPI
- CILINDRI
- RAFFREDDAMENTO
- AVVIAMENTO
- ALIMENTAZIONE
- EMISSIONI
- NUMERO MARCE

E' presente anche un bottone contenente il prezzo dell'articolo.

Questo avviene sia per le moto **KTM** che per le moto **KAWASAKI**

Se invece l'utente avesse intenzione di acquistare dei caschi, può cliccare sui pulsanti con i loghi **AIROH** e **FOX**, per poter vedere tutti caschi presenti nel nostro e-commerce.



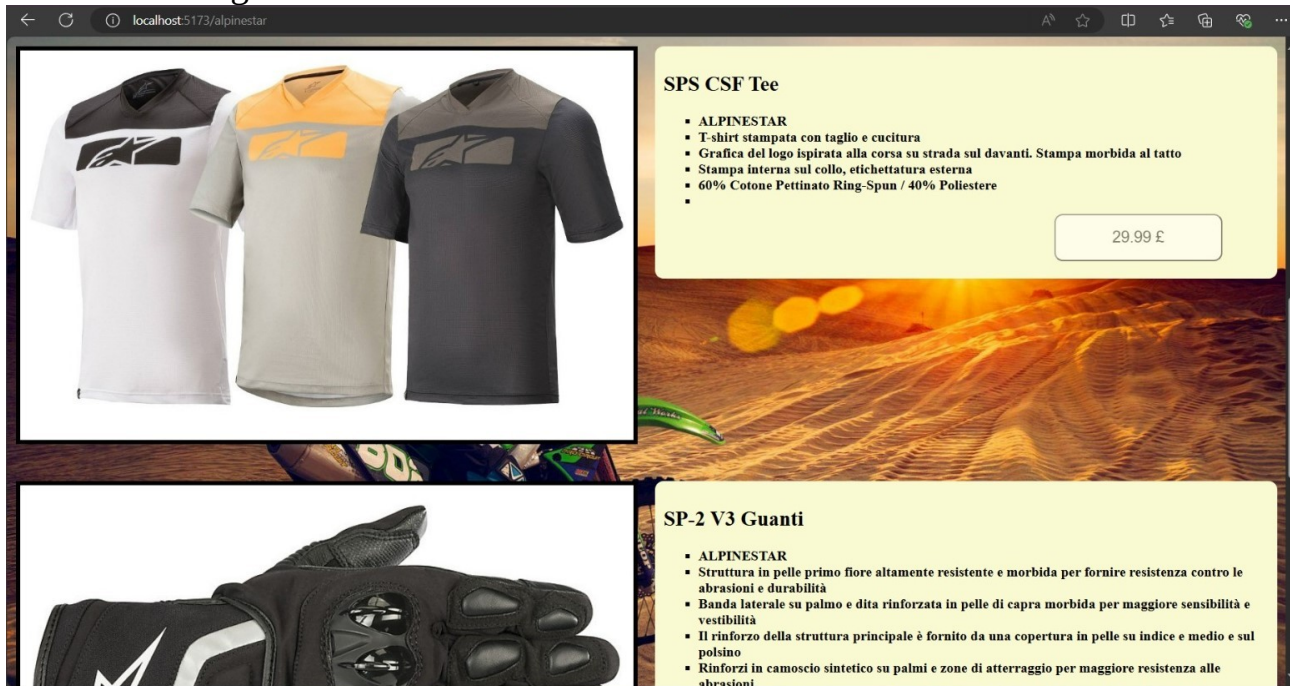
Anche in questo caso si interroga il database e in particolare le relative tabelle per visualizzare tutti i caschi.

Ogni casco è descritto attraverso 5 caratteristiche che permettono di capire come è strutturato e quali materiali sono stati utilizzati nella loro realizzazione.

Avendo nell' e-commerce sia moto da cross che moto stradali abbiamo deciso di inserire sia caschi da cross che caschi stradali.

Anche in questo caso, per ogni casco è presente un pulsante con il relativo prezzo.

Se invece l'utente volesse acquistare abbigliamento per andare in moto (giacche, guanti, pantaloni, stivali) o t-shirt e cappellini, può cliccare sui pulsanti contenenti il logo **UFO** e **ALPINESTAR**.



Anche in questo caso, ogni prodotto è descritto attraverso 5 punti che mettono in evidenza quelle che sono le caratteristiche principali.

Come tutti i prodotti presenti nel nostro e-commerce, è presente un pulsante con il relativo prezzo.



Con queste righe di codice abbiamo gestito la creazione del pulsante con il prezzo del prodotto e come si deve comportare quando viene cliccato dall'utente.

TESTING

-Test di funzionalità

Test per: tutti i collegamenti nelle pagine Web, connessione al database, moduli utilizzati per inviare o ottenere informazioni dall'utente nelle pagine Web.

Inizialmente il primo test di collegamento delle pagine Web è stato effettuato tramite JS utilizzando uno switching di pagine HTML.

Il problema riguardava l'oneroso dispendio di codice per la creazione di pagine HTML differenti e la loro gestione.

Utilizzando REACT, siamo riusciti ad ottimizzare le performance utilizzando gli strumenti che il framework offre come i componenti e la capacità di utilizzare la stessa pagine HTML per l'interazione tra esse.