Visoka škola strukovnih studija za informacione i komunikacione tehnologije

DOKUMENTACIJA

WEB PROGRAMIRANJE 2

Dinamički web sajt

https://keytech-world.netlify.app/index.html

Nikola Timotić

173/20

Smer: Internet tehnologije

Modul: Web Programiranje

Sadržaj

1 Uvod ... 1
1.1 Korišćeni programski jezici i tehonologije ... 1
1.2 Opis funkcionalnosti ... 1
1.3 Template ... 2
2 Organizacija ... 8
2.1 Šema ... 8
2.2 Mapa sajta (sitemap) ... 8
2.3 Slike stranica i opis funkcionalnosti ... 9
3 Kodovi ... 18
3.1 HTML ... 18
3.1.1 index.html ... 18
3.1.2 store.html ... 21
3.1.3 checkout.html ... 23
3.1.4 about-us.html ... 26
3.2 CSS ... 29
3.2.1 style.css ... 29
3.3 JS ... 57
3.3.1 main.js ... 57
3.4 JSON ... 90
3.4.1 brands.json ... 90
3.4.2 categories.json ... 91
3.4.3 connectionTypes.json ... 91
3.4.4 documentLinks.json ... 92
3.4.5 navLinks.json ... 93
3.4.6 products.json ... 93
3.4.7 quotes.json ... 96
3.4.8 socialLinks.json ... 97

1 Uvod

1.1 Korišćeni programski jezici i tehonologije

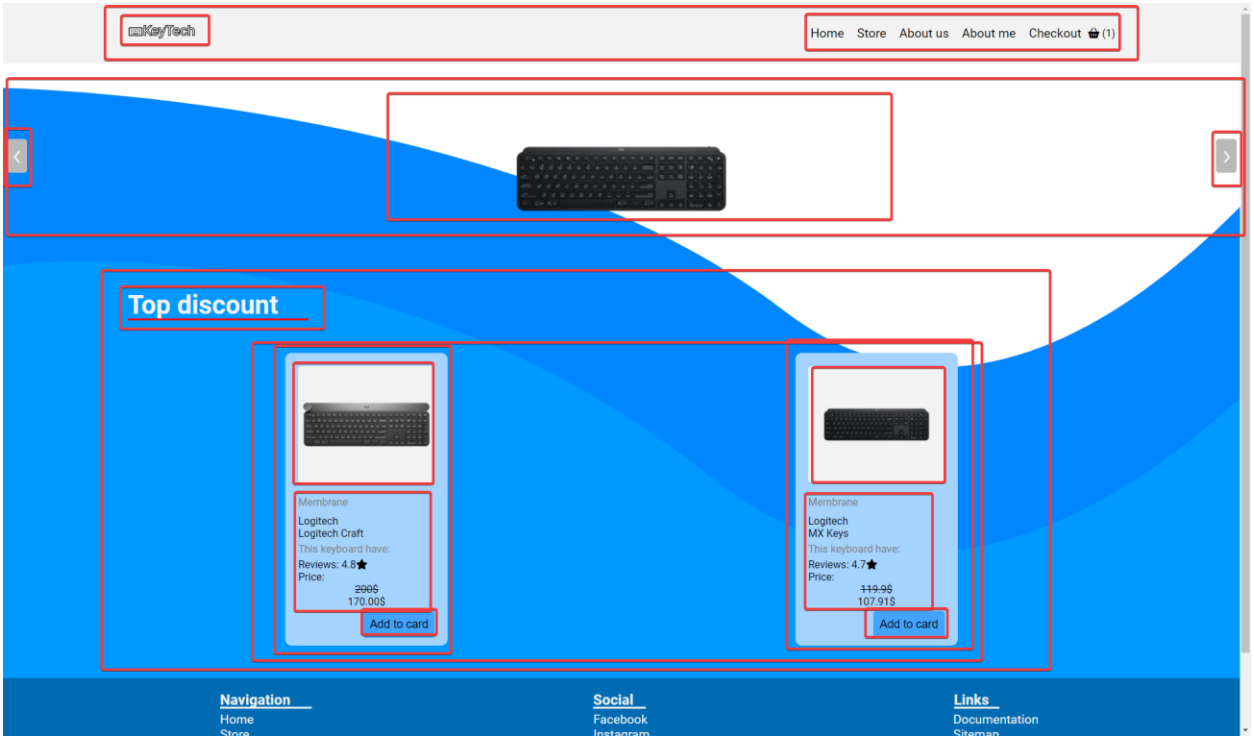
- Za izradu sajta su korišćeni: HTML, CSS, JavaScript, Ajax, XML.

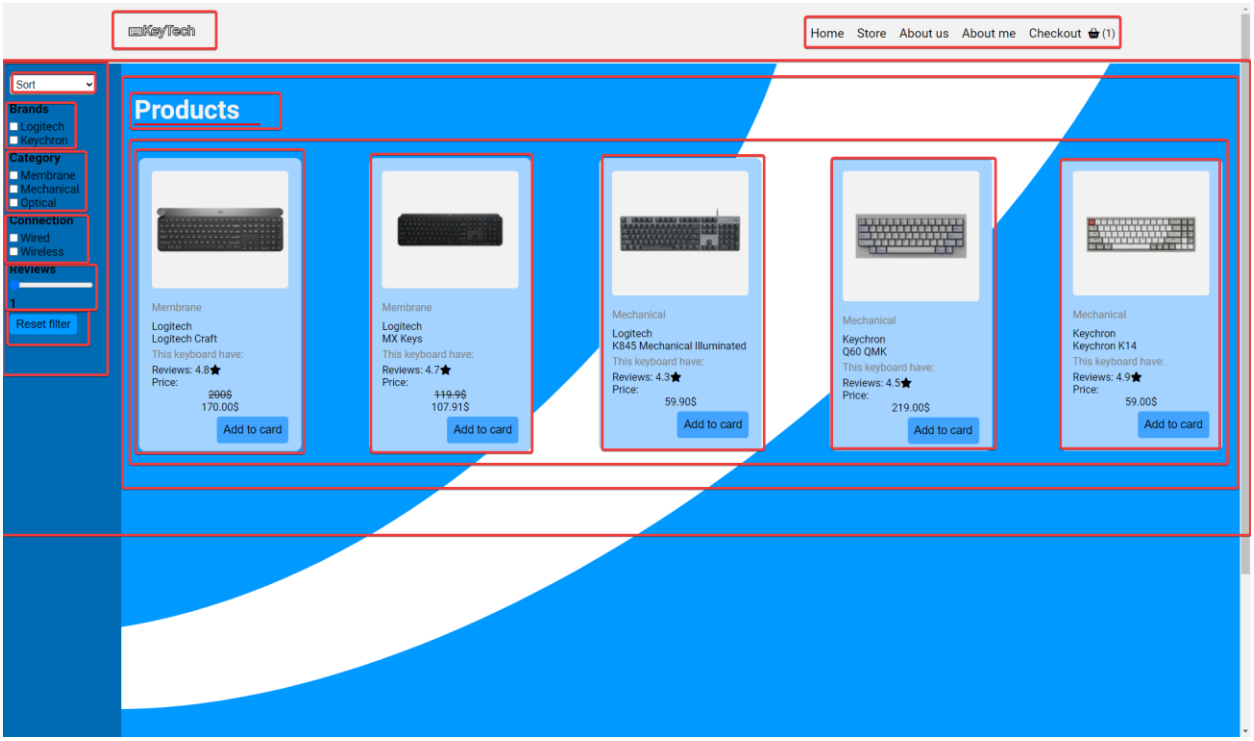
1.2 Opis funkcionalnosti

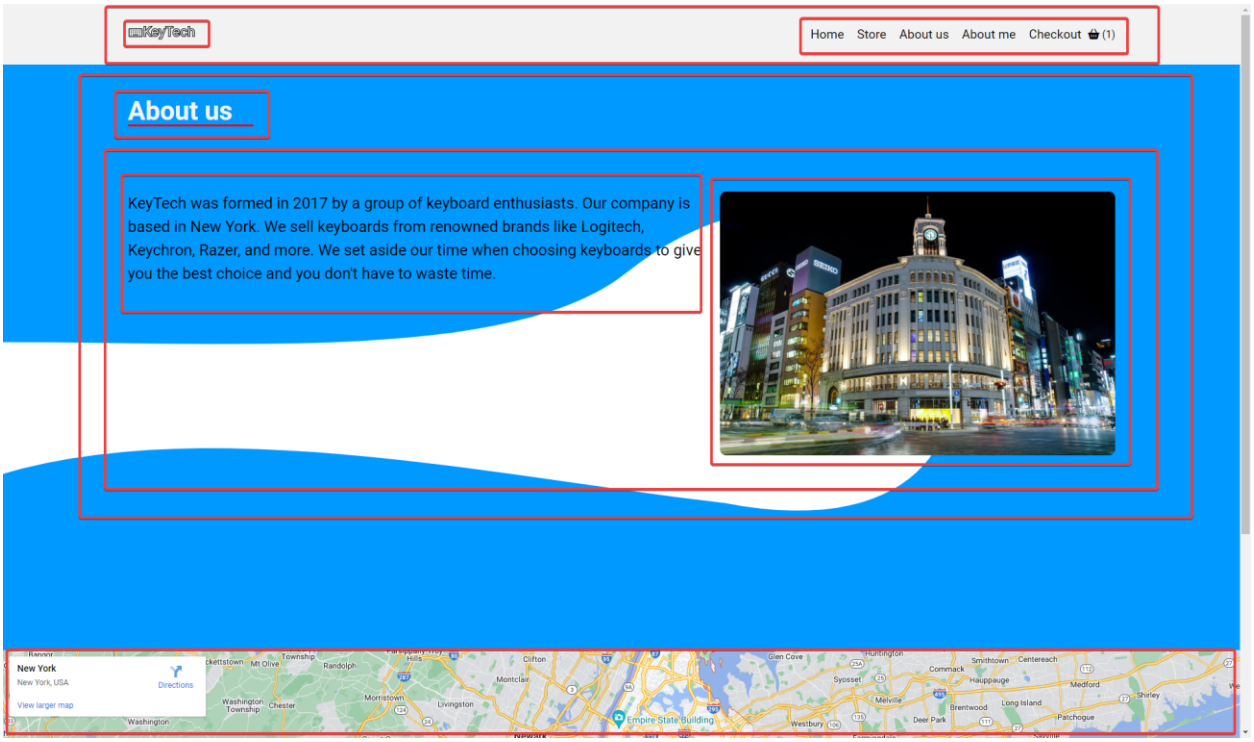
- Dinamičko kreiranje i ispisivanje menia korišćenjem Ajax zahteva;
- Dinamičko kreiranje i ispisivanje footer-a korišćenjem Ajax zahteva;
- Dinamičko kreiranje i ispisivanje proizvoda korišćenjem Ajax zahteva;
- Dinamičko kreiranje i ispisivanje kategorija proizvoda korišćenjem Ajax zahteva;
- Interaktivni slajder;
- Sinhronizovano filtriranje proizvoda po kategoriji, tipu, ocenama i načinu povezivanja;
- Sortiranja proizvoda po: imenu opadajućem, imenu rastućem, ceni opadajućem, ceni rastućem, ocenama opadajućem i ocenama rastućem redosledu.
- Dinamičko ispisivanje i izračunavanje cene na osnovu zadatog popusta.
- Dinamičko ispisivanje “Top discount” ponude.
- Interaktivna korpa koja rad sa “LocalStorage”.
- Ispisivanje broja proizvoda u korpi.
- Ispisivanje količine dodatih primeraka nekog proizvoda.
- Interaktivna promena količine odabranog proizvoda klikom na “+”, odnosno “-” dugme koje se nalazi unutar korpe.
- Brisanje pojedinačnih i svih prozvoda iz korpe.
- Korišćenje modala za ispisivanje “About me” sekcije.
- Dinamičko ispisivanje citata i autora unutar “About me” sekcije.
- Interaktivni meni za mobilne uređaje.
- Forma za kupovinu proizvoda.
- Obaveštenje prilikom dodavanja proizvoda u korpu, brisanje, kupovina i greške ukoliko je taj proizvod već u korpi, odnosno, ako je mejl već poslat.

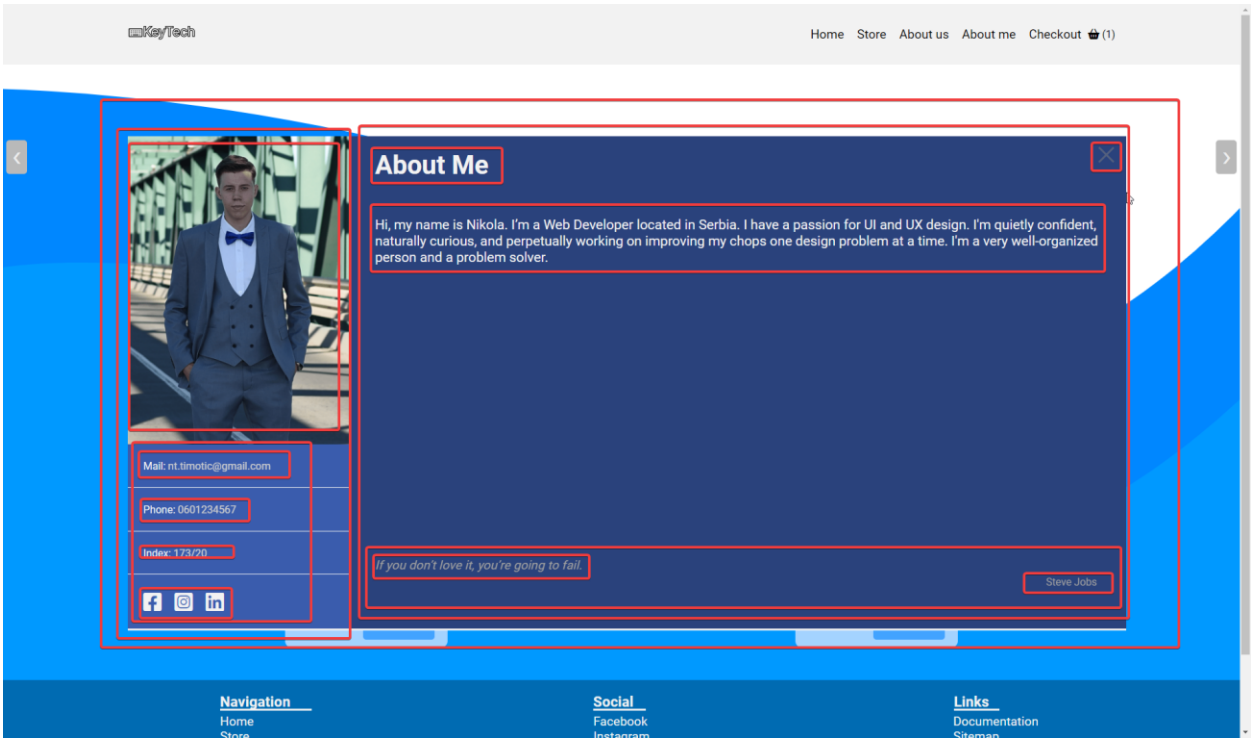
1.3 Template

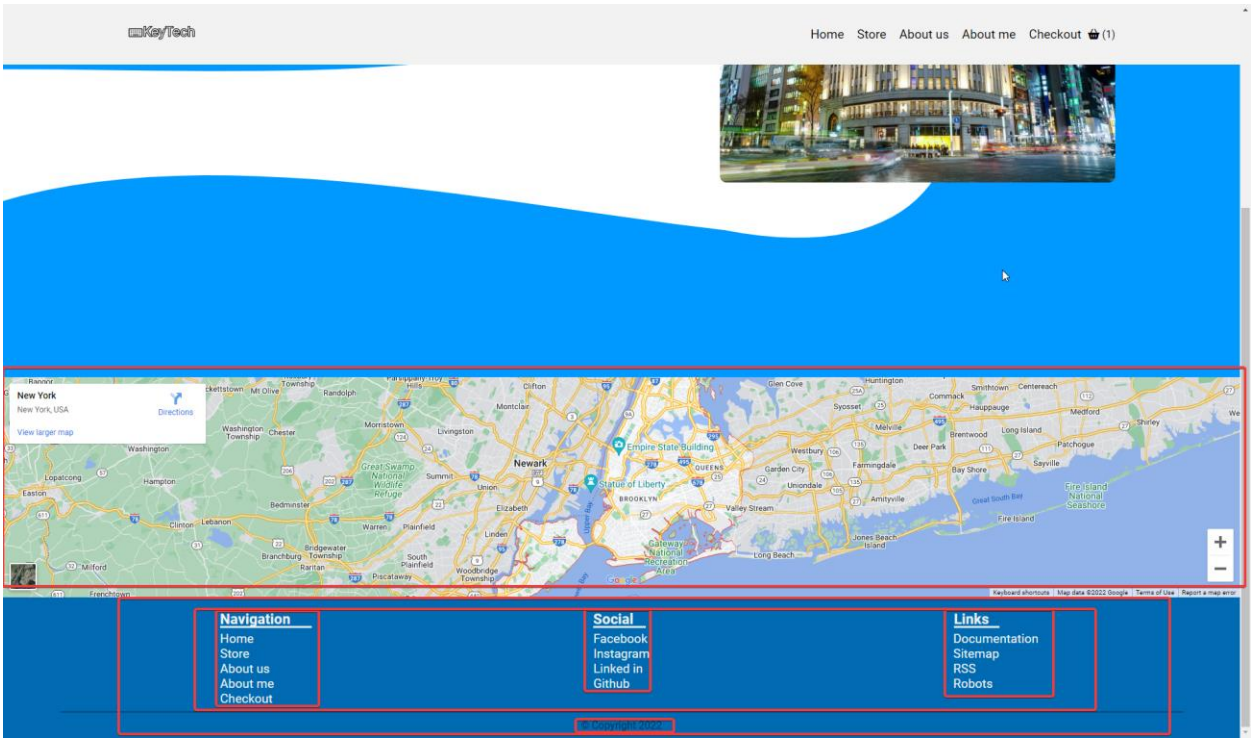
- Za izradu sajta njie korišćen templejt.

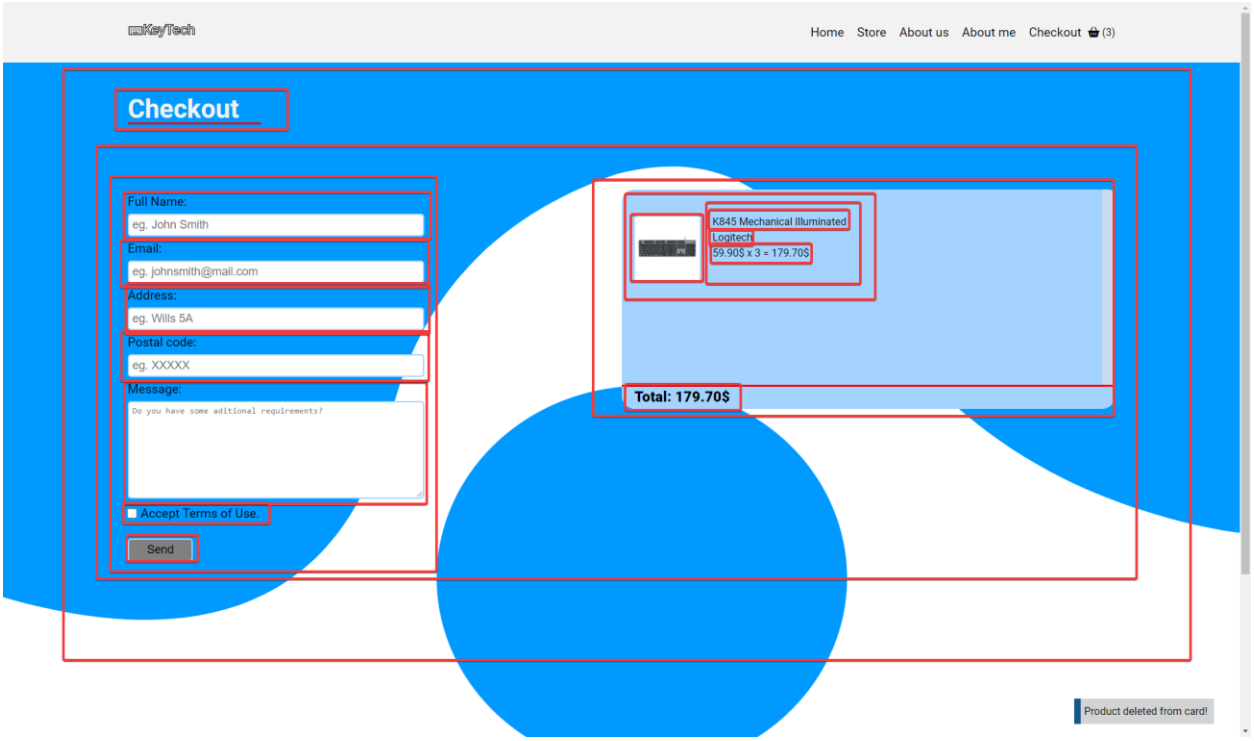


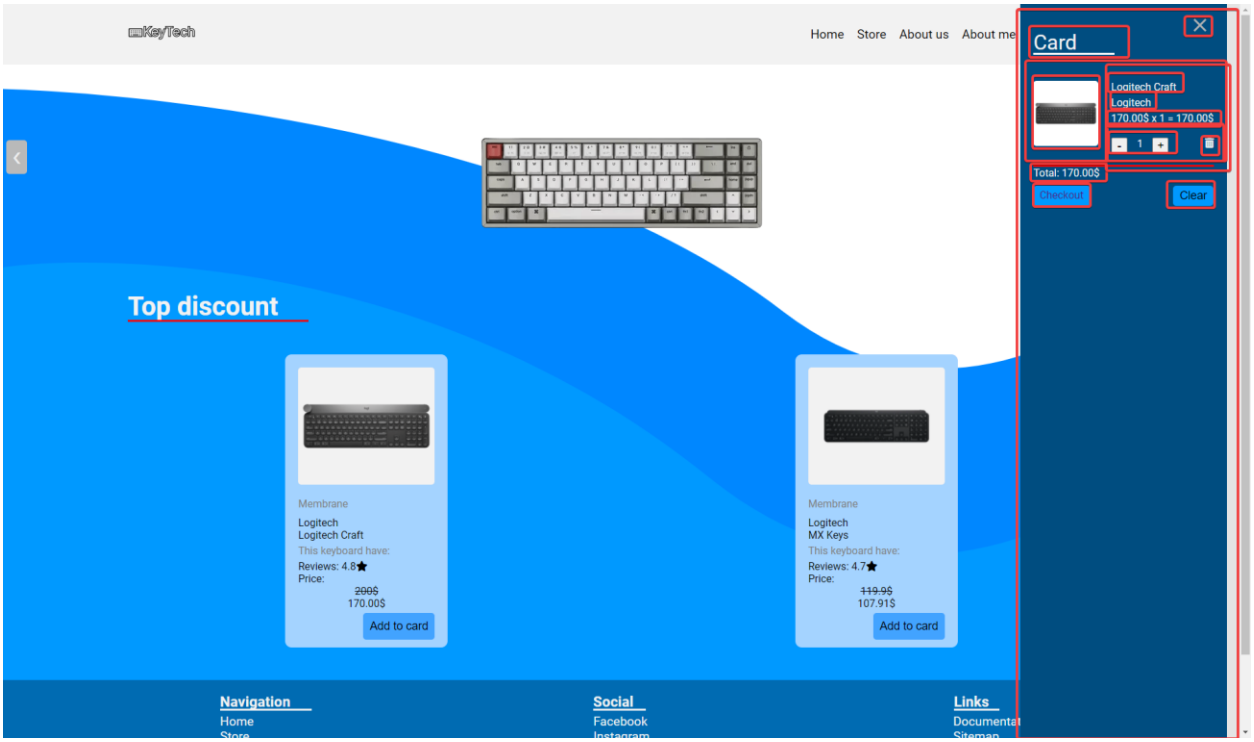



KeyTech
Home Store About us About me Checkout (1)
About us
KeyTech was formed in 2017 by a group of keyboard enthusiasts. Our company is based in New York. We sell keyboards from renowned brands like Logitech, Keychron, Razer, and more. We set aside our time when choosing keyboards to give you the best choice and you don't have to waste time.
New York
New York, USA
Directions
View larger map
Empire State Building

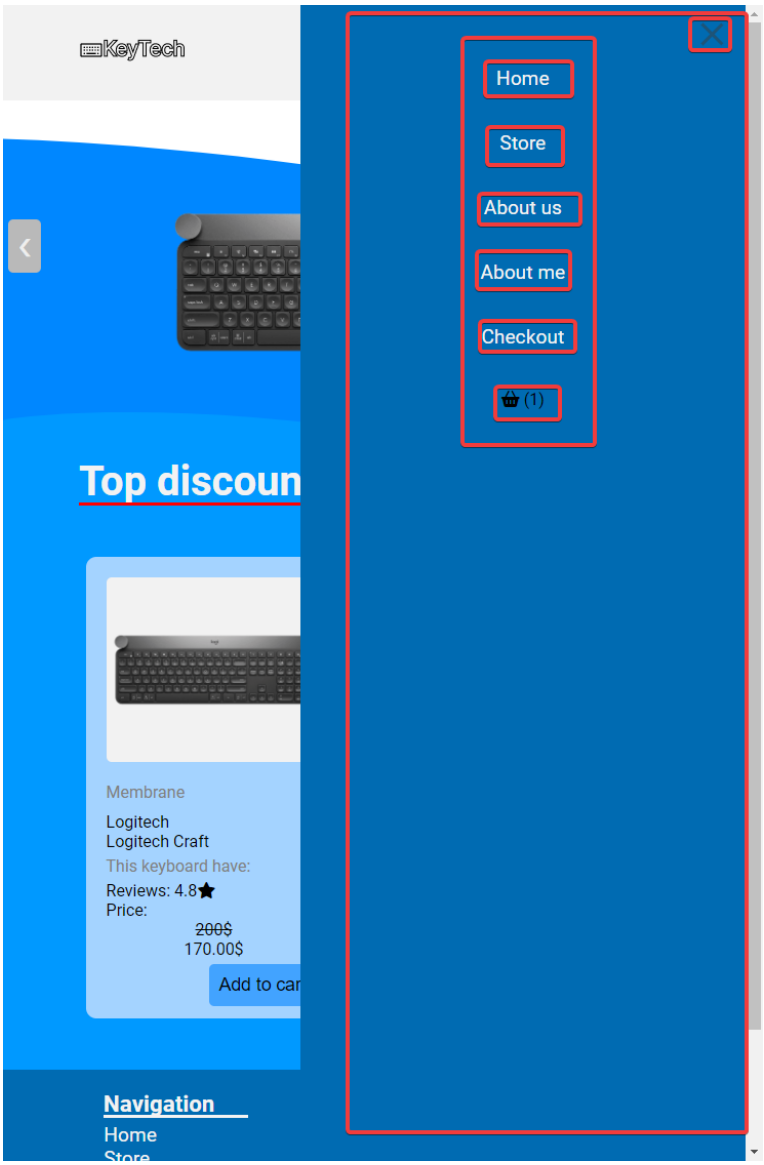

KeyTech
Home Store About us About me Checkout (1)
About Me
Hi, my name is Nikola. I'm a Web Developer located in Serbia. I have a passion for UI and UX design. I'm quietly confident, naturally curious, and perpetually working on improving my chops one design problem at a time. I'm a very well-organized person and a problem solver.
Mail: nt.timotic@gmail.com
Phone: 0601234567
Index: 173/20
If you don't love it, you're going to fail.
Steve Jobs
Navigation
Home
Social
Facebook
Links
Documentation

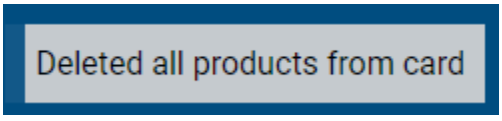

KeyTech
Home
Store
About us
About me
Checkout (1)
New York
New York, USA
Directions
View larger map
Navigation
Home
Store
About us
About me
Checkout
Social
Facebook
Instagram
Linked in
Github
Links
Documentation
Sitemap
RSS
Robots

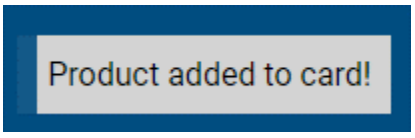

KeyTech
Home
Store
About us
About me
Checkout (3)
Checkout
Full Name:
eg. John Smith
Email:
eg. johnsmith@mail.com
Address:
eg. Wills 5A
Postal code:
eg. XXXXX
Message:
Do you have some aditional requirements?
Accept Terms of Use.
Send
K845 Mechanical Illuminated
Logitech
59.90$ x 3 = 179.70$
Total: 179.70$
Product deleted from card!

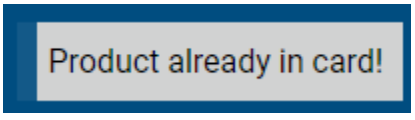

KeyTech
Home
Store
About us
About me
Card
Logitech Craft
Logitech
170.00$ x 1 = 170.00$
1
Total: 170.00$
Checkout
Clear
Top discount
Membrane
Logitech
Logitech Craft
This keyboard have:
Reviews: 4.8
Price:
200$
170.00$
Add to card
Membrane
Logitech
MX Keys
This keyboard have:
Reviews: 4.7
Price:
119.9$
107.91$
Add to card
Navigation
Home
Social
Facebook
Links

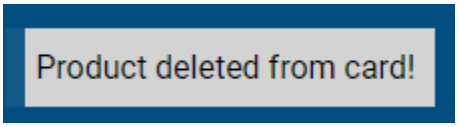
KeyTech
Top discount
Membrane
Logitech
Logitech Craft
This keyboard have:
Reviews: 4.8
Price:
200$
170.00$
Add to card
Membrane
Logitech
MX Keys
This keyboard have:
Reviews: 4.7
Price:
119.9$
107.91$
Add to card
Navigation
Home
Social
Facebook
Links
Documentation


KeyTech
Home
Store
About us
About me
Checkout
(1)
Top discoun
Membrane
Logitech
Logitech Craft
This keyboard have:
Reviews: 4.8
Price:
200$
170.00$
Add to car
Navigation
Home


Deleted all products from card

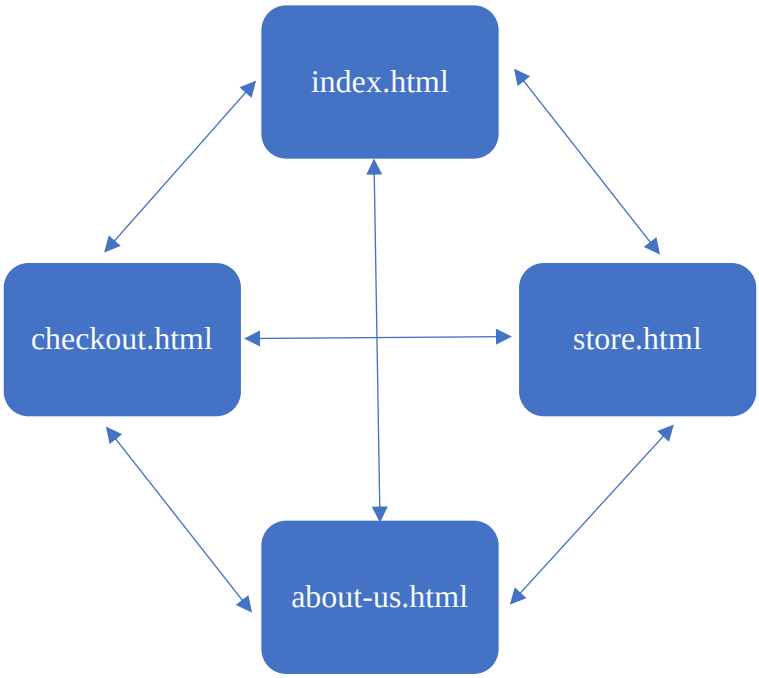

Product added to card!


Product already in card!


Product deleted from card!

2 Organizacija

2.1 Šema



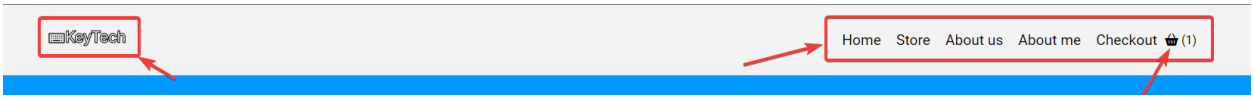
2.2 Mapa sajta (sitemap)

```
<urlset
xmlns="http://www.sitemaps.org/schemas/sitemap/0.9">
  <url>
    <loc>https://keytech-
world.netlify.app/index.html</loc>
    <lastmod>2022-04-15T08:00:00+00:00</lastmod>
    <changefreq>daily</changefreq>
    <priority>0.95</priority>
  </url>
  <url>
    <loc>https://keytech-
world.netlify.app/store.html</loc>
    <lastmod>2022-04-15T08:00:00+00:00</lastmod>
    <changefreq>daily</changefreq>
    <priority>0.90</priority>
```

```
  </url>
  <url>
    <loc>https://keytech-
world.netlify.app/checkout.html</loc>
    <lastmod>2022-04-15T08:00:00+00:00</lastmod>
    <changefreq>weekly</changefreq>
    <priority>0.85</priority>
  </url>
  <url>
    <loc>https://keytech-world.netlify.app/about-
us.html</loc>
    <lastmod>2022-04-15T08:00:00+00:00</lastmod>
    <changefreq>monthly</changefreq>
    <priority>0.70</priority>
  </url>
</urlset>
```

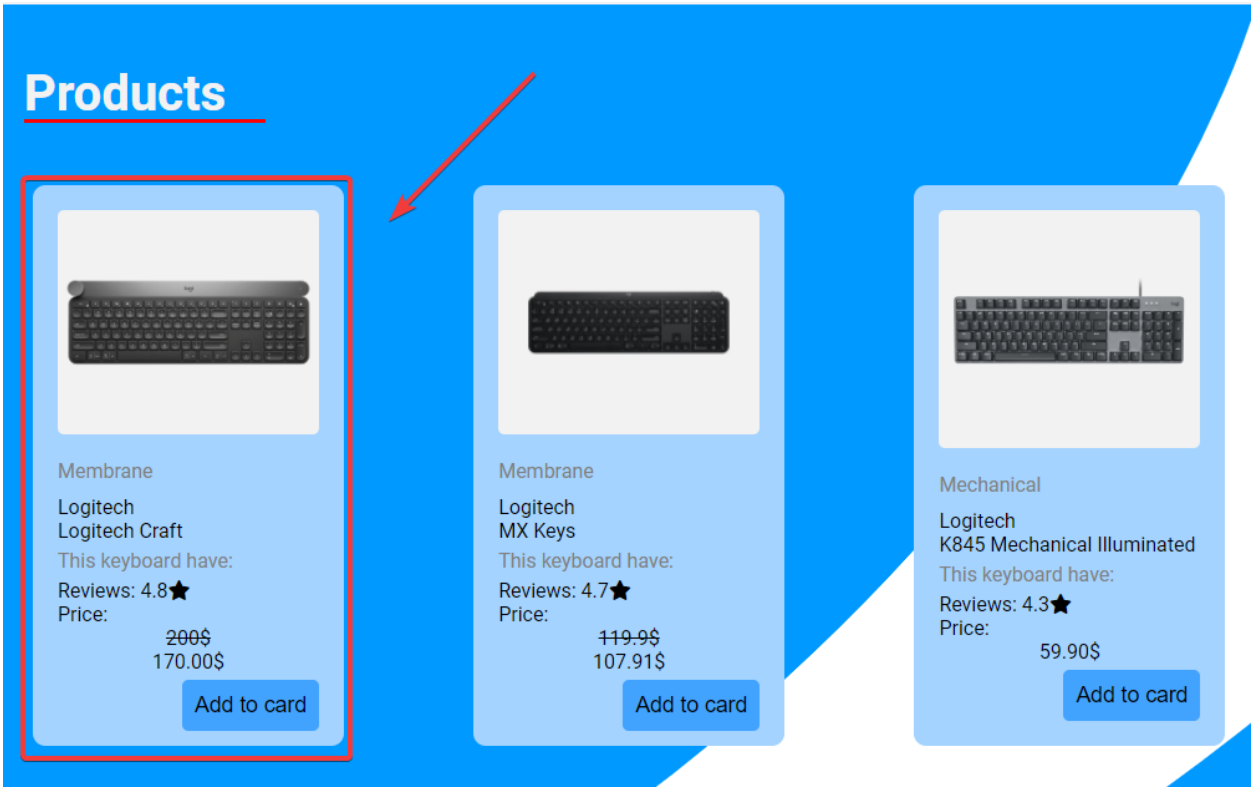
2.3 Slike stranica i opis funkcionalnosti

2.3.1 Dinamičko kreiranje i ispisivanje menia i footer-a linkova



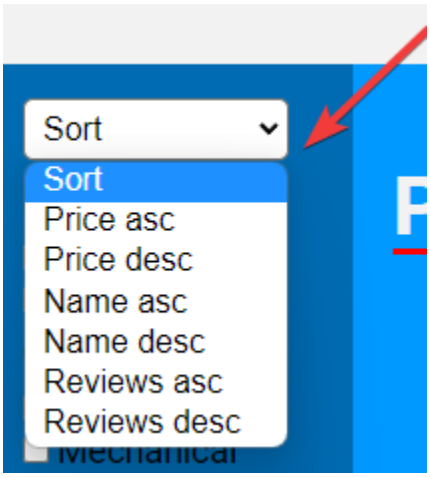
- Za kreiranje navigacionog meni-a, footer-a korišćena je “callback” funkcija “createNavBar”, koja predstavlja skup dodatnih funkcija kojima se prosledjuje povratni podatak iz “fetchData()” funckije.

2.3.2 Dinamičko kreiranje i ispisivanje proizvoda



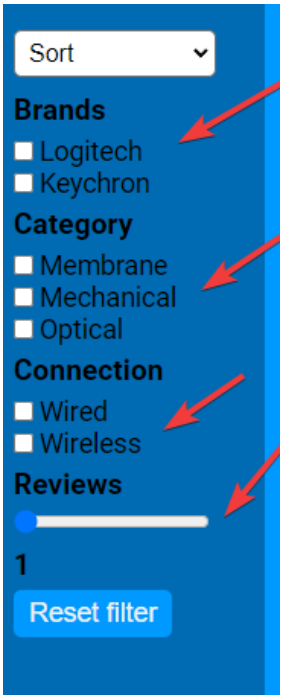
- Za dinamičko ispisivanje i kreiranje proizvoda, korišćena je funkcija “callback” funkcija “showProducts()”, koja se poziva preko funkcija “fetchData()”.
- Kao ulazni argument, prihvata niz objekata proizvoda.
- Navedena funckija je korišćena i prilikom filtriranja proizvoda.

2.3.3 Soritranje proizvoda



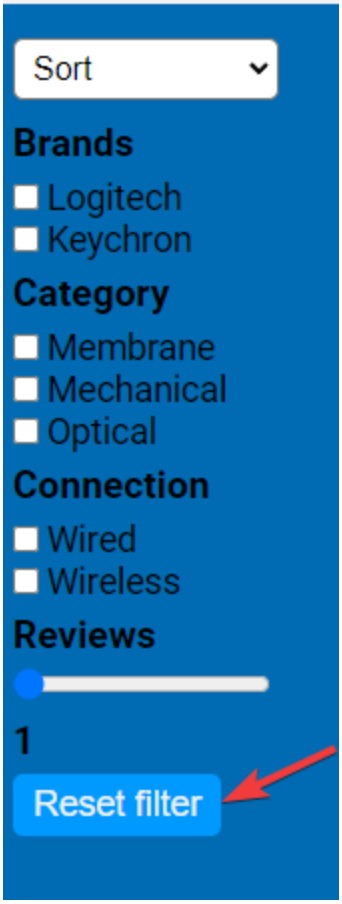
- Za sortiranje proizvoda, korišćena je funkcija “sortProducts()”, koja kao ulazni parametar ima selektovan objekat padajuće liste.
- Nakon sortiranja, poziva se funkcija “showProducts()”, koja kao ulazni parametar poseduje sortiranu listu porizvoda.
- Sortiranje je moguće po više kriterijuma:
    - Ceni rastuće;
    - Ceni opadajuće;
    - Nazivu rastućem;
    - Nazivu opadajućem,
    - Oceni rastućem;
    - Oceni opadajućem;

2.3.4 Filtriranje proizvoda po više kriterijuma



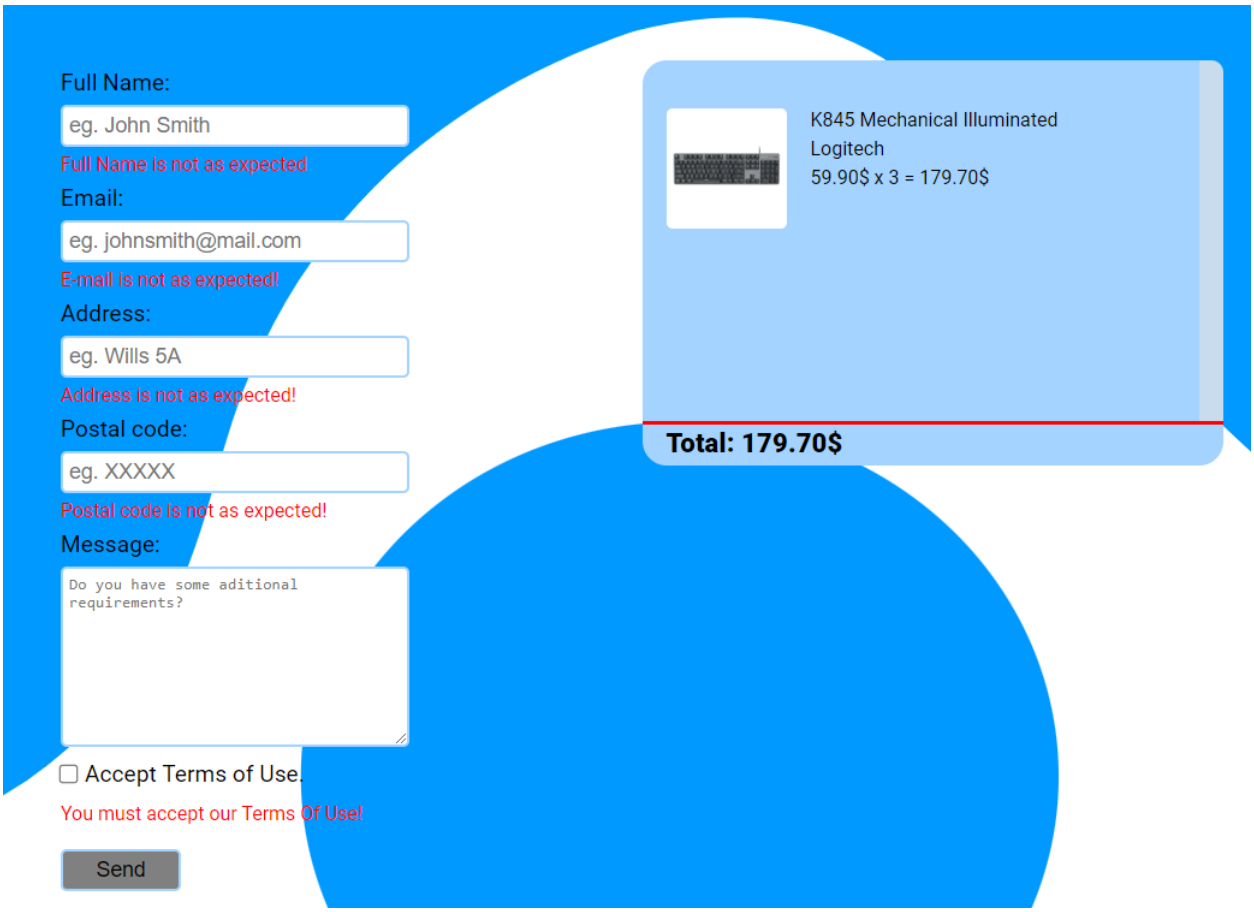
- Prilikom kreiranja liste filtera, svakom elementu filtera je dodat ".addEventListener()", koja prilikom interakcija sa elementima poziva funkciju "filterProducts()", koja kreira kopiju proizvoda LocalStorage i prosleđuje je kao ulazni argument glavnoj funkciji "showProducts()".
- Ona dalje poziva pod funkcije ("brandsFilter", "categoryFilter", "connectionType", "reviewsFilter"), koje kao ulazni parametar imaju listu elemenata, koja već može biti filtrirana zavisno od redosleda izvršavanja.

2.3.5 Resetovanje (poništavanje) filtera



- Pritiskom na taster “Reset filter”, dolazi do pozivanja funkcije “resetFilters()”, koja poništava sve filtere i sortiranje;

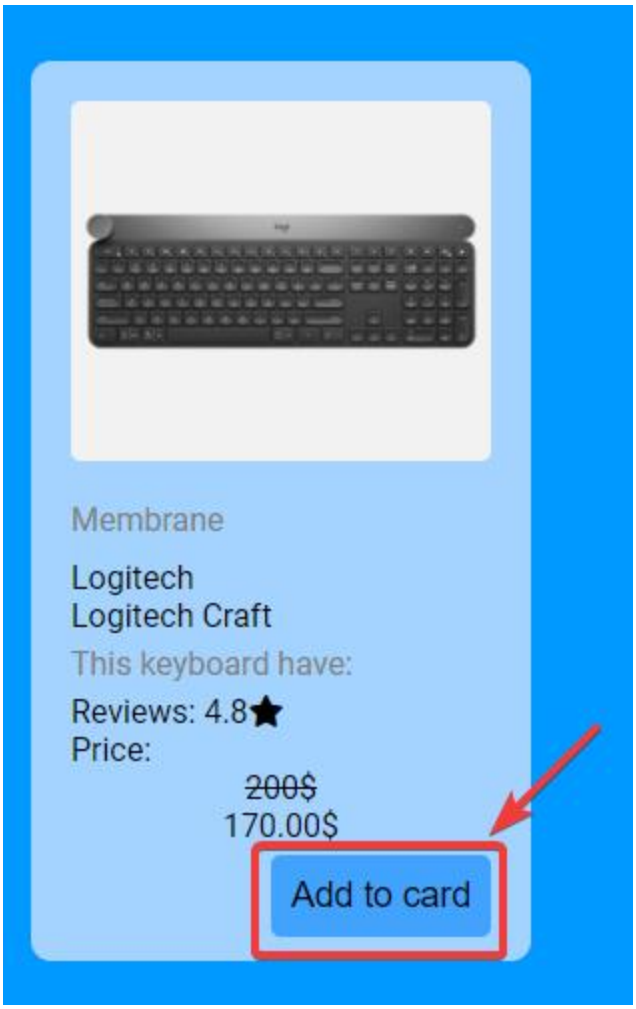
2.3.6 Checkout forma za naručivanje



- Na slici je prikazana forma koja sadrži validaciju unetih podataka.
- Ako su uneti neispravni podaci, ispod polja za popunjavanje, ispisaće nam se poruka koji podaci nisu ispravno napisani.
- “Full name” mora da sadrži najmanje jedno ime i prezime, koji počinju jednim velikim slovom, i ostalim malim uz mogućnost korišćenja “ČĆŽĐŠ”.
- “Email” prihvata e-mail adrese koje moraju početi slovom ili brojem, a mogu sadržati slova, brojeve, “.”, “-”, “_”. Moraju sadržati simbol “@”, nakon koga idu slova i brojevi, zatim “.”, nakon toga sledi tekst od 2 do 4 slova bez brojeva ili drugih simbola.
- “Address” – Obavezno polje u koje se unosi adresa, Može sadržati više reči u kombinaciji karaktera i brojeva kuća.
- “Postal code“ – Obavezno polje za unos poštanskog broja koje je u opsegu od 10000 – 99999.
- “Message” polje prihvata opcionu poruku koja može biti prazna (bez poruke), ili ukoliko korisnik odluči da napiše poruku, ista mora sadržati više od 20 karaktera, odnosno manje od 450.
- “Accept Terms of Use.” je obavezno polje.
- Nije moguće izvršiti kupovinu ukoliko je neki od podataka prazan, pogrešno unet ili ako nemate odabranih proizvoda u korpi.

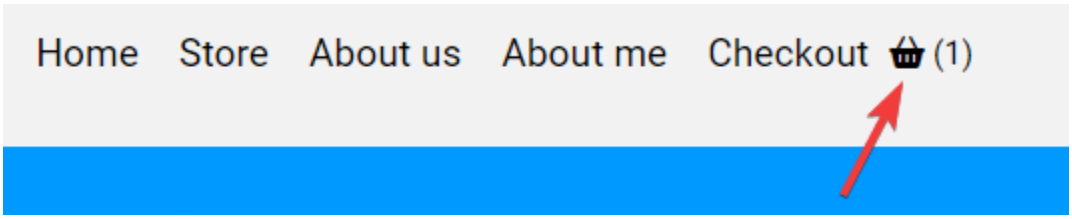
- Sa desne strane se nalazi lista, koja se dinamički osvežava, koja prikazuje odabrane proizvode iz korpe, njihove cene, količinu, i ukupnu cenu.

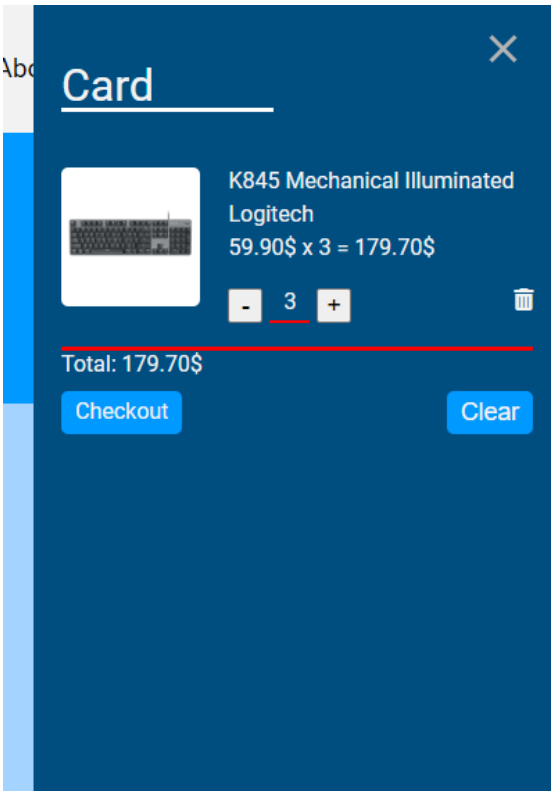
2.3.7 Dodavanje proizvoda u korpu



- Klikom na dugme “Add to card”, navedeni proizvod dodajemo u korpu.
- Pomoću funkcije “eventListener()”, koja kao ulazne parametre poseduje “ime id/klase proizvoda”, “class/id”, “tip interakcije eventlistener-a”, “ime callback funkcije (“addToCard” u ovom slučaju)”.
- Funkcija “addToCard()” prvo proverava da li se navedeni proizvod već nalazi u korpi, ukoliko je to slučaj, ispisuje se poruka u desnom uglu koja nas time obaveštava, u suprotnom, proizvod se dodaje na postojeću listu u “localStorage”, ili je kreira ukoliko već nije.
- U donjem desnom uglu se ispisuje poruka da je proizvod dodat u korpu (“LocalStorage”).

2.3.8 Interaktivna korpa

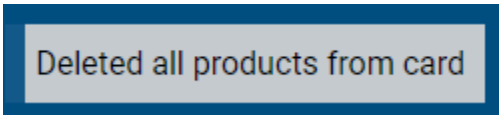


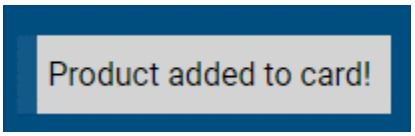


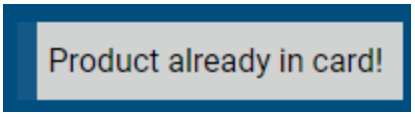
- Pored korpe, nalazi se broj proizvoda koji je odabran za kupovinu.
- Klikom na dugme prikazano na slici, dolazi do otvaranja menia korpe sa desne strane.
- U korpi se dinamički ispisuju proizvodi koje dodajemo u korpu.
- Neki od podataka su:
    - Naziv proizvoda (model)
    - Naziv brenda
    - Cena jednog komada
    - Odabrana količina,
    - Ukupna cena,
    - Ukupna cena svih proizvoda.
- Pored navedenih podataka, unutar korpe se nalatzi funkcionalnost povećavanja i smanjenja broja odabranih komada.
- Pritiskom na korpu, bišete navedeni proizvod iz korpe.

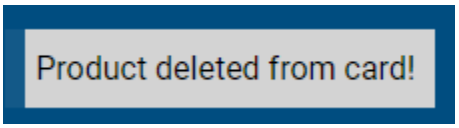
- Pritiskom na dugme “Clear”, brošete sve proizvode iz korpe.
- Klikom na “Checkout”, odlazite na stranicu za poručivanje.
- Svi proizvodi se nalaze u “LocalStorage“.

2.3.9 Obaveštenja (poruke)









- Navedene poruke se prikazuju u donjem, desnom uglu prilikom izvršavanja određenih operacija kao što su:
    - Dodavanje proizvoda u korpu,
    - Checkout forme,
    - Brisanje proizvoda iz korpe.
- Poruke se generišu pomoću funkcije “cardMessages()”, a kao ulazni parametar tekst poruke koji se ispisuje.

2.3.10 Ispisivanje citata i autora u sekciji “About me”

- Ispisivanje se vrši nakon određenog intervala.
- Svi citati i imena autora nalaze se unutar objekta u JSON fajlu (quotes.json).
- Za generisanje i ispisivanje nasumičnih citata i njihovih autora se koristi funkcija “quoteGenerator()”, koja se uz pomoć JS funkicje “setInterval()”, izvršava na svake 2 sekude.

- Na slici je naznačeno dugme pomoću koga se zatvara modal sekcije “About me”.

3 Kodovi

3.1 HTML

3.1.1 index.html

```
<!DOCTYPE html>
<html lang="en">
  <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" />
    <link rel="stylesheet" type="text/css"
href="assets/css/style.min.css" />
    <link rel="icon" type="image/x-icon"
href="assets/img/favicon.ico" />
    <meta name="keywords" content="Logitech, KeyTech,
mx, keys, craft, top, discount" />
    <meta name="description" content="KeyTech - World of
keyboard. Don't wait, buy your Logitech, Razer,
Keychron, or any from our offer." />
    <title>KeyTech | World of keyboards</title>
  </head>
  <body>
    <nav>
      <div id="nav-warpper" class="wrapper">
        <a href="index.html"><img
src="assets/img/logo.png" id="logo" alt="logo" /></a>
        <div id="nav-close-container">
          <div id="nav-close" class="close"></div>
        </div>
      </div>
    </nav>
    <header id="landing-page">
      <div class="slideshow-container">
        <div class="slide">
          <img src="assets/img/logitech-craft.png"
alt="logitech craft keyboard" />
        </div>
        <div class="slide">
          <img src="assets/img/keychron-K14.png"
alt="keychron k14 keyboard" />
        </div>
        <div class="slide">
          <img src="assets/img/Keychron-Q60-QMK.png"
alt="Keychron q60 keyboard" />
```

```
        </div>
        <div class="slide">
          <img src="assets/img/logitech-mx-keys.png"
alt="logitech mx keyboard" />
        </div>
        <a id="neg" class="prev">&#10094;</a>
        <a id="pos" class="next">&#10095;</a>
      </div>
      <div class="wrapper">
        <h1>Top discount</h1>
        <section id="products"></section>
      </div>
    </header>
    <section class="shopping-card-container">
      <div id="card-close"></div>
      <p id="card-heading">Card</p>
      <div id="card-products"></div>
      <div>
        <p id="card-total-price"></p>
      </div>
      <div id="card-btns">
        <a href="checkout.html" class="btn-
sec">Checkout</a>
        <input type="button" class="btn-sec" id="delete-
all-products" value="Clear" />
      </div>
    </section>
    <footer id="footer"></footer>
    <script
src="https://kit.fontawesome.com/92e251ce56.js"
crossorigin="anonymous"></script>
    <script src="assets/js/main.min.js"></script>
  </body>
</html>
```

3.1.2 store.html

```
<!DOCTYPE html>
<html lang="en">
  <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" />
    <link rel="stylesheet" type="text/css"
href="assets/css/style.min.css" />
    <link rel="icon" type="image/x-icon"
href="assets/img/favicon.ico" />
    <meta name="keywords" content="keytech, reviews,
logitech, mx, craft, keychron, products" />
    <meta name="description" content="KeyTech - World of
the keyboard. Don't wait, there are plenty of keyboards
waiting for you to choose one of many from our offer."
/>
    <title>KeyTech | Store</title>
  </head>
  <body>
    <nav>
      <div id="nav-warpper" class="wrapper">
        <a href="index.html"><img
src="assets/img/logo.png" id="logo" alt="logo" /></a>
        <div id="nav-close-container">
          <div id="nav-close" class="close"></div>
        </div>
      </div>
    </nav>
    <header id="store">
      <div id="store-container">
        <div id="right-products-container">
```

```
          <h1>Products</h1>
          <div id="products"></div>
        </div>
        <div id="side-filter">
          <div id="sort-products">
            <select name="sortProduct" id="sort-product-
placeholder">
              <option value="0">Sort</option>
              <option value="1">Price asc</option>
              <option value="2">Price desc</option>
              <option value="3">Name asc</option>
              <option value="4">Name desc</option>
              <option value="5">Reviews asc</option>
              <option value="6">Reviews desc</option>
            </select>
          </div>
          <div id="brands-filter"></div>
          <div id="category-filter"></div>
          <div id="connection-type-filter"></div>
          <div id="reviews-filter"></div>
          <input type="button" class="btn-sec"
id="reset-filter" value="Reset filter" />
        </div>
      </div>
    </header>
    <section class="shopping-card-container">
      <div id="card-close"></div>
      <p id="card-heading">Card</p>
      <div id="card-products"></div>
      <div>
        <p id="card-total-price"></p>
      </div>
      <div id="card-btns">
        <a href="checkout.html" class="btn-
sec">Checkout</a>
```

```
        <input type="button" class="btn-sec" id="delete-
all-products" value="Clear" />
      </div>
    </section>
    <footer id="footer"></footer>
    <script
src="https://kit.fontawesome.com/92e251ce56.js"
crossorigin="anonymous"></script>
    <script src="assets/js/main.min.js"></script>
  </body>
</html>
```

3.1.3 checkout.html

```
<!DOCTYPE html>
<html lang="en">
  <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" />
    <link rel="stylesheet" type="text/css"
href="assets/css/style.min.css" />
    <link rel="icon" type="image/x-icon"
href="assets/img/favicon.ico" />
    <meta name="keywords" content="keytech, checkout,
logitech, razer, mx" />
    <meta name="description" content="This is checkout
page for our company. Buy a new keyboard." />
    <title>KeyTech | Checkout</title>
  </head>
  <body>
```

```
    <nav>
      <div id="nav-warpper" class="wrapper">
        <a href="index.html"><img
src="assets/img/logo.png" id="logo" alt="logo" /></a>
        <div id="nav-close-container">
          <div id="nav-close" class="close"></div>
        </div>
      </div>
    </nav>
    <header id="checkout">
      <div class="wrapper">
        <h1>Checkout</h1>
        <div id="checkout-container">
          <div id="checkout-left">
            <form>
              <div id="form-top">
                <div class="label-container">
                  <label for="fullName">Full
Name:</label>
                  <input type="text" name="fullName"
id="fullName" placeholder="eg. John Smith" />
                  <p></p>
                </div>
                <div class="label-container">
                  <label for="email">Email:</label>
                  <input type="text" name="email"
id="email" placeholder="eg. johnsmith@mail.com" />
                  <p></p>
                </div>
              </div>
              <div id="form-address">
                <div class="label-container">
                  <label for="address">Address:</label>
                  <input type="text" name="address"
id="address" placeholder="eg. Wills 5A" />
                  <p></p>
```

```
                </div>
                <div class="label-container">
                  <label for="postal-code">Postal
code:</label>
                  <input type="number" id="postal-code"
name="postal-code" min="10000" max="99999"
placeholder="eg. XXXXX" />
                  <p></p>
                </div>
              </div>
              <div id="form-textarea">
                <div class="label-container">
                  <label for="message">Message:</label>
                  <textarea name="message" id="message"
placeholder="Do you have some aditional
requirements?"></textarea>
                  <p></p>
                </div>
              </div>
              <div id="form-bottom">
                <div class="label-container">
                  <input type="checkbox" id="accept-tos"
name="accept-tos" />
                  <label for="accept-tos">Accept Terms
of Use.</label>
                  <p></p>
                </div>
                <div class="button-container">
                  <input class="btn-primary"
type="button" value="Send" id="contact-submit" />
                  <p></p>
                </div>
              </div>
            </form>
          </div>
          <div id="checkout-right">
```

```
            <div id="checkout-product-list"></div>
            <p id="checkout-total"></p>
          </div>
        </div>
      </div>
    </header>
    <section class="shopping-card-container">
      <div id="card-close"></div>
      <p id="card-heading">Card</p>
      <div id="card-products"></div>
      <div>
        <p id="card-total-price"></p>
      </div>
      <div id="card-btns">
        <a href="checkout.html" class="btn-
sec">Checkout</a>
        <input type="button" class="btn-sec" id="delete-
all-products" value="Clear" />
      </div>
    </section>
    <footer id="footer"></footer>
    <script
src="https://kit.fontawesome.com/92e251ce56.js"
crossorigin="anonymous"></script>
    <script src="assets/js/main.min.js"></script>
  </body>
</html>
```

3.1.4 about-us.html

```
<!DOCTYPE html>
<html lang="en">
  <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" />
    <link rel="stylesheet" type="text/css"
href="assets/css/style.min.css" />
    <link rel="icon" type="image/x-icon"
href="assets/img/favicon.ico" />
    <meta name="keywords" content="KeyTech, keyboard,
enthusiasts, new york, razer, logitech, keychron,
choice" />
    <meta name="description" content="We are a group of
keyboard enthusiasts. Our company is based in New York.
Our job is to offer the best keyboards, and it's up to
you to choose." />
    <title>KeyTech | About Us</title>
  </head>
  <body>
    <nav>
      <div id="nav-warpper" class="wrapper">
        <a href="index.html"><img
src="assets/img/logo.png" id="logo" alt="logo" /></a>
        <div id="nav-close-container">
          <div id="nav-close" class="close"></div>
        </div>
      </div>
    </nav>
    <header id="about-us">
      <div class="wrapper">
        <h1>About us</h1>
        <div id="about-us-container">
          <p>KeyTech was formed in 2017 by a group of
keyboard enthusiasts. Our company is based in New York.
We sell keyboards from renowned brands like Logitech,
Keychron, Razer, and more. We set aside our time when
```

```
choosing keyboards to give you the best choice and you
don't have to waste time.</p>
          <img src="assets/img/store.jpg" alt="Store
building" />
        </div>
      </div>
    </header>
    <section id="about-us-map">
      <iframe
src="https://www.google.com/maps/embed?pb=!1m18!1m12!1m3
!1d193595.15831093167!2d-
74.11976278775128!3d40.69766374797441!2m3!1f0!2f0!3f0!3m
2!1i1024!2i768!4f13.1!3m3!1m2!1s0x89c24fa5d33f083b%3A0xc
80b8f06e177fe62!2sNew%20York%2C%20NY%2C%20USA!5e0!3m2!1s
en!2srs!4v1650319694935!5m2!1sen!2srs" loading="lazy"
referrerpolicy="no-referrer-when-downgrade"></iframe>
    </section>
    <section class="shopping-card-container">
      <div id="card-close"></div>
      <p id="card-heading">Card</p>
      <div id="card-products"></div>
      <div>
        <p id="card-total-price"></p>
      </div>
      <div id="card-btns">
        <a href="checkout.html" class="btn-
sec">Checkout</a>
        <input type="button" class="btn-sec" id="delete-
all-products" value="Clear" />
      </div>
    </section>
    <footer id="footer"></footer>
    <script
src="https://kit.fontawesome.com/92e251ce56.js"
crossorigin="anonymous"></script>
    <script src="assets/js/main.min.js"></script>
```

```
  </body>
</html>
```

3.2 CSS

3.2.1 style.css

```
@import
url("https://fonts.googleapis.com/css2?family=Roboto:wgh
t@300;400;700;900&display=swap");

:root {
  --primary-font: #f2f2f2;
}

/* CSS reset */
*,
*::before,
*::after {
  margin: 0;
  padding: 0;
  box-sizing: border-box;
}

body {
  font-size: 100%;
  font-family: "Roboto", sans-serif;
}

a {
  text-decoration: none;
}

ul {
```

```
  list-style-type: none;
}

img {
  max-width: 100%;
}
h1 {
  display: inline-block;
  margin: 50px 0;
  font-size: 2.5rem;
  position: relative;
}
h1::after {
  content: "";
  position: absolute;
  bottom: 0;
  left: 0;
  width: 120%;
  height: 3px;
  background-color: red;
}
.wrapper {
  margin: 0 10vw;
}
body {
  position: relative;
}
nav {
  position: fixed;
  margin-top: -90px;
  width: 100%;
  background-color: var(--primary-font);
  z-index: 2;
}
nav ul li a {
  font-size: 1.2rem;
```

```
}

header {
  margin-top: 90px;
}

nav .wrapper {
  display: flex;
  justify-content: space-between;
  align-items: center;
  padding: 30px 0;
  z-index: 3;
}
.navbar-list {
  display: flex;
  justify-content: space-between;
  align-items: center;
}
.navbar-list li {
  list-style-type: none;
}

.navbar-list li a {
  padding: 5px;
  margin: 0 5px;
  color: black;
  position: relative;
}
.navbar-list li a:hover::after {
  content: "";
  position: absolute;
  bottom: 0;
  left: 0;
  width: 100%;
  height: 3px;
  background-color: red;
```

```
  transition: all 1s;
}

#nav-close-container {
  display: none;
  position: relative;
  width: 28px;
  height: 24px;
  z-index: 2;
}
#nav-close-container:hover .close {
  opacity: 1;
}
#nav-close {
  z-index: 2;
}
#nav-close,
#nav-close::after,
#nav-close::before {
  background-color: #3d3d3d;
}

/* Landing page */
#landing-page {
  background-image: url("/assets/img/header-
graphic.svg");
  background-repeat: no-repeat;
  background-size: cover;
}
#landing-page h1 {
  color: var(--primary-font);
}
header {
  position: relative;
}
#store-container {
```

```
  display: flex;
  flex-direction: row-reverse;
  min-height: 100vh;
}
#side-filter {
  width: 200px;
  padding-top: 20px;
  padding-left: 10px;
  background-color: #006bb2;
}
#side-filter p {
  font-size: 1.2rem;
  font-weight: 600;
  pointer-events: none;
  margin: 5px 0;
}
#side-filter label,
input {
  font-size: 1.1rem;
}
#sort-product-placeholder {
  font-size: 1rem;
  padding: 5px;
  margin-bottom: 5px;
  border-radius: 5px;
}

footer {
  background-color: #006bb2;
}
footer .wrapper {
  display: flex;
  align-items: center;
  flex-direction: column;
}
```

```css
#footer-top {
  display: flex;
  position: relative;
  justify-content: space-between;
  align-items: flex-start;
  padding: 20px 0 10px 0;
}
#footer-top::after {
  position: absolute;
  content: "";
  bottom: 0;
  left: 0;
  width: 100%;
  height: 2px;
  background-color: #00497a;
  border-radius: 50%;
}
#footer-top > div {
  width: 30vw;
  display: flex;
  flex-direction: column;
  justify-content: center;
  align-items: center;
}
#footer-top > div > p {
  font-weight: 600;
  margin-bottom: 5px;
  position: relative;
  font-size: 1.4rem;
  color: var(--primary-font);
}
#footer-top > div > p::after {
  position: absolute;
  content: "";
  bottom: 0;
  left: 0;
```

```
  width: 130%;
  height: 3px;
  background-color: var(--primary-font);
}
#footer-top ul li a {
  color: var(--primary-font);
  font-size: 1.2rem;
}

#footerNav p {
  margin-left: 25px;
}
#footerSocial p {
  margin-left: -25px;
}
#footerDocument p {
  margin-left: -75px;
}
#footer-bottom {
  padding: 10px 0;
}
#footer-bottom p {
  color: #002d4b;
}
/* Product */
#products {
  display: flex;
  flex-wrap: wrap;
  justify-content: space-around;
  gap: 50px;
  width: 100%;
  padding-bottom: 50px;
}
#right-products-container {
  background-image: url("/assets/img/store-
background.svg");
```

```
  background-repeat: no-repeat;
  background-size: cover;
  width: 100%;
}
#right-products-container h1 {
  margin-left: 20px;
  color: var(--primary-font);
}
.product {
  display: flex;
  flex-direction: column;
  position: relative;
  height: 450px;
  width: 250px;
  background-color: rgb(164 211 255);
  padding: 20px;
  border-radius: 10px;
}
.product img {
  margin-bottom: 20px;
  border-radius: 5px;
  background-color: var(--primary-font);
  height: 200px;
  object-fit: contain;
}
.product .product-category-name {
  color: gray;
  margin-bottom: 10px;
}
.product .product-add-to-card-btn {
  align-self: flex-end;
  display: block;
  padding: 5px 10px;
}
.product .product-description {
  padding: 5px 0;
```

```
  color: gray;
}
.product .product-price {
  display: flex;
  flex-direction: column;
}
.product .product-price .product-price-new,
.product .product-price .product-price-old {
  align-self: center;
}
.product .product-add-to-card-btn {
  background-color: #42a3ff;
  border: 0;
  padding: 10px;
  border-radius: 5px;
  margin-top: 5px;
}

/* Shop card menu */
.shopping-card-container {
  top: 0;
  right: -100%;
  position: fixed;
  display: flex;
  flex-direction: column;
  min-width: 400px;
  height: 100%;
  background-color: #004d80;
  padding: 20px;
  overflow-y: scroll;
  overflow-y: auto;
  overflow-y: scroll;
  direction: ltr;
  scrollbar-color: #d4aa70 #e4e4e4;
  scrollbar-width: thin;
  transition: all 1s;
```

```
  z-index: 3;
  color: var(--primary-font);
}
.shopping-card-active {
  right: 0;
  transition: all 0.5s;
}

.shopping-card-container::-webkit-scrollbar {
  width: 20px;
}
.shopping-card-container::-webkit-scrollbar-track {
  background-color: #e4e4e4;
}
.shopping-card-container::-webkit-scrollbar-thumb {
  border-radius: 100px;
  border: 3px solid transparent;
  background-clip: content-box;
  background-color: #929292;
}
#card-close {
  position: absolute;
  right: 30px;
  top: 20px;
  width: 26px;
  height: 26px;
}
#card-close::before {
  content: "";
  position: absolute;
  top: 0;
  left: 13px;
  width: 3px;
  height: 25px;
  background-color: rgb(180, 180, 180);
  transform: rotate(45deg);
```

```
}
#card-close::after {
  content: "";
  position: absolute;
  top: 0;
  left: 13px;
  width: 3px;
  height: 25px;
  background-color: rgb(180, 180, 180);
  transform: rotate(-45deg);
}
#card-heading {
  position: relative;
  font-size: 2rem;
  margin: 20px 0;
}
#card-heading::after {
  content: "";
  position: absolute;
  bottom: 0;
  left: 0;
  width: 45%;
  height: 3px;
  background-color: rgb(255, 255, 255);
}
.card-product {
  padding: 20px 0;
  display: flex;
}
.card-product img {
  height: 100px;
  width: 100px;
  object-fit: contain;
  margin-right: 20px;
  background-color: #ffffff;
  border-radius: 5px;
```

```
}
.card-product-right {
  width: 80%;
}
.card-product-right p {
  margin-bottom: 5px;
}
.card-product-tools {
  display: flex;
  justify-content: space-between;
  margin-top: 20px;
}
/* .card-product-tools i {
  display: block;
} */
.card-product-quantity {
  display: flex;
  justify-content: space-between;
  margin-right: 25px;
}
.card-product-quantity input[type="button"] {
  width: 25px;
  height: 25px;
}
.card-product-quantity p {
  position: relative;
  padding: 0 10px;
  margin: 0 5px;
}
.card-product-quantity p::after {
  position: absolute;
  content: "";
  bottom: 0;
  left: 0;
  width: 100%;
  height: 2px;
```

```
  background-color: red;
}
.card-product:last-child {
  position: relative;
}
.card-product:last-child::after {
  position: absolute;
  content: "";
  bottom: 0;
  left: 0;
  width: 100%;
  height: 3px;
  background-color: red;
}
/* Empty card */
#empty-card {
  margin-bottom: 10px;
  font-size: 1.2rem;
  font-weight: 500;
}
.fa-trash:hover {
  color: red;
}
/* Btns */
#card-btns {
  display: flex;
  justify-content: space-between;
  margin-top: 10px;
}
.btn-sec {
  background-color: #0099ff;
  padding: 5px 10px;
  border: none;
  border-radius: 5px;
  cursor: pointer;
  color: #f2f2f2;
```

```
}
.btn-sec:hover {
  background-color: #003355;
  color: var(--primary-font);
}

/* Message modal */
.message {
  position: fixed;
  bottom: 20px;
  right: 40px;
  padding: 10px;
  background-color: rgb(211, 211, 211);
  animation: addToCardMessageAnimation 5s;
  opacity: 0;
  z-index: 5;
}
.message::before {
  content: "";
  position: absolute;
  width: 10px;
  height: 100%;
  top: 0;
  left: -5px;
  background-color: #135988;
}

/* Slideshow */
.slideshow-container {
  height: 300px;
  position: relative;
}

.slide {
  display: none;
  height: 100%;
```

```
  text-align: center;
  transition: all;
}
.slide img {
  object-fit: scale-down;
  height: 120%;
}
.prev,
.next {
  position: absolute;
  background: rgb(185, 185, 185);
  padding: 10px;
  top: 43%;
  font-size: 24px;
  margin-top: -10px;
  color: #eee;
  border-radius: 5px;
  user-select: none;
  cursor: pointer;
  transition: all 0.2s;
}

.prev {
  left: 5px;
}
.next {
  right: 5px;
}

/* Checkout */
#checkout {
  min-height: 100vh;
  background-image: url("/assets/img/checkout-bg.svg");
  background-position: top;
  background-repeat: no-repeat;
  background-size: cover;
```

```
}
#checkout h1 {
  color: var(--primary-font);
}
#checkout-container {
  display: flex;
}
#checkout-left,
#checkout-right {
  width: 50%;
}
#checkout-left input,
#checkout-left textarea {
  border: 2px solid rgb(164, 211, 255);
  border-radius: 5px;
}
#checkout-right {
  background-color: rgb(164, 211, 255);
  border-radius: 20px;
  height: 50%;
}
#checkout-product-list {
  display: flex;
  flex-direction: column;
  overflow-y: scroll;
  overflow-y: auto;
  overflow-y: scroll;
  direction: ltr;
  scrollbar-color: #d4aa70 #e4e4e4;
  scrollbar-width: thin;
  height: 300px;
  padding: 20px;
}

#checkout-product-list::-webkit-scrollbar {
  width: 20px;
```

```
}
#checkout-product-list::-webkit-scrollbar-track {
  border-radius: 0 10px 0 0;
  background-color: rgb(201 220 237);
}
#checkout-product-list::-webkit-scrollbar-thumb {
  border-radius: 100px;
  border: 3px solid transparent;
  background-clip: content-box;
  background-color: #a0a0a0;
}

#checkout-total {
  font-weight: 800;
  font-size: 1.4rem;
  padding: 5px 20px;
  position: relative;
}
#checkout-total::before {
  content: "";
  top: 0;
  left: 0;
  width: 100%;
  height: 3px;
  background-color: red;
  position: absolute;
}

#checkout-product-list .card-product:last-child::after {
  display: none;
}

.label-container label {
  display: block;
  font-size: 1.2rem;
  padding: 7px 0;
```

```
}
#form-bottom label {
  display: inline-block;
}
#form-bottom input[type="checkbox"] {
  transform: scale(1.2);
  margin-right: 3px;
}

.label-container input[type="text"],
.label-container input[type="number"] {
  padding: 5px;
  width: 60%;
}
.label-container textarea {
  padding: 5px;
  min-height: 150px;
  width: 60%;
}
.button-container .btn-primary {
  height: 35px;
  width: 100px;
  display: block;
  margin-top: 20px;
}
.btn-primary {
  background-color: gray;
  position: relative;
  border: none;
  z-index: 1;
  cursor: pointer;
  display: block;
  border-bottom: #65a10e 2px solid;
}
.btn-primary:hover {
  transition: 0.5s ease-in-out;
```

```
  background-color: rgb(17, 148, 17);
}
.bad-form-element {
  color: red;
  padding-top: 5px;
}
#checkout-container {
  margin-top: 50px;
}
/* About me */
#about-me-container {
  position: absolute;
  top: 10vh;
  left: 10vw;
  animation: fadeIn 1s;
}
#about-me {
  background-color: #2a427c;
  max-height: 100vh;
  width: 80vw;
  margin: 0 auto;
  display: flex;
  position: relative;
}
#about-me-right {
  padding: 20px 40px;
}
#about-me-left {
  min-width: 200px;
  background-color: #3a5cad;
}
#about-me-righ {
  width: 70%;
}
#about-me-content {
  height: 80%;
```

```
  display: flex;
  flex-direction: column;
  justify-content: space-between;
}
#about-me-left img {
  display: block;
  max-height: 70%;
  background-color: #2b2b2b;
  object-fit: cover;
}
#about-social {
  display: flex;
  flex-direction: row;
}
#about-me-info-content li {
  padding: 7%;
  position: relative;
}
#about-me-info-content li::after {
  content: "";
  position: absolute;
  bottom: 0;
  left: 0;
  width: 100%;
  height: 1px;
  background-color: var(--primary-font);
}
#about-me-info-content li {
  color: var(--primary-font);
}
#about-me-info-content li a,
#about-me-info-content li p span {
  color: var(--primary-font);
  filter: brightness(0.85);
}
#about-social ul {
```

```
  display: flex;
}
#about-social {
  padding: 7%;
}
#about-social li a {
  font-size: 2rem;
  margin-right: 20px;
  color: var(--primary-font);
}
#about-me-right h2 {
  font-size: 2.5rem;
  position: relative;
  display: inline-block;
  margin-bottom: 5%;
  color: var(--primary-font);
}
#about-me-right h2::after {
  content: "";
  position: absolute;
  bottom: 0;
  left: 0;
  width: 120%;
  height: 3px;
  background-color: var(--primary);
}
#about-me-text {
  font-size: 1.3rem;
  color: var(--primary-font);
}
/* Quotes */
#quote-box {
  opacity: 0;
  animation: fade infinite 2s;
  color: rgb(160, 160, 160);
}
```

```
#quote {
  font-size: 1.2rem;
}
#author {
  text-align: right;
  padding: 5px;
}
#about-me #about-me-close {
  position: absolute;
  top: 20px;
  right: 20px;
  width: 28px;
  height: 24px;
}
#about-me #about-me-close .close {
  position: absolute;
  width: 32px;
  right: 10px;
  top: 10px;
}
#about-me-close:hover .close {
  opacity: 1;
}
#about-me::after {
  content: "";
  position: absolute;
  bottom: 0;
  left: 0;
  width: 100%;
  height: 3px;
  background-color: var(--primary-font);
}
#about-me-close::before,
#about-me-close::after {
  background-color: gray;
}
```

```
.cross::before,
.cross::after {
  background-color: red;
}
.close {
  position: absolute;
  left: 0;
  top: 10px;
  width: 100%;
  height: 3px;
  opacity: 0.5;
  transition: all 0.3s;
  background-color: rgb(128, 128, 128);
  transition: all 0.3s ease;
}
.close:hover {
  opacity: 1;
}
.close:before,
.close:after {
  content: "";
  position: absolute;
  left: 0;
  height: 3px;
  width: 100%;
  background-color: var(--primary-font);
}
.close:before {
  top: -10px;
}
.close:after {
  top: 10px;
}
.cross::before {
  transform: rotate(-45deg) translate(-8px, 6px);
  opacity: 1;
```

```
  background-color: rgb(128, 128, 128);
}
.cross {
  height: 0;
  transform: rotate(180deg);
}
.cross::after {
  transform: rotate(45deg) translate(-8px, -6px);
  opacity: 1;
  background-color: rgb(128, 128, 128);
}

/* About us */
#about-us {
  min-height: 80vh;
  background-image: url("/assets/img/about-us-bg.svg");
  background-position: bottom;
  background-repeat: no-repeat;
  background-size: cover;
}
#about-us h1 {
  color: var(--primary-font);
}
#about-us-container {
  display: flex;
  margin-top: 50px;
  gap: 20px;
}
#about-us-container p {
  font-size: 1.5rem;
  line-height: 1.5;
}
#about-us-container img {
  width: 40%;
  border-radius: 10px;
  object-fit: cover;
```

```
}
#about-us-map iframe {
  display: block;
  background-color: #0099ff;
  height: 30vh;
  width: 100%;
  border: 0;
}
/* Response */
@media only screen and (max-width: 1025px) {
  #checkout-left input[type="text"],
  .label-container input[type="number"],
  #checkout-left textarea {
    width: 80%;
  }
}

@media only screen and (max-width: 769px) {
  /* Navigation */
  .navbar-list {
    position: fixed;
    right: 10vw;
    top: 0;
    flex-direction: column;
    justify-content: flex-start;
    height: 100vh;
    width: 60%;
    padding-top: 40px;
    transform: translateX(100%);
    transition: transform 0.5s ease-in-out;
    background-color: #006bb2;
    margin: 0 -10vw;
  }
  .navbar-list li {
    padding: 20px;
  }
```

```
.navbar-list li a {
  color: var(--primary-font);
}
#nav-close-container {
  display: block;
  position: fixed;
  right: 20px;
  top: 20px;
  z-index: 5;
}
.nav-active {
  transform: translateX(0%);
}

/* About me */
#about-me-container {
  left: 5vw;
}
#about-me {
  max-height: unset;
  width: 90vw;
}
#about-me-right {
  padding: 20px;
}
#about-me-right h2 {
  font-size: 1.5rem;
}
#about-me-text {
  font-size: 1.1rem;
}
#quote {
  font-size: 1rem;
}
#checkout-container {
  flex-direction: column-reverse;
```

```
    gap: 30px;
  }
  #checkout-right,
  #checkout-left {
    width: 100%;
    padding-bottom: 50px;
  }
  #checkout-left input[type="text"],
  .label-container input[type="number"],
  #checkout-left textarea {
    width: 100%;
  }
  /* About us */
  #about-us h1 {
    margin: 50px 0 0 0;
  }
  #about-us-container {
    flex-direction: column;
  }
  #about-us-container img {
    width: 100%;
    margin-bottom: 20px;
  }
}

@media only screen and (max-width: 481px) {
  #about-me {
    flex-direction: column-reverse;
  }
  #about-me-left {
    display: flex;
    flex-direction: column-reverse;
  }
  #quote-box {
    padding-top: 20px;
  }
```

```
  .sponsor-item i {
    font-size: 5rem;
  }
  #store-container {
    flex-direction: column-reverse;
  }
  #side-filter {
    width: 100%;
    display: flex;
    justify-content: center;
    align-items: center;
    flex-direction: row;
    flex-wrap: wrap;
    gap: 5%;
  }
  #side-filter > div {
    width: 40%;
  }
  #side-filter > input {
    display: block;
    width: 40%;
  }
  #footer-top {
    flex-wrap: wrap;
    justify-content: center;
    gap: 25px;
  }
  .shopping-card-container {
    width: 100%;
    min-width: auto;
  }
}

/* Animations */
@keyframes addToCardMessageAnimation {
  0% {
```

```
    right: -100px;
  }
  25% {
    right: 40px;
  }
  50% {
    opacity: 1;
  }
  80% {
    opacity: 1;
  }
  100% {
    opacity: 0;
  }
}
@keyframes fade {
  20% {
    opacity: 1;
  }
  80% {
    opacity: 1;
  }
}
```

3.3 JS

3.3.1 main.js

```
window.onload = () => {
  fetchData("navLinks", createNavBar);
  fetchData("categories");
  fetchData("brands");
  fetchData("connectionTypes");
  fetchData("quotes");
  fetchData("navLinks", createFooterNavLinks);
  fetchData("socialLinks", createFooterSocialLinks);
```

```
  fetchData("documentLinks", createFooterDocumentLinks);

  function fetchData(fileName, callBack) {
    fetch(`./assets/data/${fileName}.json`)
      .then((response1) => response1.json())
      .then((response2) => {
        if (callBack != null) {
          callBack(response2);
        }
        return response2;
      })
      .then((response3) => {
        localStorage.setItem(`${fileName}Local`,
JSON.stringify(response3));
        return response3;
      })
      .then((response4) => {
        return response4;
      });
  }

  function createNavBar(data) {
    navLinks("nav-warpper", data);
    navLinkShop("navbar-container");
    eventListener("shopping-card", "id", "click",
openCard);
    eventListener("card-close", "id", "click",
closeCard);
    createFooter();
  }

  function navLinkShop(ulID) {
    let ulTag = document.getElementById(`${ulID}`);
    let liTag = document.createElement("li");
    let iTag = document.createElement("i");
```

```
    iTag.setAttribute("class", "fa-solid fa-basket-
shopping");
    iTag.setAttribute("id", "shopping-card");
    liTag.appendChild(iTag);
    let pTag = document.createElement("span");
    pTag.setAttribute("id", "incard-product-number");
    liTag.appendChild(pTag);
    ulTag.appendChild(liTag);
    inCardCount();
  }

  function navLinks(id, navLinksLocal) {
    let container = document.getElementById(`${id}`);
    let ulTag = document.createElement("ul");
    ulTag.setAttribute("id", "navbar-container");
    ulTag.setAttribute("class", "navbar-list");
    for (let link of navLinksLocal) {
      let liTag = document.createElement("li");
      let aTag = document.createElement("a");
      let aTagContent =
document.createTextNode(`${link.name}`);
      aTag.append(aTagContent);
      aTag.setAttribute("href", `${link.link}`);
      liTag.appendChild(aTag);
      ulTag.appendChild(liTag);
    }
    container.appendChild(ulTag);
  }

  function brandsFilter(productsArray) {
    let selectedBrands = [];
    let brandsElements =
document.querySelectorAll(".brands-filter-
item:checked");
    for (let element of brandsElements) {
      selectedBrands.push(parseInt(element.value));
```

```
    }
    if (selectedBrands.length != 0) {
      return productsArray.filter(function (el) {
        for (let item of selectedBrands) {
          if (el.brandID == item) {
            return el;
          }
        }
      });
    }
    return productsArray;
  }

  function categoryFilter(productsArray) {
    let selectedCategory = [];
    let categoryElements =
document.querySelectorAll(".category-filter-
item:checked");
    for (let element of categoryElements) {
      selectedCategory.push(parseInt(element.value));
    }
    if (selectedCategory.length != 0) {
      return productsArray.filter(function (el) {
        for (let item of selectedCategory) {
          if (el.brandID == item) {
            return el;
          }
        }
      });
    }
    return productsArray;
  }

  function connectionTypeFilter(productsArray) {
    let connectionTypes = [];
```

```
    let connectionTypeElement =
document.querySelectorAll(".connection-type-filter-
item:checked");
    for (let element of connectionTypeElement) {
      connectionTypes.push(parseInt(element.value));
    }
    if (connectionTypes.length != 0) {
      return productsArray.filter(function (el) {
        for (let item of connectionTypes) {
          if (el.connectionTypeID == item) {
            return el;
          }
        }
      });
    }
    return productsArray;
  }

  function showProducts(productsArray) {
    productsArray = brandsFilter(productsArray);
    productsArray = categoryFilter(productsArray);
    productsArray = connectionTypeFilter(productsArray);
    productsArray = reviewsFilter(productsArray);
    let html = "";
    let container = document.getElementById("products");
    if (productsArray.length != 0) {
      for (let product of productsArray) {
        html += `<div class="product">
                <img src="${product.image.src}"
alt="${product.image.alt}" />
                <p class="product-category-
name">${categoryName(product.categoryID)}</p>
                <p class="product-
brandname">${showBrand(product.brandID)}</p>
                <p class="product-
name">${product.name}</p>
```

```
                <p class="product-
description">${product.descript}</p>
                <p class="product-reviews">Reviews:
${product.score}<i class="fa-solid fa-star"></i></p>
                <div class="product-price">
                  <p>Price:</p>
                  <s class="product-price-
old">${product.price.discount != 0 ? product.price.old +
"$" : ""}</s>
                  <p class="product-price-
new">${priceCalculator(product.price.old,
product.price.discount)}$</p>
                </div>
                <input class="product-add-to-card-btn"
type="button" data-id="${product.id}" value="Add to
card" />
              </div>`;
      }
    } else {
      html += `<div id="product-message">
                  <h2>There is no products with selected
criteria!</h2>
                </div>`;
    }
    container.innerHTML = html;
    inCardProductsShow();
    let scoreP = document.getElementById("reviews-range-
place");
    if (scoreP != null) {
      let rangeSlider =
document.getElementById("reviews-range-sllider");
      scoreP.innerHTML = rangeSlider.value / 10;
    }
    eventListener("product-add-to-card-btn", "class",
"click", addToCard);
  }
```

```
  function categoryName(productCategoryID) {
    let categoriesArray = jsonParse("categoriesLocal");
    let categoryName = categoriesArray.filter(function
(el) {
      return el.id == productCategoryID;
    });
    return categoryName[0].name;
  }
  function priceCalculator(productPrice,
productDiscount) {
    let price = productPrice * ((100 - productDiscount)
/ 100);
    return price.toFixed(2);
  }
  function showBrand(brandID) {
    let html = "";
    let brandsArray = jsonParse("brandsLocal");
    for (let brand of brandsArray) {
      if (brand.id == brandID) {
        html += `${brand.name}`;
      }
    }
    return html;
  }

  function filterProducts() {
    let productsArray = jsonParse("productsLocal");
    showProducts(productsArray);
  }
  // Make side filters
  function showFiltersBrand(data) {
    showProductsFilter("Brands", "brands-filter", data,
"brands-filter-item");
  }
  function showFiltersCategory(data) {
```

```
    showProductsFilter("Category", "category-filter",
data, "category-filter-item");
  }
  function showFiltersConnectionType(data) {
    showProductsFilter("Connection", "connection-type-
filter", data, "connection-type-filter-item");
  }

  function showProductsFilter(filterName, containerName,
arrayName, className) {
    let html = "";
    html += `<p>${filterName}</p>
              <ul>`;
    let container =
document.getElementById(containerName);
    for (let item of arrayName) {
      html += `<li>
                    <input type="checkbox"
name="${item.name}" class="${className}"
value="${item.id}" id="${item.name}">
                    <label
for="${item.name}">${item.name} </label>
                  </li>`;
    }
    html += `</ul>`;
    container.innerHTML = html;
    eventListener(className, "class", "change",
filterProducts);
  }

  // Show Reviews side filter

  function showReviews() {
    let container = document.getElementById("reviews-
filter");
    let html = `<p>Reviews</p>
```

```
          <div class="slidecontainer">
            <input type="range" min="10" max="50"
value="1" class="slider reviews-filter-item"
id="reviews-range-sllider" />
            <p id="reviews-range-place"></p>
          </div>`;
    container.innerHTML = html;
    let score = document.getElementById("reviews-range-
sllider");
    let scoreP = document.getElementById("reviews-range-
place");
    scoreP.innerHTML = "1";
    score.addEventListener("click", filterProducts);
  }
  // Filter products per reviews
  function reviewsFilter(productsArray) {
    let scoreContainer =
document.getElementById("reviews-range-sllider");
    let score;
    if (scoreContainer == null) {
      score = 1;
    } else {
      score = scoreContainer.value / 10;
    }
    return productsArray.filter(function (el) {
      if (el.score >= score) {
        return el;
      }
    });
  }

  function createFooter() {
    let footerId = document.getElementById("footer");
    let footerWrapper = document.createElement("div");
    footerWrapper.setAttribute("class", "wrapper");
    let footerTop = document.createElement("div");
```

```
    let navLinksLocal = jsonParse("navLinksLocal");
    footerTop.setAttribute("id", "footer-top");
    let footerBottom = document.createElement("div");
    let footerCopyright = document.createElement("p");
    let footerCopyrightContent =
document.createTextNode("© Copyright 2022");
    footerCopyright.appendChild(footerCopyrightContent);
    footerBottom.appendChild(footerCopyright);
    footerBottom.setAttribute("id", "footer-bottom");
    footerWrapper.appendChild(footerTop);
    footerWrapper.appendChild(footerBottom);
    footerId.appendChild(footerWrapper);
  }

  function createFooterNavLinks(data) {
    let footerTop = document.getElementById("footer-
top");
    createUlLinks(data, footerTop, "footerNav",
"Navigation");
    aboutMePage();
  }
  function createFooterSocialLinks(data) {
    let footerTop = document.getElementById("footer-
top");
    createUlLinks(data, footerTop, "footerSocial",
"Social");
  }
  function createFooterDocumentLinks(data) {
    let footerTop = document.getElementById("footer-
top");
    createUlLinks(data, footerTop, "footerDocument",
"Links");
  }

  function createUlLinks(dataArray, mainContainer,
divId, pContent) {
```

```
    let divContainer = document.createElement("div");
    divContainer.setAttribute("id", `${divId}`);
    let pContaier = document.createElement("p");
    let pContaierContent =
document.createTextNode(`${pContent}`);
    pContaier.appendChild(pContaierContent);
    divContainer.appendChild(pContaier);
    let containerUl = document.createElement("ul");
    divContainer.appendChild(containerUl);
    for (let link of dataArray) {
      let navLinksLi = document.createElement("li");
      let navLinksA = document.createElement("a");
      let navLinksAContent =
document.createTextNode(`${link.name}`);
      navLinksA.setAttribute("href", `${link.link}`);
      navLinksA.appendChild(navLinksAContent);
      navLinksLi.appendChild(navLinksA);
      containerUl.appendChild(navLinksLi);
    }
    mainContainer.appendChild(divContainer);
  }

  function eventListener(name, tagType, eventMethod,
functionName) {
    if (tagType == "id") {
      let item = document.querySelector(`#${name}`);
      item.addEventListener(`${eventMethod}`,
functionName);
    } else {
      let items = document.querySelectorAll(`.${name}`);
      for (let item of items) {
        item.addEventListener(`${eventMethod}`,
functionName);
      }
    }
  }
```

```
  function addToCard() {
    let dataID = this.getAttribute("data-id");
    if (localStorage.getItem("addToCardList")) {
      let addToCardList = jsonParse("addToCardList");
      if (addToCardList.filter((e) => e.id ==
dataID).length) {
        cardMessages("Product already in card!");
      } else {
        addToCardList.push({
          id: dataID,
          quantity: 1,
        });
        cardMessages("Product added to card!");
        localStorage.setItem("addToCardList",
JSON.stringify(addToCardList));
      }
    } else {
      let jsonTe = [];
      jsonTe[0] = {
        id: dataID,
        quantity: 1,
      };
      localStorage.setItem("addToCardList",
JSON.stringify(jsonTe));
      cardMessages("Product added to card!");
    }
    inCardCount();
    inCardProductsShow();
    classGe();
  }

  // Test shopping-card-container
  function openCard() {
    let shoppingCardContainer =
document.querySelector(".shopping-card-container");
```

```
    shoppingCardContainer.classList.add("shopping-card-
active");
  }
  function closeCard() {
    let shoppingCardContainer =
document.querySelector(".shopping-card-container");
    shoppingCardContainer.classList.remove("shopping-
card-active");
  }

  // Add to card
  const addToCardBtnsList =
document.querySelectorAll(".product-add-to-card-btn");
  for (let addToCardBtn of addToCardBtnsList) {
    addToCardBtn.addEventListener("click", function () {
      let dataID = this.getAttribute("data-id");
      if (localStorage.getItem("addToCardList")) {
        let addToCardList = jsonParse("addToCardList");
        if (addToCardList.filter((e) => e.id ==
dataID).length) {
          cardMessages("Product already in card!");
        } else {
          addToCardList.push({
            id: dataID,
            quantity: 1,
          });
          cardMessages("Product added to card!");
          localStorage.setItem("addToCardList",
JSON.stringify(addToCardList));
        }
      } else {
        let jsonTe = [];
        jsonTe[0] = {
          id: dataID,
          quantity: 1,
        };
```

```
        localStorage.setItem("addToCardList",
JSON.stringify(jsonTe));
      }
      inCardCount();
      inCardProductsShow();
    });
  }

  // Add to card message
  function cardMessages(text) {
    let messageDivTest =
document.querySelector(".message");
    if (messageDivTest) {
      messageDivTest.parentNode.removeChild(messageDivTe
st);
    }
    let messageDiv = document.createElement("div");
    messageDiv.setAttribute("class", "message");
    let pTag = document.createElement("p");
    let pTagContent =
document.createTextNode(`${text}`);
    pTag.appendChild(pTagContent);
    messageDiv.appendChild(pTag);
    let bodyTag = document.getElementsByTagName("body");
    bodyTag[0].appendChild(messageDiv);
    var clearElement = setInterval(() => {
      if (messageDiv.parentElement == null) {
        clearInterval(clearElement);
      } else {
        messageDiv.parentElement.removeChild(messageDiv)
;
      }
    }, 5000);
  }

  // In card counter
```

```
  function inCardCount() {
    let pTag = document.getElementById("incard-product-
number");
    if (localStorage.getItem("addToCardList")) {
      let cardCounts =
jsonParse("addToCardList").length;
      pTag.innerHTML = ` (${cardCounts})`;
    } else {
      pTag.innerHTML = " (0)";
    }
  }
  // InCardProducts
  function inCardProductsShow() {
    let addToCardList = jsonParse("addToCardList");
    let container = document.getElementById("card-
products");
    let checkoutContainer =
document.getElementById("checkout-product-list");
    container.innerHTML = "";
    if (addToCardList != null) {
      for (let item of addToCardList) {
        let cardProdict = document.createElement("div");
        cardProdict.setAttribute("class", "card-
product");
        let imgContainer =
document.createElement("img");
        imgContainer.setAttribute("src",
`${showInCardProductData(item.id, "image", "src")}`);
        imgContainer.setAttribute("alt",
`${showInCardProductData(item.id, "image", "alt")}`);
        cardProdict.appendChild(imgContainer);
        let cardProduictsRight =
document.createElement("div");
        cardProduictsRight.setAttribute("class", "card-
product-right");
        cardProdict.appendChild(cardProduictsRight);
```

```
        let pProductName = document.createElement("p");
        let pProductNameContent =
document.createTextNode(`${showInCardProductData(item.id
, "name")}`);
        pProductName.appendChild(pProductNameContent);
        cardProduictsRight.appendChild(pProductName);
        let pBrandName = document.createElement("p");
        let pPBrandNameContent =
document.createTextNode(`${showInCardProductData(item.id
, "brandID")}`);
        pBrandName.appendChild(pPBrandNameContent);
        cardProduictsRight.appendChild(pBrandName);
        let pProductPrice = document.createElement("p");
        let pProductPriceContent =
document.createTextNode(`${showInCardProductData(item.id
, "price", "new")}`);
        pProductPrice.appendChild(pProductPriceContent);
        cardProduictsRight.appendChild(pProductPrice);
        let cardProductTools =
document.createElement("div");
        cardProductTools.setAttribute("class", "card-
product-tools");
        cardProduictsRight.appendChild(cardProductTools)
;
        let cardProductQuantity =
document.createElement("div");
        cardProductQuantity.setAttribute("class", "card-
product-quantity");
        let inputValueMinus =
document.createElement("input");
        inputValueMinus.setAttribute("type", "button");
        inputValueMinus.setAttribute("class",
"quantityRegul");
        inputValueMinus.setAttribute("data-id",
`${item.id}`);
        inputValueMinus.setAttribute("value", "-");
```

```
        cardProductQuantity.appendChild(inputValueMinus)
;
        let itemQuantity = document.createElement("p");
        let itemQuantityContent =
document.createTextNode(`${item.quantity}`);
        itemQuantity.appendChild(itemQuantityContent);
        cardProductQuantity.appendChild(itemQuantity);
        let inputValuePlus =
document.createElement("input");
        inputValuePlus.setAttribute("type", "button");
        inputValuePlus.setAttribute("class",
"quantityRegul");
        inputValuePlus.setAttribute("data-id",
`${item.id}`);
        inputValuePlus.setAttribute("value", "+");
        cardProductQuantity.appendChild(inputValuePlus);
        cardProductTools.appendChild(cardProductQuantity
);
        let iBin = document.createElement("i");
        iBin.setAttribute("class", "fa fa-trash");
        iBin.setAttribute("data-id", `${item.id}`);
        iBin.setAttribute("aria-hidden", "true");
        cardProductTools.appendChild(iBin);
        container.appendChild(cardProdict);
      }
    } else {
      let pTag = document.createElement("p");
      let pTagContent = document.createTextNode("Card is
empty.");
      pTag.appendChild(pTagContent);
      pTag.setAttribute("id", "empty-card");
      container.appendChild(pTag);
    }

    if (addToCardList != null && checkoutContainer !=
null) {
```

```
      container = document.getElementById("checkout-
product-list");
      container.innerHTML = "";
      for (let item of addToCardList) {
        let cardProdict = document.createElement("div");
        cardProdict.setAttribute("class", "card-
product");
        let imgContainer =
document.createElement("img");
        imgContainer.setAttribute("src",
`${showInCardProductData(item.id, "image", "src")}`);
        imgContainer.setAttribute("alt",
`${showInCardProductData(item.id, "image", "alt")}`);
        cardProdict.appendChild(imgContainer);
        let cardProduictsRight =
document.createElement("div");
        cardProduictsRight.setAttribute("class", "card-
product-right");
        cardProdict.appendChild(cardProduictsRight);
        let pProductName = document.createElement("p");
        let pProductNameContent =
document.createTextNode(`${showInCardProductData(item.id
, "name")}`);
        pProductName.appendChild(pProductNameContent);
        cardProduictsRight.appendChild(pProductName);
        let pBrandName = document.createElement("p");
        let pPBrandNameContent =
document.createTextNode(`${showInCardProductData(item.id
, "brandID")}`);
        pBrandName.appendChild(pPBrandNameContent);
        cardProduictsRight.appendChild(pBrandName);
        let pProductPrice = document.createElement("p");
        let pProductPriceContent =
document.createTextNode(`${showInCardProductData(item.id
, "price", "new")}`);
        pProductPrice.appendChild(pProductPriceContent);
```

```
        cardProduictsRight.appendChild(pProductPrice);
        container.appendChild(cardProdict);
      }
    } else if (checkoutContainer != null) {
      container = document.getElementById("checkout-
product-list");
      container.innerHTML = "";
    }
    deleteProducts();
    classGe();
  }

  function totalPrice() {
    let addToCardList = jsonParse("addToCardList");
    let totalPrice = 0;
    let priceContainer = document.getElementById("card-
total-price");
    let checkoutContainer =
document.getElementById("checkout-total");

    if (addToCardList != null && addToCardList !=
"undefined") {
      let quantityPrice = addToCardList.map(function
(el) {
        let productsArray = jsonParse("productsLocal");
        for (let item of productsArray) {
          if (item.id == el.id) {
            return el.quantity *
priceCalculator(item.price.old, item.price.discount);
          }
        }
      });
      for (let item of quantityPrice) {
        totalPrice += item;
      }
```

```
      priceContainer.innerHTML = `Total:
${totalPrice.toFixed(2)}$`;
    }
    if (checkoutContainer != null && addToCardList !=
"undefined") {
      checkoutContainer.innerHTML = `Total:
${totalPrice.toFixed(2)}$`;
    }
  }
  function calculateItemPrice(itemID, price = null) {
    let addToCardList = jsonParse("addToCardList");
    let filtered = addToCardList.filter(function (el) {
      return el.id == itemID;
    });
    if (price == null) {
      let productsArray = jsonParse("productsLocal");
      let filteredPrice = productsArray.filter(function
(el) {
        return el.id == itemID;
      });
      price =
priceCalculator(filteredPrice[0].price.old,
filteredPrice[0].price.discount);
    }
    let totalPrice = filtered[0].quantity * price;
    return `${price}$ x ${filtered[0].quantity} =
${totalPrice.toFixed(2)}$`;
  }

  function showInCardProductData(itemID, objectName,
objectName2 = null) {
    let productsArray = jsonParse("productsLocal");
    let productName = productsArray.filter(function (el)
{
      return el.id == itemID;
    });
```

```
    if (objectName == "brandID") {
      let brandsArray = jsonParse("brandsLocal");
      for (let item of brandsArray) {
        if (productName[0].brandID == item.id) {
          return item.name;
        }
      }
    }
    if (objectName == "price") {
      return calculateItemPrice(productName[0].id,
productName[0][objectName][objectName2]);
    }
    if (objectName2 != null) {
      return productName[0][objectName][objectName2];
    } else {
      return productName[0][objectName];
    }
  }
  function classGe() {
    let quantityRegulList =
document.querySelectorAll(".quantityRegul");
    for (let item of quantityRegulList) {
      item.addEventListener("click", function () {
        let dataQuantity = jsonParse("addToCardList");
        if (this.value == "+") {
          for (let data of dataQuantity) {
            if (data.id == this.getAttribute("data-id"))
{
              data.quantity += 1;
              localStorage.setItem("addToCardList",
JSON.stringify(dataQuantity));
              this.parentElement.getElementsByTagName("p
")[0].innerHTML = data.quantity;
            }
          }
        } else {
```

```
          for (let data of dataQuantity) {
            if (data.id == this.getAttribute("data-id"))
{
              if (data.quantity > 1) {
                data.quantity -= 1;
                localStorage.setItem("addToCardList",
JSON.stringify(dataQuantity));
                this.parentElement.getElementsByTagName(
"p")[0].innerHTML = data.quantity;
              }
            }
          }
        }
        totalPrice();
        calculateItemPrice(item.getAttribute("data-
id"));
        inCardProductsShow();
      });
    }
  }

  function deleteProducts() {
    let binItems = document.querySelectorAll(".fa-
trash");
    for (let item of binItems) {
      item.addEventListener("click", () => {
        let addToCardList = jsonParse("addToCardList");
        for (let data of addToCardList) {
          if (data.id == item.getAttribute("data-id")) {
            let filteredList =
addToCardList.filter(function (el) {
              return el.id !== `${data.id}`;
            });
            localStorage.setItem("addToCardList",
JSON.stringify(filteredList));
          }
```

```
        }
        if (jsonParse("addToCardList").length == 0) {
          localStorage.removeItem("addToCardList");
        }
        inCardProductsShow();
        cardMessages("Product deleted from card!");
      });
    }
    totalPrice();
  }
  function deleteAllProducts() {
    localStorage.removeItem("addToCardList");
    cardMessages("Deleted all products from card");
    inCardProductsShow();
    totalPrice();
  }

  eventListener("delete-all-products", "id", "click",
deleteAllProducts);
  function jsonParse(data) {
    let jsonData =
JSON.parse(localStorage.getItem(`${data}`));
    return jsonData;
  }
  function resetFilters() {
    let brandsFilters =
document.querySelectorAll(".brands-filter-item");
    let categoriesFilters =
document.querySelectorAll(".category-filter-item");
    let connectionTypesFilters =
document.querySelectorAll(".connection-type-filter-
item");
    let sortContainer = document.getElementById("sort-
product-placeholder");
    sortContainer.selectedIndex = 0;
    resetCheckbox(brandsFilters);
```

```
    resetCheckbox(categoriesFilters);
    resetCheckbox(connectionTypesFilters);
    let reviewsFilter =
document.getElementById("reviews-range-sllider");
    reviewsFilter.value = 1;
    let productsArray = jsonParse("productsLocal");
    showProducts(productsArray);
  }
  function resetCheckbox(arrayName) {
    for (let item of arrayName) {
      item.checked = false;
    }
  }

  inCardProductsShow();
  function sortProduct(container) {
    container.addEventListener("change", function () {
      let productsArray = jsonParse("productsLocal");
      productsArray.sort(function (a, b) {
        if (sortingPlaceholder.value == 0) {
          return 0;
        } else if (sortingPlaceholder.value == 1) {
          a = parseFloat(priceCalculator(a.price.old,
a.price.discount));
          b = parseFloat(priceCalculator(b.price.old,
b.price.discount));
          if (a < b) {
            return -1;
          }
          if (a > b) {
            return 1;
          }
          if (a == b) {
            return 0;
          }
        } else if (sortingPlaceholder.value == 2) {
```

```
          a = parseFloat(priceCalculator(a.price.old,
a.price.discount));
          b = parseFloat(priceCalculator(b.price.old,
b.price.discount));
          if (a > b) {
            return -1;
          }
          if (a < b) {
            return 1;
          } else {
            return 0;
          }
        } else if (sortingPlaceholder.value == 3) {
          if (a.name < b.name) {
            return -1;
          }
          if (a.name > b.name) {
            return 1;
          } else {
            return 0;
          }
        } else if (sortingPlaceholder.value == 4) {
          if (a.name > b.name) {
            return -1;
          }
          if (a.name < b.name) {
            return 1;
          } else {
            return 0;
          }
        }
      });
      showProducts(productsArray);
    });
  }
```

```
  const url = window.location.pathname;
  if (url == "/store.html") {
    fetchData("products", showProducts);
    fetchData("brands", showFiltersBrand);
    fetchData("categories", showFiltersCategory);
    fetchData("connectionTypes",
showFiltersConnectionType);
    showReviews();
    var sortingPlaceholder =
document.getElementById("sort-product-placeholder");
    sortProduct(sortingPlaceholder);
    let resetBtn = document.getElementById("reset-
filter");
    resetBtn.addEventListener("click", resetFilters);
  } else if (url == "/index.html" || url == "/") {
    fetchData("products", topDiscountProducts);
    let slideIndex = 0;
    let slideTimer;
    const slideDelay = 3;
    const slides =
document.getElementsByClassName("slide");
    ShowSlide = (index) => {
      if (index == slides.length) slideIndex = 0;
      else if (index < 0) slideIndex = slides.length -
1;
      else slideIndex = index;
      for (i = 0; i < slides.length; i++) {
        slides[i].style.display = "none";
      }
      SlideReset();
      slides[slideIndex].style.display = "block";
    };
    SlideReset = () => {
      window.clearInterval(slideTimer);
      setTimeout(function () {
        slideTimer = window.setInterval(function () {
```

```
          ShowSlide((slideIndex += 1));
        }, slideDelay * 1000);
      }, 10);
    };
    neg.addEventListener("click", function () {
      ShowSlide((slideIndex -= 1));
    });
    pos.addEventListener("click", function () {
      ShowSlide((slideIndex += 1));
    });
    ShowSlide(slideIndex);
  } else if (url == "/checkout.html") {
    inCardProductsShow();

    // Buying form validation
    let contactSubmit =
document.getElementById("contact-submit");
    let mailNotSent = true;
    contactSubmit.addEventListener("click", () => {
      const regularFullName = /^[A-ZČĆŽĐŠ][a-
zćčžđš]{1,14}\s([A-ZČĆŽĐŠ][a-zćčžđš]{1,14})?\s?[A-
ZČĆŽŠĐ][a-zćčžđš]{1,19}$/;
      let fullNameField =
document.getElementById("fullName");
      let fullNameFieldValue = fullNameField.value;
      let fullNameConfirm;
      let mailConfirm;
      let messageConfirm;
      if (regularFullName.test(fullNameFieldValue)) {
        fullNameField.nextElementSibling.innerHTML = "";
        fullNameField.nextElementSibling.setAttribute("c
lass", "good-form-element");
        fullNameConfirm = true;
      } else {
        fullNameField.nextElementSibling.innerHTML =
"Full Name is not as expected";
```

```
        fullNameField.nextElementSibling.setAttribute("c
lass", "bad-form-element");
        fullNameConfirm = false;
      }
      const regularMail = /^[a-zA-Z0-9]([a-z]|[0-
9])+\.?-?_?([a-z]|[0-9])*\.?([a-z]|[0-9])*\@[a-
z]{3,}\.([a-z]{2,4}\.)?([a-z]{2,4})$/g;
      let mailField = document.getElementById("email");
      let mailFieldFieldValue = mailField.value;
      if (regularMail.test(mailFieldFieldValue)) {
        mailField.nextElementSibling.innerHTML = "";
        mailField.nextElementSibling.setAttribute("class
", "good-form-element");
        mailConfirm = true;
      } else {
        mailField.nextElementSibling.innerHTML = "E-mail
is not as expected!";
        mailField.nextElementSibling.setAttribute("class
", "bad-form-element");
        mailConfirm = false;
      }
      const regularPostalCode = /^[0-9]{5}$/;
      let postalCode = document.getElementById("postal-
code");
      postalCodeFieldValue = postalCode.value;
      if (regularPostalCode.test(postalCodeFieldValue))
{
        postalCode.nextElementSibling.innerHTML = "";
        postalCode.nextElementSibling.setAttribute("clas
s", "good-form-element");
        postalCodeConfirm = true;
      } else {
        postalCode.nextElementSibling.innerHTML =
"Postal code is not as expected!";
        postalCode.nextElementSibling.setAttribute("clas
s", "bad-form-element");
```

```
        postalCodeConfirm = false;
      }
      const regularStreetAddress = /^[A-Za-
zČĆŽĐŠčćžđš'\.\-\s\,0-9]{3,}$/;
      let streetAddress =
document.getElementById("address");
      streetAddressFieldValue = streetAddress.value;
      if
(regularStreetAddress.test(streetAddressFieldValue)) {
        streetAddress.nextElementSibling.innerHTML = "";
        streetAddress.nextElementSibling.setAttribute("c
lass", "good-form-element");
        streetAddressConfirm = true;
      } else {
        streetAddress.nextElementSibling.innerHTML =
"Address is not as expected!";
        streetAddress.nextElementSibling.setAttribute("c
lass", "bad-form-element");
        streetAddressConfirm = false;
      }
      let acceptToS = document.getElementById("accept-
tos");
      if (acceptToS.checked == true) {
        acceptToS.nextElementSibling.nextElementSibling.
innerHTML = "";
        acceptToS.nextElementSibling.nextElementSibling.
setAttribute("class", "good-form-element");
        acceptToSConfirm = true;
      } else {
        acceptToS.nextElementSibling.nextElementSibling.
innerHTML = "You must accept our Terms Of Use!";
        acceptToS.nextElementSibling.nextElementSibling.
setAttribute("class", "bad-form-element");
        acceptToSConfirm = false;
      }
```

```
      let messageBox =
document.getElementById("message");
      if (messageBox.value.length > 450) {
        messageBox.nextElementSibling.innerHTML =
"Message can't be longer than 450 characters!";
        messageBox.nextElementSibling.setAttribute("clas
s", "bad-form-element");
        messageConfirm = false;
      } else if (messageBox.value.length > 0 &&
messageBox.value.length < 20) {
        messageBox.nextElementSibling.innerHTML =
"Message can't be smaller then 20 characters!";
        messageBox.nextElementSibling.setAttribute("clas
s", "bad-form-element");
        messageConfirm = false;
      } else {
        messageBox.nextElementSibling.innerHTML = "";
        messageBox.nextElementSibling.setAttribute("clas
s", "good-form-element");
        messageConfirm = true;
      }
      let addToCardLocal = jsonParse("addToCardList");
      if (fullNameConfirm && mailConfirm &&
postalCodeConfirm && streetAddressConfirm &&
acceptToSConfirm && messageConfirm && mailNotSent &&
addToCardLocal != null) {
        cardMessages("Mail sent!");
        mailNotSent = false;
      } else if (mailNotSent == false) {
        cardMessages("Mail already sent!");
      } else if (addToCardLocal == null) {
        cardMessages("Card is empty!");
      }
    });
  }
```

```
  function topDiscountProducts(data) {
    let filteredDiscount = data.filter(function (el) {
      return el.price.discount >= 10;
    });
    showProducts(filteredDiscount);
  }

  function aboutMePage() {
    let aboutMeLink =
document.querySelectorAll(`[href="#about-me.html"]`);
    for (let item of aboutMeLink) {
      item.addEventListener("click", () => {
        let bodyEl =
document.getElementsByTagName("body");
        let header =
document.getElementsByTagName("header");
        let aboutMeContainer =
document.querySelector("#about-me-container");
        if (aboutMeContainer == null) {
          header[0].innerHTML += `<section id="about-me-
container">
            <div id="about-me">
              <div id="about-me-left">
                <img src="assets/img/nikola-timotic.png"
alt="profile-img" />
                <div id="about-me-info">
                  <ul id="about-me-info-content">
                    <li>
                      <p>
                        Mail:
                        <a
href="mailto:nt&#46;timotic&#64;gmail&#46;com">nt&#46;ti
motic&#64;gmail&#46;com</a>
                      </p>
                    </li>
                    <li>
```

```
                    <p>
                      Phone:
                      <a href="tel:060-123-
4567">0601234567</a>
                    </p>
                  </li>
                  <li>
                    <p>Index: <span>173/20</span></p>
                  </li>
                </ul>
                <ul id="about-social">
                  <li>
                    <a
href="https://www.facebook.com">
                      <i class="fab fa-facebook-
square"></i>
                    </a>
                  </li>
                  <li>
                    <a
href="https://www.instagram.com">
                      <i class="fab fa-instagram-
square"></i>
                    </a>
                  </li>
                  <li>
                    <a
href="https://www.linkedin.com/in/nikolatimotic/">
                      <i class="fab fa-linkedin"></i>
                    </a>
                  </li>
                </ul>
              </div>
            </div>
            <div id="about-me-right">
              <h2>About Me</h2>
```

```
                <div id="about-me-content">
                  <p id="about-me-text">Hi, my name is
Nikola. I’m a Web Developer located in Serbia. I have a
passion for UI and UX design. I'm quietly confident,
naturally curious, and perpetually working on improving
my chops one design problem at a time. I'm a very well-
organized person and a problem solver.</p>
                  <div id="quote-box">
                    <cite id="quote"></cite>
                    <p id="author"></p>
                  </div>
                </div>
              </div>
              <div id="about-me-close">
                <div class="close cross" id="about-me-
exit"></div>
              </div>
            </div>
          </section>`;
          quoteGeneratorInterval =
setInterval(quoteGenerator, 2000);
          aboutMeModalRemoved = false;
        }
        // About me
        let aboutMeExit =
document.getElementById("about-me-close");
        aboutMeExit.addEventListener("click", () => {
          header[0].removeChild(header[0].lastChild);
          aboutMeModalRemoved = true;
          clearInterval(quoteGeneratorInterval);
        });
      });
    }
  }
  // Quote generator
  var random1;
```

```
  var random2;

  function quoteGenerator() {
    let quoteList = jsonParse("quotesLocal");
    if (!aboutMeModalRemoved) {
      random1 = Math.floor(Math.random() *
quoteList.length);
      while (random1 == random2) {
        random1 = Math.floor(Math.random() *
quoteList.length);
      }
      document.getElementById("quote").innerHTML =
quoteList[random1].quote;
      document.getElementById("author").innerHTML =
quoteList[random1].author;
      random2 = random1;
    } else {
      clearInterval(quoteGeneratorInterval);
    }
  }
  var burger = document.getElementById("nav-close-
container");
  burger.addEventListener("click", () => {
    document.getElementById("nav-
close").classList.toggle("cross");
    document.getElementById("navbar-
container").classList.toggle("nav-active");
  });
};
```

3.4 JSON

3.4.1 brands.json

```json
[
  {
    "id": 0,
    "name": "Logitech"
  },
  {
    "id": 1,
    "name": "Keychron"
  }
]
```

3.4.2 categories.json

```json
[
  {
    "id": 0,
    "name": "Membrane"
  },
  {
    "id": 1,
    "name": "Mechanical"
  },
  {
    "id": 2,
    "name": "Optical"
  }
]
```

3.4.3 connectionTypes.json

```json
[
```

```
  {
    "id": 0,
    "name": "Wired"
  },
  {
    "id": 1,
    "name": "Wireless"
  }
]
```

3.4.4 documentLinks.json

```
[
  {
    "id": 0,
    "name": "Documentation",
    "link": "documentation.pdf"
  },
  {
    "id": 1,
    "name": "Sitemap",
    "link": "sitemap.xml"
  },
  {
    "id": 2,
    "name": "RSS",
    "link": "rss.xml"
  },
  {
    "id": 3,
    "name": "Robots",
    "link": "robots.txt"
  }
]
```

3.4.5 navLinks.json

```
[
  {
    "id": 0,
    "name": "Home",
    "link": "index.html"
  },
  {
    "id": 1,
    "name": "Store",
    "link": "store.html"
  },
  {
    "id": 2,
    "name": "About us",
    "link": "about-us.html"
  },
  {
    "id": 3,
    "name": "About me",
    "link": "#about-me.html"
  },
  {
    "id": 4,
    "name": "Checkout",
    "link": "checkout.html",
    "icon": "fa-solid fa-basket-shopping"
  }
]
```

3.4.6 products.json

```
[
```

```
{
  "id": 0,
  "name": "Logitech Craft",
  "image": {
    "src": "assets/img/logitech-craft.png",
    "alt": "logitech craft keyboard"
  },
  "brandID": 0,
  "categoryID": 0,
  "connectionTypeID": 1,
  "price": {
    "old": 200,
    "discount": 15
  },
  "descript": "This keyboard have: ",
  "score": 4.8
},
{
  "id": 1,
  "name": "MX Keys",
  "image": {
    "src": "assets/img/logitech-mx-keys.png",
    "alt": "logitech mx keys keyboard"
  },
  "brandID": 0,
  "categoryID": 0,
  "connectionTypeID": 1,
  "price": {
    "old": 119.9,
    "discount": 10
  },
  "descript": "This keyboard have: ",
  "score": 4.7
},
{
  "id": 2,
```

```
    "name": "K845 Mechanical Illuminated",
    "image": {
      "src": "assets/img/k845-mechanical-
illuminated.png",
      "alt": "k845 mechanical illuminated keyboard"
    },
    "brandID": 0,
    "categoryID": 1,
    "connectionTypeID": 0,
    "price": {
      "old": 59.9,
      "discount": 0
    },
    "descript": "This keyboard have: ",
    "score": 4.3
  },
  {
    "id": 3,
    "name": "Q60 QMK",
    "image": {
      "src": "assets/img/Keychron-Q60-QMK.png",
      "alt": "Keychron Q60 QMK keyboard"
    },
    "brandID": 1,
    "categoryID": 1,
    "connectionTypeID": 1,
    "price": {
      "old": 219,
      "discount": 0
    },
    "descript": "This keyboard have: ",
    "score": 4.5
  },
  {
    "id": 4,
    "name": "Keychron K14",
```

```
    "image": {
      "src": "assets/img/keychron-K14.png",
      "alt": "Keychron Q60 QMK keyboard"
    },
    "brandID": 1,
    "categoryID": 1,
    "connectionTypeID": 1,
    "price": {
      "old": 59,
      "discount": 0
    },
    "descript": "This keyboard have: ",
    "score": 4.9
  }
]
```

3.4.7 quotes.json

```
[
  {
    "quote": "Knowledge is power.",
    "author": "Francis Bacon"
  },
  {
    "quote": "First, solve the problem. Then, write the
code.",
    "author": "John Johnson"
  },
  {
    "quote": "Code is like humor. When you have to
explain it, it’s bad.",
    "author": "Cory House"
  },
  {
```

```
    "quote": "Talk is cheap. Show me the code.",
    "author": "Linus Torvalds"
  },
  {
    "quote": "If you don’t love it, you’re going to
fail.",
    "author": "Steve Jobs"
  },
  {
    "quote": "Before software can be reusable it first
has to be usable.",
    "author": "Ralph Johnson"
  }
]
```

3.4.8 socialLinks.json

```
[
  {
    "id": 0,
    "name": "Facebook",
    "link": "https://www.facebook.com"
  },
  {
    "id": 1,
    "name": "Instagram",
    "link": "https://www.instagram.com"
  },
  {
    "id": 2,
    "name": "Linked in",
    "link": "https://www.linkedin.com/"
  },
  {
```

```
    "id": 3,
    "name": "Github",
    "link": "https://github.com/"
  }
]
```