

Visoka škola strukovnih studija za informacione i komunikacione tehnologije



# Dokumentacija

<http://dermacare-webstore.herokuapp.com/>

Marko Rusic 22/17

# Sadržaj

1. UVOD.....	4
1.1 Opis projekta.....	4
1.2 Korišćeni programski / markap jezici.....	4
1.3 Opis stanica.....	4
2. Layout.....	7
2.1 Header.....	7
2.2 Footer.....	9
2.3 Breadcrumbs.....	11
3. Markap pojedinačnih stranica.....	13
3.1 SEO.....	13
3.2 Početna strana.....	14
3.3 Strana kolekcija.....	15
3.4 Strana kolekcije.....	15
3.4 Strana proizvoda.....	16
3.5 Strana korpe.....	17
3.6 Kontakt strana.....	19
3.7 Strana o autoru.....	21
4. Stilizovanje HTML elemenata - CSS.....	22
4.1 Generalni stilovi.....	22
4.2 Layout.....	23
4.3 Specifični stilovi.....	23
4.4 CSS helperi.....	33
5. JavaScript.....	35
5.1 Entry fajl.....	35
5.2 app.js.....	36
5.3 Shared modul.....	37
5.3.1 EventEmmitter.js.....	38
5.3.2 FormValidation.js.....	39
5.3.3 pagination.js.....	41
5.3.4 toaster.js.....	42
5.3.4 url.js.....	44
5.3.5 View.js.....	45
5.3.6 wait.js.....	45
5.3.7 fakeHttp.js.....	46
5.4 Category modul.....	47

5.4.1 categoryService.js.....	47
5.4.2 view.js.....	48
5.4.3 index.js.....	48
5.5 Product modul.....	49
5.5.1 productService.js.....	50
5.5.2 ordering.js.....	51
5.5.3 view.js.....	52
5.5.4 index.js.....	53
5.6 Cart modul.....	54
5.6.1 config.js.....	54
5.6.2 view.js.....	55
5.6.3 store.js.....	56
5.6.4 index.js.....	58

# 1. UVOD

## 1.1 Opis projekta

“Dermacare” predstavlja web prodavnicu koja korisnicima pruža mogućnost da pregledaju i kupe nešto iz širokog assortimana prodavnice dermatoloških preparata. Preparati su rasporedjeni po kategorijama kako bi veliki broj potencijalnih klijenata mogao da pronađe nešto što želi da kupi.

Proizvode je po kategorijama moguće sortirati po datumu kreiranja i po ceni kako bi se potencijalnim kupcima omogućilo brzo pronalaženje proizvoda koji ima najviše odgovara.

Svaki proizvod sadrži detaljan opis, specifikaciju, način upotrebe i upozorenja o korišćenju. Pored toga, nalazi se i dugme koje dodaje proizvod u korpu. Kada korisnik završi sa odabirom proizvoda, odlazi da pregleda svoju korpu , popunjava formu za slanje paketa na željenu adresu i potvrđuje kupovinu.

## 1.2 Korišćeni programski / markap jezici

- Za izradu ovog sajta korišćeni su sledeće tehnologije:

- Markap jezici: HTML5
- Serverski jezici: PHP
- Skripting-programske jezici: JavaScript
- CSS3
- jQuery: JavaScript biblioteka
- Bootstrap: CSS framework

## 1.3 Opis stanica

### Zajedničko za sve stranice

- SEO - sadržaj title i meta tagova u zavisnosti od stranice, alt atributi na slikama, sitemap.xml, humans.txt, itd...
- Header sa linkovima ka drugim stranicama, svg ikonicom korpe sa dinamičkom vrednošću koja predstavlja broj proizvoda u korpi
- Footer sa opštim informacijama, social media linkovima, newsletter formom
- Breadcrumbs sekcija koja omogućava bolju/napredniju navigaciju kroz sajt
- Simulacija povlačenja mokapovanog sadržaja(kategorija i proizvoda) sa servera
- Validacija formi pomoću regularnih izraza i FormValidation utility klase
- Toaster za prikazivanje notifikacija korisniku

**Početna strana** (index.php)

- Prikaz izdvojenog sadržaja
- Dinamički prikaz proizvoda iz izdvojenih kategorija
- Testimonials sekcija sa slajderom

**Strana kolekcija** (collections.php)

- Dinamički prikaz svih kolekcija(kategorija proizvoda)

**Pojedinačna strana kolekcije** (collection.php)

- Dinamički prikaz proizvoda kategorije
- Prikaz količine proizvoda u kategoriji
- Mogućnost sortiranja proizvoda po ceni (opadajuće/rastuće) i datumu kreiranja
- Mogućnost parcijalnog prikazivanja proizvoda

**Pojedinačna strana prozvoda** (product.php)

- Detaljan prikaz proizvoda i njegovih atributa
- Mogućnost odabira količine
- Mogućnost dodavanja proizvoda u korpu

**Strana korpa** (cart.php)

- Prikaz proizvoda koji se trenutno nalaze u korpi
- Mogućnost izmene količine porizvoda
- Mogućnost brisanja proizvoda i korpe
- Dinamičko računanje subtotal i total vrednosti korpe
- Mogućnost kupovine, tj finalni korak koji se realizuje popunjavanjem Order forme

**Kontakt strana** (contact.php)

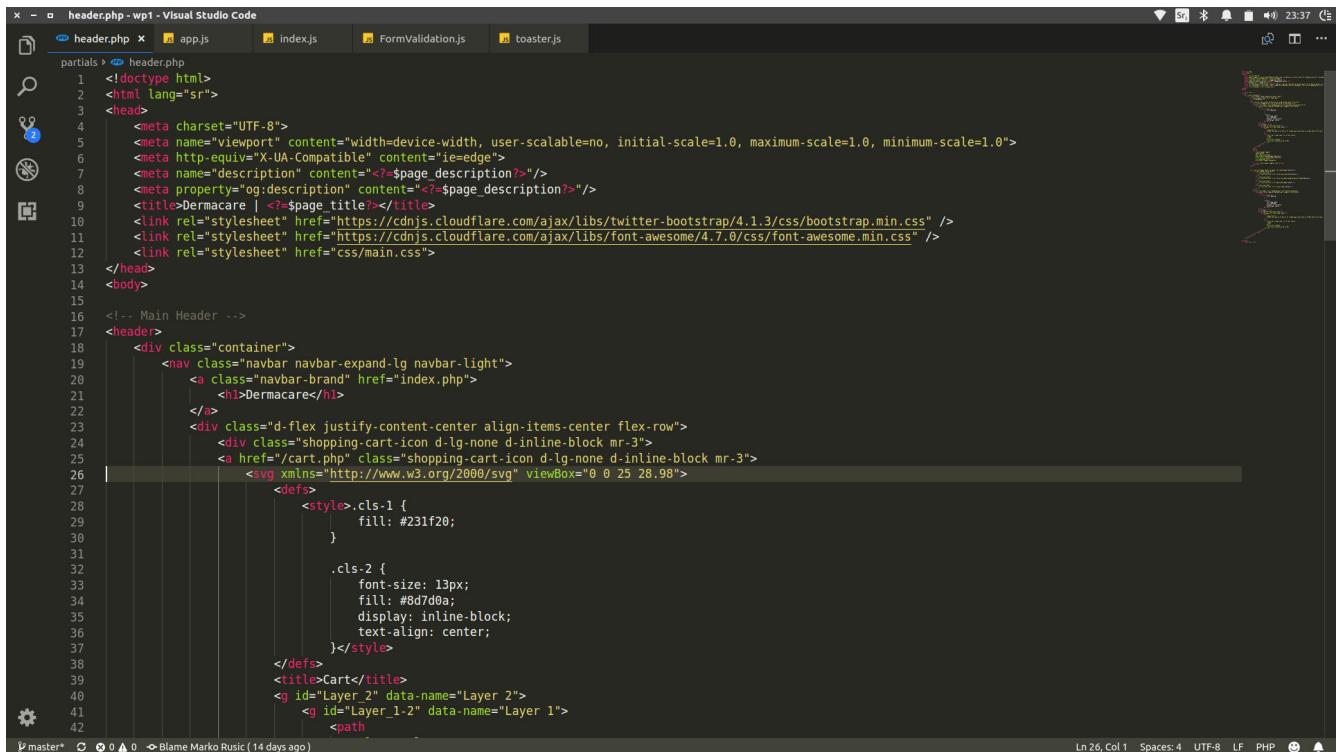
- Kontakt forma sa standardnim poljima
- Opšti podaci o sajtu

**Strana o autoru** (about.php)

- Opšte informacije o autoru
- Korisni linkovi (link do repozitorijuma sajta)

## 2. Layout

### 2.1 Header



A screenshot of the Visual Studio Code interface showing the `header.php` file. The code is a PHP template for a header, including meta tags, a title, and a navigation bar. The code editor shows syntax highlighting for PHP and CSS. A sidebar on the left contains icons for search, file operations, and other code-related functions. The status bar at the bottom right indicates the current file is `header.php`, the line number is 26, column 1, and the file type is PHP.

```
<!DOCTYPE html>
<html lang="sr">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, user-scalable=no, initial-scale=1.0, maximum-scale=1.0, minimum-scale=1.0">
    <meta http-equiv="X-UA-Compatible" content="ie=edge">
    <meta name="description" content=<?=$page_description?>/>
    <meta property="og:description" content=<?= $page_description ?>/>
    <title>Dermacare | <?= $page_title ?></title>
    <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/twitter-bootstrap/4.1.3/css/bootstrap.min.css" />
    <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/4.7.0/css/font-awesome.min.css" />
    <link rel="stylesheet" href="css/main.css">
</head>
<body>
<!-- Main Header -->
<header>
    <div class="container">
        <nav class="navbar navbar-expand-lg navbar-light">
            <a class="navbar-brand" href="index.php">
                <h1>Dermacare</h1>
            </a>
            <div class="d-flex justify-content-center align-items-center flex-row">
                <div class="shopping-cart-icon d-lg-none d-inline-block mr-3">
                    <a href="#">
                        <img alt="Cart icon" data-name="Layer 1" data-id="Layer_1-2" data-layer="Layer 1" style="width: 25px; height: 25px; border-radius: 50%; border: 1px solid #ccc; vertical-align: middle; margin-right: 5px;" />
                        <span>Cart</span>
                    </a>
                </div>
                <div class="d-lg-none d-inline-block mr-3">
                    <img alt="Cart icon" data-name="Layer 2" data-id="Layer_2" data-layer="Layer 2" style="width: 25px; height: 25px; border-radius: 50%; border: 1px solid #ccc; vertical-align: middle; margin-right: 5px;" />
                    <span>Cart</span>
                </div>
            </div>
        </nav>
    </div>
</header>
```

```
header.php - wpt - Visual Studio Code
header.php x app.js index.js FormValidation.js toaster.js
partials > header.php
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
        class="cls-1"
        d="M23.78,3.95a.57.57,0,0,0-.57-.54H18.4a.6.6.8,0,0,0-11.79,0H1.79a.57.57,0,0,0-.57.54L0,28.38a.57.57,0,0,0,.57.6H24.43a.57
      >
      <text
        class="cart-items-count cls-2 text-center"
        x="40%"
        y="9"
        text-anchor="middle"
        transform="translate(1.82 21.99)"
      ></text>
    </g>
  </svg>
</a>
</div>
<button
  class="navbar-toggler"
  type="button"
  data-toggle="collapse"
  data-target="#navbarSupportedContent"
  aria-controls="navbarSupportedContent"
  aria-expanded="false"
  aria-label="Toggle navigation"
>
  <span class="navbar-toggler-icon"></span>
</button>
</div>

<div class="collapse navbar-collapse" id="navbarSupportedContent">
  <ul class="navbar-nav mr-auto">
    <li class="nav-item">
      <a class="nav-link" href="collections.php">Collections</a>
    </li>
    <li class="nav-item">
      <a class="nav-link" href="collection.php?slug=glow-edition">Glow Edition</a>
    </li>
    <li class="nav-item">
      <a class="nav-link" href="about.php">About author</a>
    </li>
    <li class="nav-item">
      <a class="nav-link" href="contact.php">Contact</a>
    </li>
  </ul>
</div>
```

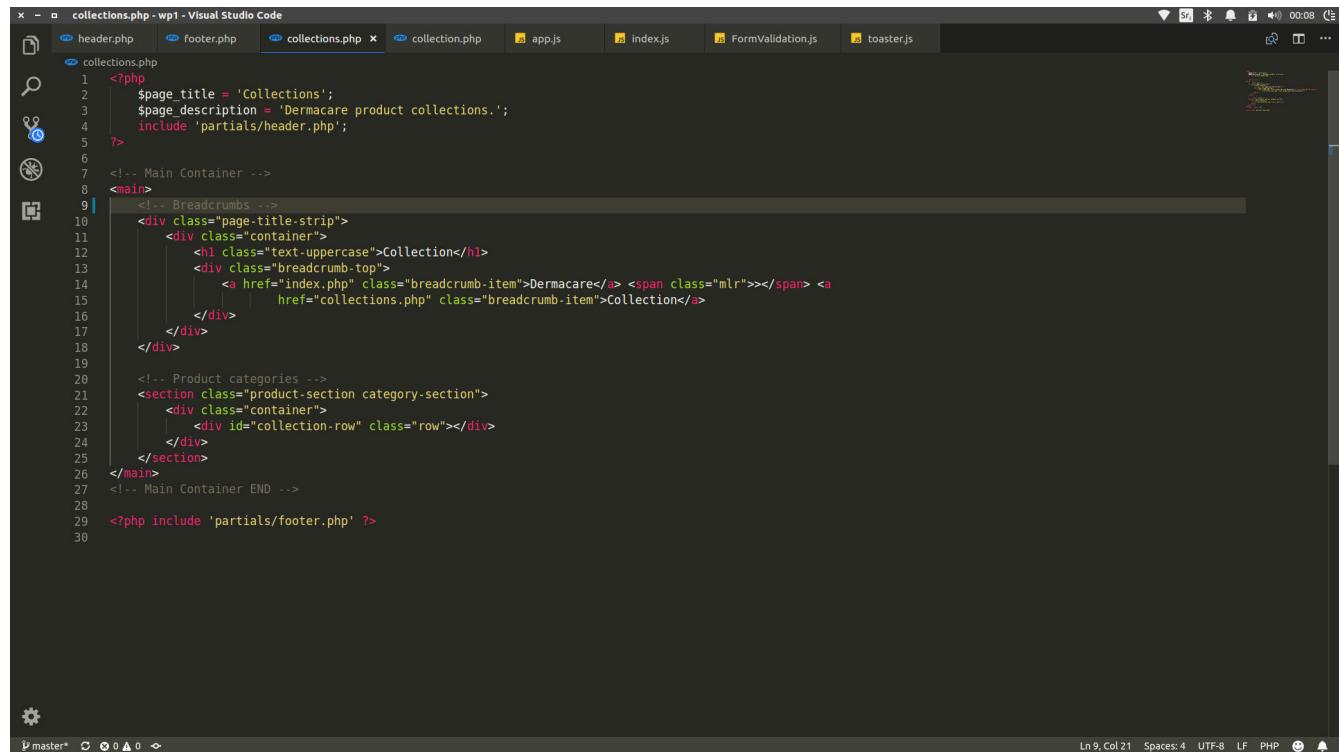
## 2.2 Footer

```
 3 <footer>
 4     <div class="container">
 5         <div class="row">
 6             <div class="col-md-4">
 7                 <div>
 8                     <div class="heading-wrap text-center">
 9                         | <span>Dermacare</span>
10                     </div>
11
12                     <p class="text-center">
13                         <span class="mb-1 d-block">Adresa 213a, Beograd</span>
14                         <span><a href="mailto:marko.rusic.22.17@ict.edu.rs" style="color:white;">marko.rusic.22.17@ict.edu.rs</a></span>
15                     </p>
16                 </div>
17             </div>
18
19             <div class="col-md-4">
20                 <div>
21                     <div class="heading-wrap text-center">
22                         | <h5 class="d-inline-block bg-white text-uppercase">Follow us on social meida</h5>
23                     </div>
24
25                     <ul class="list-reset text-center">
26                         <li class="d-inline-block mr-40"><a href="#"></a></li>
27                         <li class="d-inline-block mr-40"><a href="#"></a></li>
28                         <li class="d-inline-block"><a href="#"></a></li>
29                     </ul>
30                 </div>
31             </div>
32
33             <div class="col-md-4">
34                 <div>
35                     <div class="heading-wrap text-center">
36                         | <h5 class="d-inline-block bg-white text-uppercase">Sing up for our Newsletter</h5>
37                     </div>
38
39                     <form id="newsletter-form">
40                         <input
41                             type="text"
42                             placeholder="Enter your email address"
43                             data-validate
44                             data-required="Email is required" />

```

```
header.php  footer.php x app.js index.js FormValidation.js toaster.js
partials > footer.php
36
37         <form id="newsletter-form">
38             <input
39                 type="text"
40                 placeholder="Enter your email address"
41                 data-validate
42                 data-required="Email is required"
43                 data-pattern="\S+@\S+\.\S+"
44                 data-pattern-message="Invalid email format"
45             />
46             <button></button>
47         </form>
48     </div>
49 </div>
50 </div>
51 </div>
52 </div>
53 </div>
54 <div class="copyright">
55     <div class="container text-center">
56         <div class="text-center">
57             <small class="text-center">
58                 Dermacare® <?= date("Y") ?>.
59             </small>
60             <p class="text-center">Site designed & developed by <a href="about.php">Marko Rusic</a></p>
61         </div>
62     </div>
63 </div>
64 </div>
65 <!-- Main Footer END -->
66 <script src="https://code.jquery.com/jquery-3.3.1.min.js" integrity="sha256-FgCb/KJQLNf0u91ta32o/NMZxltwRo8QtmkMRdAu8=" crossorigin="anonymous"></script>
67 <script src="https://cdnjs.cloudflare.com/ajax/libs/twitter-bootstrap/4.1.3/js/bootstrap.bundle.min.js"></script>
68 <!-- Repozitorijum ovog sajta: https://github.com/markorusic/wpl -->
69 <script src="js/dist/index.js"></script>
70 </body>
71 </html>
```

## 2.3 Breadcrumbs



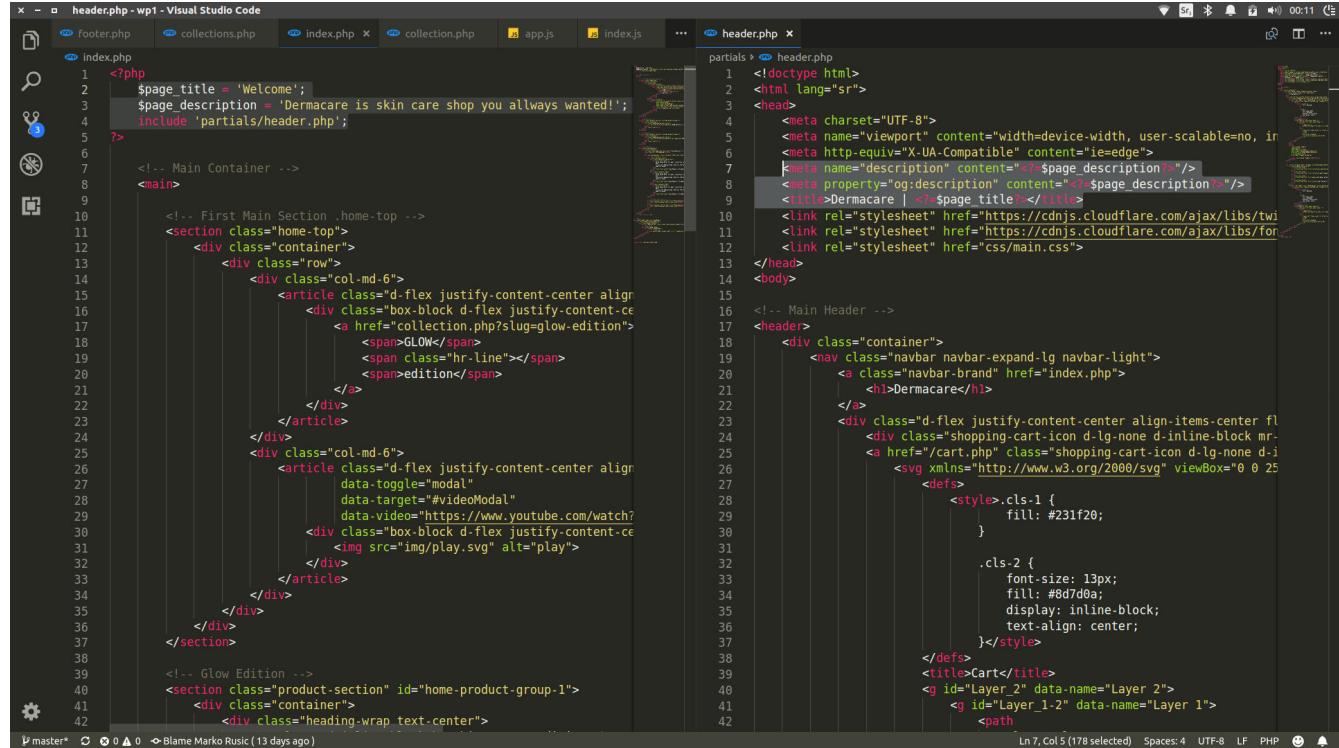
```
<?php
    $page_title = 'Collections';
    $page_description = 'DermaCare product collections.';
    include 'partials/header.php';
?>

<!-- Main Container -->
<main>
    <!-- Breadcrumbs -->
    <div class="page.title.strip">
        <div class="container">
            <h1 class="text-uppercase">Collection</h1>
            <div class="breadcrumb-top">
                <a href="index.php" class="breadcrumb-item">DermaCare</a> <span class="mlr">>></span> <a
                |     href="collections.php" class="breadcrumb-item">Collection</a>
            </div>
        </div>
    </div>
    <!-- Product categories -->
    <section class="product-section category-section">
        <div class="container">
            <div id="collection-row" class="row"></div>
        </div>
    </section>
</main>
<!-- Main Container END -->
<?php include 'partials/footer.php' ?>
```

# 3. Markap pojedinačnih stranica

## 3.1 SEO

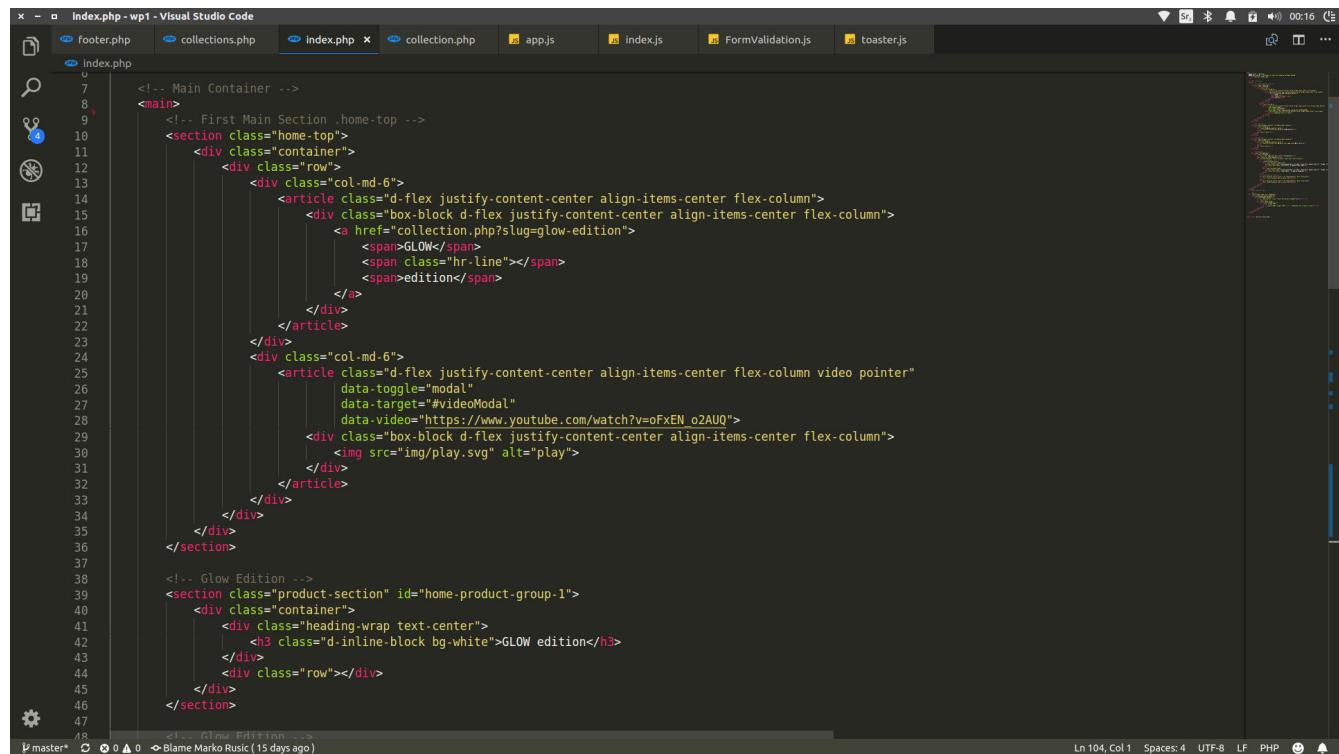
Svaka stranica sadrži title i description koji se prikazuje u headeru sajta



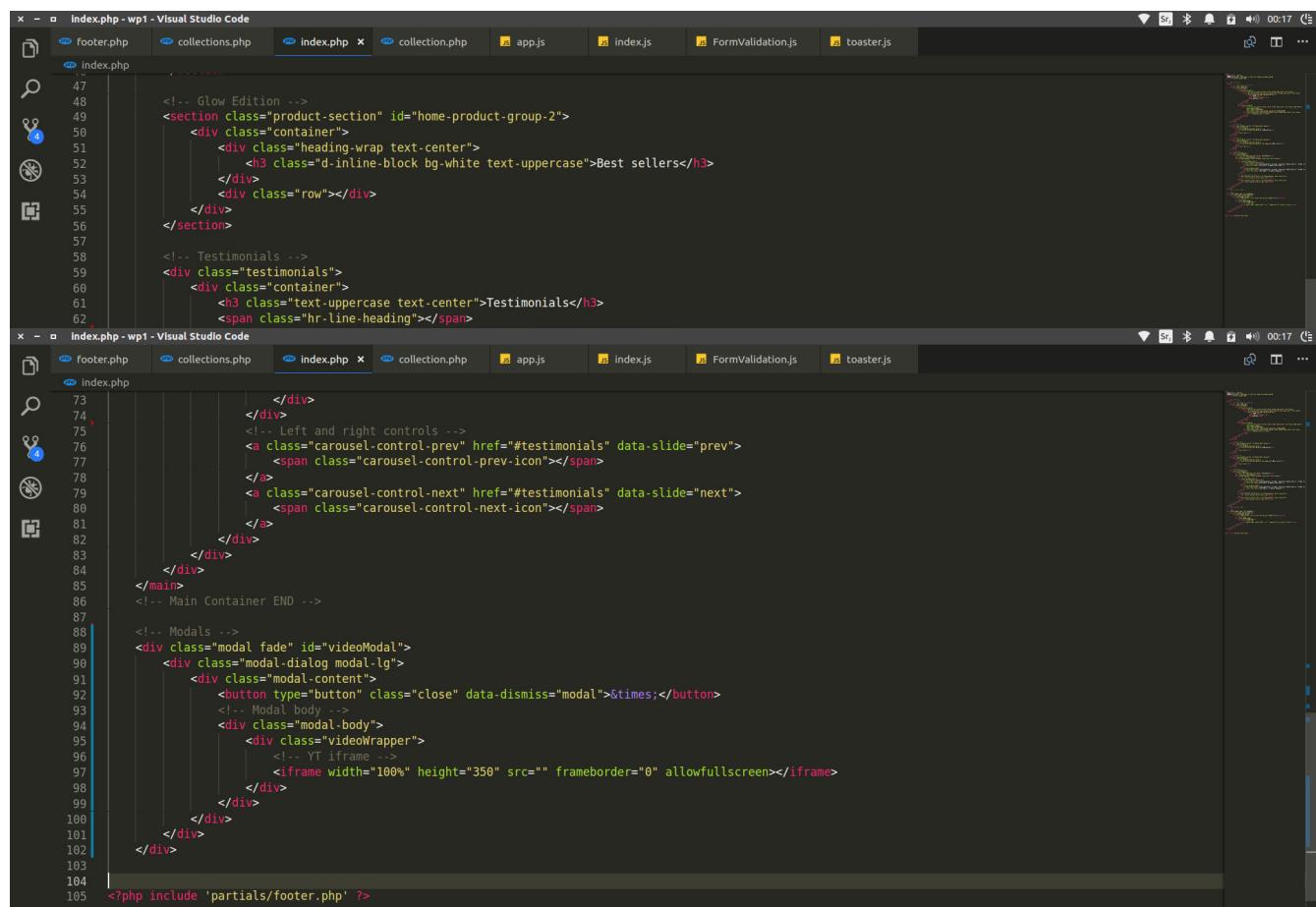
```
header.php - wp1 - Visual Studio Code
index.php
header.php

1 <?php
2     $page_title = 'Welcome';
3     $page_description = 'Dermacare is skin care shop you always wanted!';
4     include 'partials/header.php';
5   ?>
6
7   <!-- Main Container -->
8   <main>
9
10    <!-- First Main Section .home-top -->
11    <section class="home-top">
12      <div class="container">
13        <div class="row">
14          <div class="col-md-6">
15            <article class="d-flex justify-content-center align-items-center">
16              <a href="collection.php?slug=glow-edition">
17                <span>GLOW</span>
18                <span class="hr-line"></span>
19                <span>edition</span>
20              </a>
21            </div>
22          </article>
23        </div>
24        <div class="col-md-6">
25          <article class="d-flex justify-content-center align-items-center">
26            <a href="#" data-toggle="modal" data-target="#videoModal" data-video="https://www.youtube.com/watch?v=..."/>
27            <div class="box-block d-flex justify-content-center align-items-center">
28              
29            </div>
30          </article>
31        </div>
32      </div>
33    </section>
34
35    <!-- Glow Edition -->
36    <section class="product-section" id="home-product-group-1">
37      <div class="container">
38        <div class="heading-wrap text-center">
39
40          <meta charset="UTF-8">
41          <meta name="viewport" content="width=device-width, user-scalable=no, initial-scale=1.0, maximum-scale=1.0, minimum-scale=1.0">
42          <meta http-equiv="X-UA-Compatible" content="ie=edge">
43          <meta name="description" content="{$page_description}">
44          <meta property="og:description" content="{$page_description}">
45          <title>Dermacare | {$page_title}</title>
46          <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/twbs/bootstrap/4.5.2/css/bootstrap.min.css">
47          <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/5.15.4/css/all.min.css">
48          <link rel="stylesheet" href="css/main.css">
49
50        </div>
51      </div>
52    </section>
53
54    <!-- Main Header -->
55    <header>
56      <div class="container">
57        <nav class="navbar navbar-expand-lg navbar-light">
58          <a class="navbar-brand" href="index.php">
59            <h1>Dermacare</h1>
60          </a>
61          <div class="d-flex justify-content-center align-items-center flex-grow-1">
62            <div class="shopping-cart-icon d-lg-none d-inline-block mr-2">
63              <a href="#" class="shopping-cart-icon d-lg-none d-inline-block mr-2">
64                
65                <span>Cart</span>
66              </a>
67            </div>
68            <div class="def">
69              <style>.cls-1 { fill: #231f20; } .cls-2 { font-size: 13px; fill: #8d7d0a; display: inline-block; text-align: center; }</style>
70            </div>
71          </div>
72          <title>Cart</title>
73          <g id="Layer_2" data-name="Layer 2">
74            <g id="Layer_1-2" data-name="Layer 1">
75              <path>
76            </g>
77          </g>
78        </nav>
79      </div>
80    </header>
81
82    <!-- Main Content -->
83    <div class="main-content">
84      <div class="container">
85        <div class="row">
86          <div class="col-md-4">
87            <div class="product-item">
88              
89              <div class="product-info">
90                <h3>Product 1</h3>
91                <p>Description for Product 1</p>
92                <button>Buy Now</button>
93              </div>
94            </div>
95          </div>
96          <div class="col-md-4">
97            <div class="product-item">
98              
99              <div class="product-info">
100                <h3>Product 2</h3>
101                <p>Description for Product 2</p>
102                <button>Buy Now</button>
103              </div>
104            </div>
105          </div>
106          <div class="col-md-4">
107            <div class="product-item">
108              
109              <div class="product-info">
110                <h3>Product 3</h3>
111                <p>Description for Product 3</p>
112                <button>Buy Now</button>
113              </div>
114            </div>
115          </div>
116        </div>
117      </div>
118    </div>
119
120    <!-- Footer -->
121    <footer>
122      <div class="container">
123        <div class="row">
124          <div class="col-md-4">
125            <h3>Footer Content</h3>
126            <p>Some footer text here</p>
127          </div>
128          <div class="col-md-4">
129            <h3>Footer Content</h3>
130            <p>Some footer text here</p>
131          </div>
132          <div class="col-md-4">
133            <h3>Footer Content</h3>
134            <p>Some footer text here</p>
135          </div>
136        </div>
137      </div>
138    </footer>
139
140    <script src="https://code.jquery.com/jquery-3.5.1.slim.min.js" integrity="sha384-DfXdz2htPH0lsSSs5Nort4zHO1Jkq1uO+mgfZQ9F7Gy6t7P1hUpvE8YQJ5xa" crossorigin="anonymous"></script>
141    <script src="https://cdn.jsdelivr.net/npm/@popperjs/core@2.5.4/dist/umd/popper.min.js" integrity="sha384-q2kxW9yNMh2VzfgJ0HdCeD2OmdKuTl0qHNfjlBipTgq33P8tW1Og6L+jn+Q" crossorigin="anonymous"></script>
142    <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/js/bootstrap.min.js" integrity="sha384-...></script>
143
144    <script>
145      $(document).ready(function() {
146        // Your JavaScript code here
147      });
148    </script>
149
150  </body>
151
152</html>
```

## 3.2 Početna strana



```
<!-- Main Container -->
<main>
    <!-- First Main Section .home-top -->
    <section class="home-top">
        <div class="container">
            <div class="row">
                <div class="col-md-6">
                    <article class="d-flex justify-content-center align-items-center flex-column">
                        <div class="box-block d-flex justify-content-center align-items-center flex-column">
                            <a href="collection.php?slug=glow-edition">
                                <span>GLOW</span>
                                <span class="hr-line"></span>
                                <span>edition</span>
                            </a>
                        </div>
                    </article>
                </div>
                <div class="col-md-6">
                    <article class="d-flex justify-content-center align-items-center flex-column video pointer" data-toggle="modal" data-target="#videoModal" data-video="https://www.youtube.com/watch?v=oFxEN_o2AUQ">
                        <div class="box-block d-flex justify-content-center align-items-center flex-column">
                            
                        </div>
                    </article>
                </div>
            </div>
        </div>
    </section>
    <!-- Glow Edition -->
    <section class="product-section" id="home-product-group-1">
        <div class="container">
            <div class="heading-wrap text-center">
                <h3 class="d-inline-block bg-white">GLOW edition</h3>
            </div>
            <div class="row"></div>
        </div>
    </section>
```



```
<!-- Glow Edition -->
<section class="product-section" id="home-product-group-2">
    <div class="container">
        <div class="heading-wrap text-center">
            <h3 class="d-inline-block bg-white text-uppercase">Best sellers</h3>
        </div>
        <div class="row"></div>
    </div>
</section>

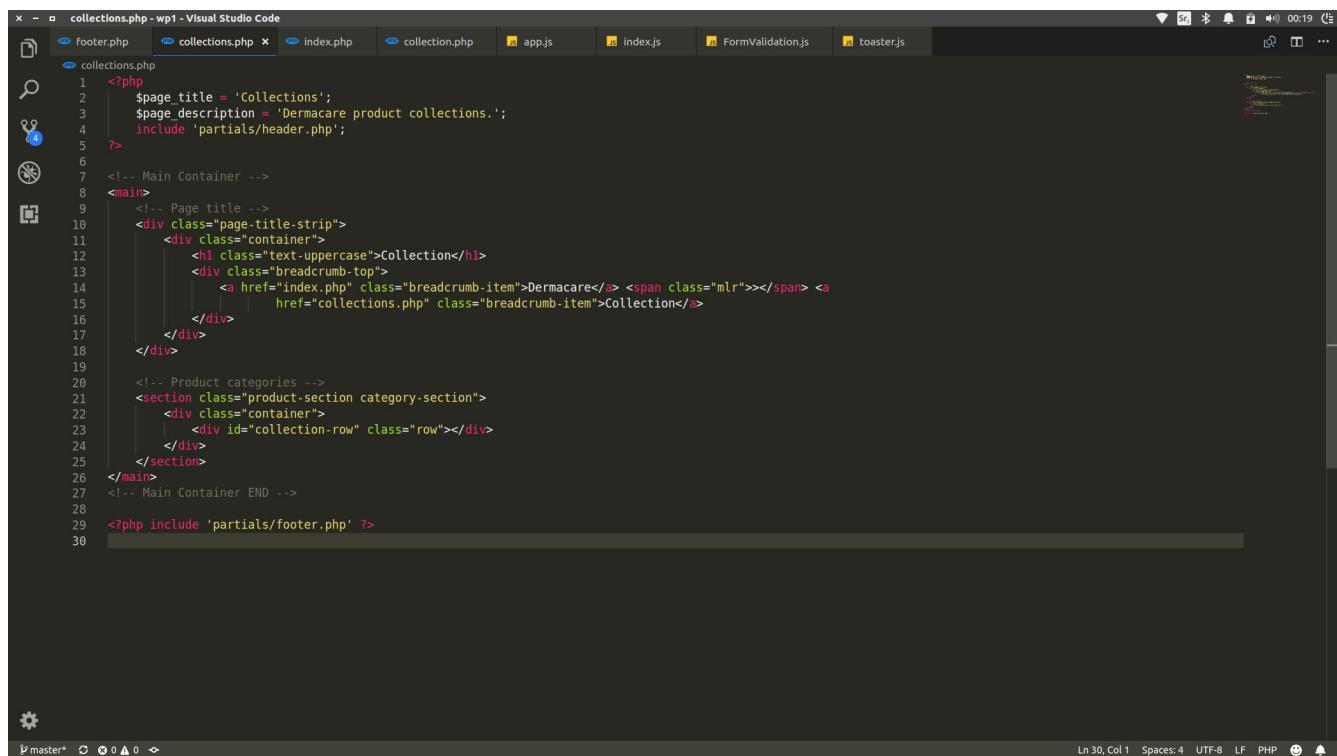
<!-- Testimonials -->
<div class="testimonials">
    <div class="container">
        <h3 class="text-uppercase text-center">Testimonials</h3>
        <span class="hr-line-heading"></span>
    </div>
</div>

<!-- Main Container END -->
<div class="modal fade" id="videoModal">
    <div class="modal-dialog modal-lg">
        <div class="modal-content">
            <button type="button" class="close" data-dismiss="modal">&times;</button>
            <!-- Modal body -->
            <div class="modal-body">
                <div class="videoWrapper">
                    <!-- YT iframe -->
                    <iframe width="100%" height="350" src="" frameborder="0" allowfullscreen></iframe>
                </div>
            </div>
        </div>
    </div>
</div>

<?php include 'partials/footer.php' ?>
```

### 3.3 Strana kolekcija

Kolekcije se prikazuju dinamički kroz JavaScript



The screenshot shows the Visual Studio Code interface with the file "collections.php" open. The code is a PHP script with some HTML and CSS. It includes a header section with page title and description, a main container with a breadcrumb navigation, and a product categories section. The code uses classes like "page-title-strip", "text-uppercase", "product-section", and "category-section". It also includes a "collection-row" div with an ID. The code ends with an include statement for "partials/footer.php". The status bar at the bottom shows the file is in master branch, has 0 changes, and is in PHP mode.

```
<?php
    $page_title = 'Collections';
    $page_description = 'Dermacare product collections.';
    include 'partials/header.php';
?>

<!-- Main Container -->
<main>
    <!-- Page title -->
    <div class="page-title-strip">
        <div class="container">
            <h1 class="text-uppercase">Collection</h1>
            <div class="breadcrumb-top">
                <a href="index.php" class="breadcrumb-item">Dermacare</a> <span class="mlr">>></span> <a
                    href="collections.php" class="breadcrumb-item">Collection</a>
            </div>
        </div>
    </div>
    <!-- Product categories -->
    <section class="product-section category-section">
        <div class="container">
            <div id="collection-row" class="row"></div>
        </div>
    </section>
</main>
<!-- Main Container END -->
<?php include 'partials/footer.php' ?>
```

### 3.4 Strana kolekcije

Proizvodi se prikazuju dinamički kroz JavaScript

Screenshot of Visual Studio Code showing the file `collection.php`. The code is a PHP template for a product collection page. It includes sections for the main container, page title, breadcrumb navigation, product filtering, and a product list section. The product list section contains a 'load more' button.

```

10     <!-- Main Container -->
11     <main>
12         <!-- Page title -->
13         <div class="page-title-strip">
14             <div class="container">
15                 <h1 class="text-uppercase"><?= $category['name'] ?></h1>
16                 <div class="breadcrumb-top">
17                     <a href="index.php" class="breadcrumb-item">Dermacare</a>
18                     <span class="mlr">></span>
19                     <a href="collection.php?slug=<?= $category['slug'] ?>" class="breadcrumb-item"><?= $category['name'] ?></a>
20                 </div>
21             </div>
22         </div>
23         <div class="container">
24             <div class="product-filter d-flex flex-column justify-content-md-between flex-md-row num-of-showed-products">
25                 <>>Show <span class="currently-showing"></span> of total <span class="total"></span> products</p>
26             </div>
27             <div>
28                 <form class="product-ordering">
29                     <select name="orderby" class="orderby">
30                         <option value="date" selected="selected">Sort by creation date</option>
31                         <option value="price">Sort by price: ascending</option>
32                         <option value="price-desc">Sort by price: descending</option>
33                     </select>
34                 </form>
35             </div>
36         </div>
37     </div>
38     </div>
39     <!-- Product list -->
40     <section class="product-section">
41         <div class="container">
42             <div class="row" id="product-list"></div>
43             <div class="heading-wrap text-center lmb-wrap" style="display: none;">
44                 <h3 class="d-inline-block bg-white"><a id="loadMore" class="load-more-btn">Show more</a></h3>
45             </div>
46         </div>
47     </section>
48 </main>
49 <!-- Main Container END -->

```

## 3.4 Strana proizvoda

Screenshot of Visual Studio Code showing the file `product.php`. This file displays a single product's details. It shows the product's name, category, price, and a form for adding it to the cart. The cart addition form includes fields for quantity and a submit button.

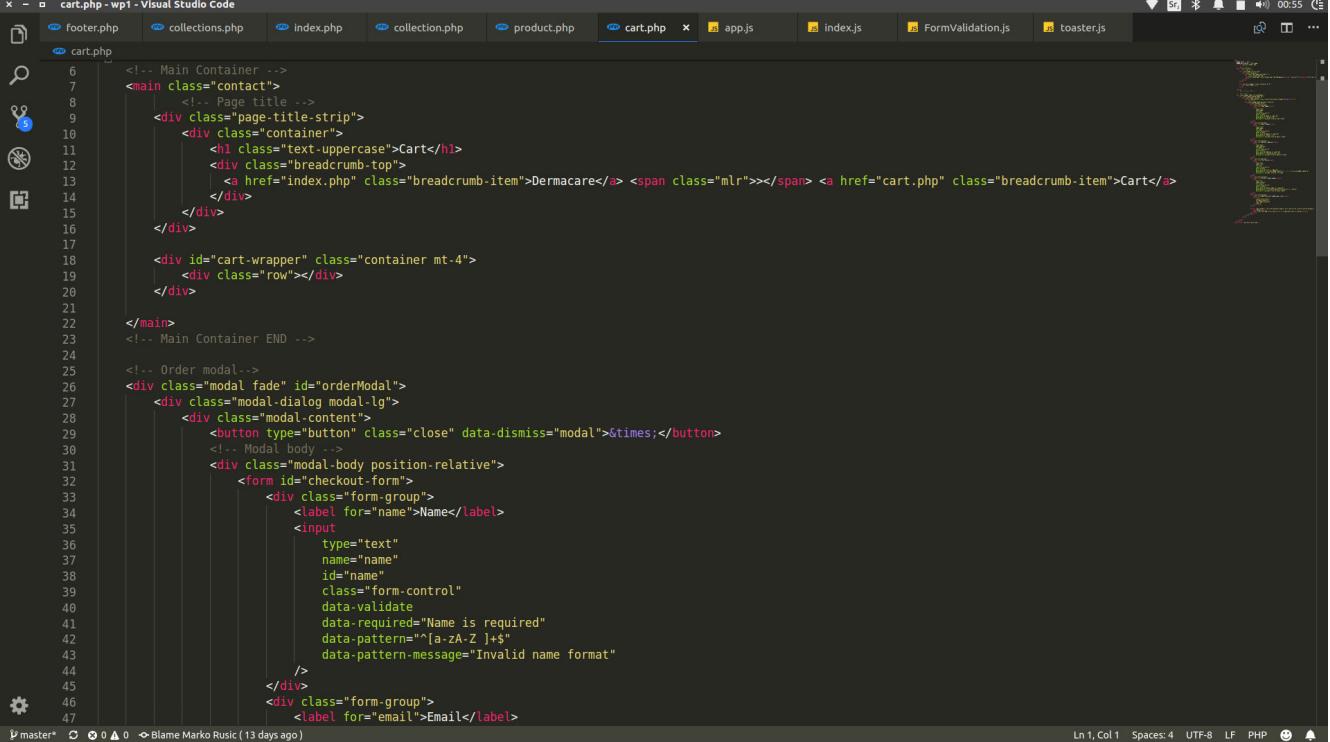
```

18     <!-- Main Container -->
19     <main>
20         <!-- Page title -->
21         <div class="page-title-strip">
22             <div class="container">
23                 <h1 class="text-uppercase"><?= $product['name'] ?></h1>
24                 <div class="breadcrumb-top">
25                     <a href="index.php" class="breadcrumb-item">Dermacare</a>
26                     <span class="mlr">></span>
27                     <a href="collection.php?slug=<?= $product['category']['slug'] ?>" class="breadcrumb-item"><?= $product['category']['name'] ?></a>
28                     <span class="mlr">></span>
29                     <a href="product.php?slug=<?= $product['slug'] ?>" class="breadcrumb-item"><?= $product['name'] ?></a>
30                 </div>
31             </div>
32         </div>
33     </div>
34
35     <div class="col-md-6">
36         <div class="d-flex justify-content-between">
37             <h2 class="product-price"><?= $product['price'] ?> <span class="currency"></span></h2>
38         </div>
39         <hr>
40         <div class="add-to-cart-block">
41             <form action="">
42                 <input id="add-to-cart-quantity" class="input-number" type="number" step="1" min="1" name="quantity" value="1">
43                 <div class="d-inline-block">
44                     <button data-product-id=<?= $product['id'] ?> type="submit" class="btn-add-to-cart d-flex justify-content-between">
45                         <span>Add to cart</span>
46                         <span class="btn-add-to-cart-plus"></span>
47                     </button>
48                 </div>
49             </form>
50         </div>
51         <hr>
52         <div class="product-description">
53             <h4 class="mt-4 mb-4">Product description</h4>
54             <p><?= $product['description'] ?></p>
55         </div>
56     </div>
57     </div>
58     </div>
59     </div>
60     </main>
61     <!-- Main Container END -->
62
63     <?php include 'partials/footer.php' ?>
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80

```

## 3.5 Strana korpe

Sadržaj korpe se prikazuje dinamički. Checkout forma se dinamički validira kroz JavaScript na osnovu data atributa na svakom polju.



```
<!-- Main Container -->
<main class="contact">
    <!-- Page title -->
    <div class="page-title-strip">
        <div class="container">
            <h1 class="text-uppercase">Cart</h1>
            <div class="breadcrumb-top">
                | <a href="index.php" class="breadcrumb-item">Dermacare</a> <span class="mlr">></span> <a href="cart.php" class="breadcrumb-item">Cart</a>
            </div>
        </div>
    </div>

    <div id="cart-wrapper" class="container mt-4">
        <div class="row">
    </div>
</main>
<!-- Main Container END -->

<!-- Order modal-->
<div class="modal fade" id="orderModal">
    <div class="modal-dialog modal-lg">
        <div class="modal-content">
            <button type="button" class="close" data-dismiss="modal">&times;</button>
            <!-- Modal body -->
            <div class="modal-body position-relative">
                <form id="checkout-form">
                    <div class="form-group">
                        <label for="name">Name</label>
                        <input
                            type="text"
                            name="name"
                            id="name"
                            class="form-control"
                            data-validate
                            data-required="Name is required"
                            data-pattern="^[\w\-\s]{2,}$"
                            data-pattern-message="Invalid name format"
                        />
                    </div>
                    <div class="form-group">
                        <label for="email">Email</label>
                    </div>
                </form>
            </div>
        </div>
    </div>
</div>
```

cart.php - wp1 - Visual Studio Code

```
46 <div class="form-group">
47   <label for="email">Email</label>
48   <input
49     type="text"
50     name="email"
51     id="email"
52     class="form-control"
53     data-validate
54     data-required="Email is required"
55     data-pattern="^[_@a-zA-Z0-9.-]+\.[_a-zA-Z0-9.-]+$"
56     data-pattern-message="Invalid email format"
57   />
58 </div>
59 <div class="form-group">
60   <label for="address">Address</label>
61   <input
62     type="text"
63     name="address"
64     id="address"
65     class="form-control"
66     data-validate
67     data-required="Address is required"
68     data-pattern="^[_0-9a-zA-Z.,-]+$"
69     data-pattern-message="Invalid address format"
70   />
71 </div>
72 <div class="form-group">
73   <label for="city">City</label>
74   <input
75     type="text"
76     name="city"
77     id="city"
78     class="form-control"
79     data-validate
80     data-required="City is required"
81     data-pattern="^([a-zA-Z\u0080-\u024F]+(?:\.\s|-|\s|'))*[a-zA-Z\u0080-\u024F]*$"
82     data-pattern-message="Invalid city name"
83   />
84 </div>
85 <div class="form-group">
86   <label for="contact">Phone number</label>
87   <input
88     type="tel"
89     name="contact"
90     id="contact"
91     class="form-control"
92     data-validate
93     data-required="Phone number is required"
94     data-pattern="^[-+]?[()0-9][0-9]{1,4}[()0-9][-.\s/.0-9]*$"
95     data-pattern-message="Invalid phone number"
96   />
97 </div>
98 <div class="form-group">
99   <label for="additionalInfo">Additional info</label>
100  <textarea
101    class="form-control"
102    id="additionalInfo"
103    name="additionalInfo"
104    rows="3"
105  ></textarea>
106 </div>
107 <button type="submit" class="btn-derma btn-add-to-cart-look d-flex justify-content-between">
108   <span>Order</span>
109   <span class="btn-add-to-cart-plus"></span>
110 </button>
111 </form>
112 </div>
113 </div>
114 </div>
115 </div>
116 </div>
117 <?php include 'partials/footer.php' ?>
```

Ln 1, Col 1 Spaces: 4 UTF-8 LF PHP 🎉

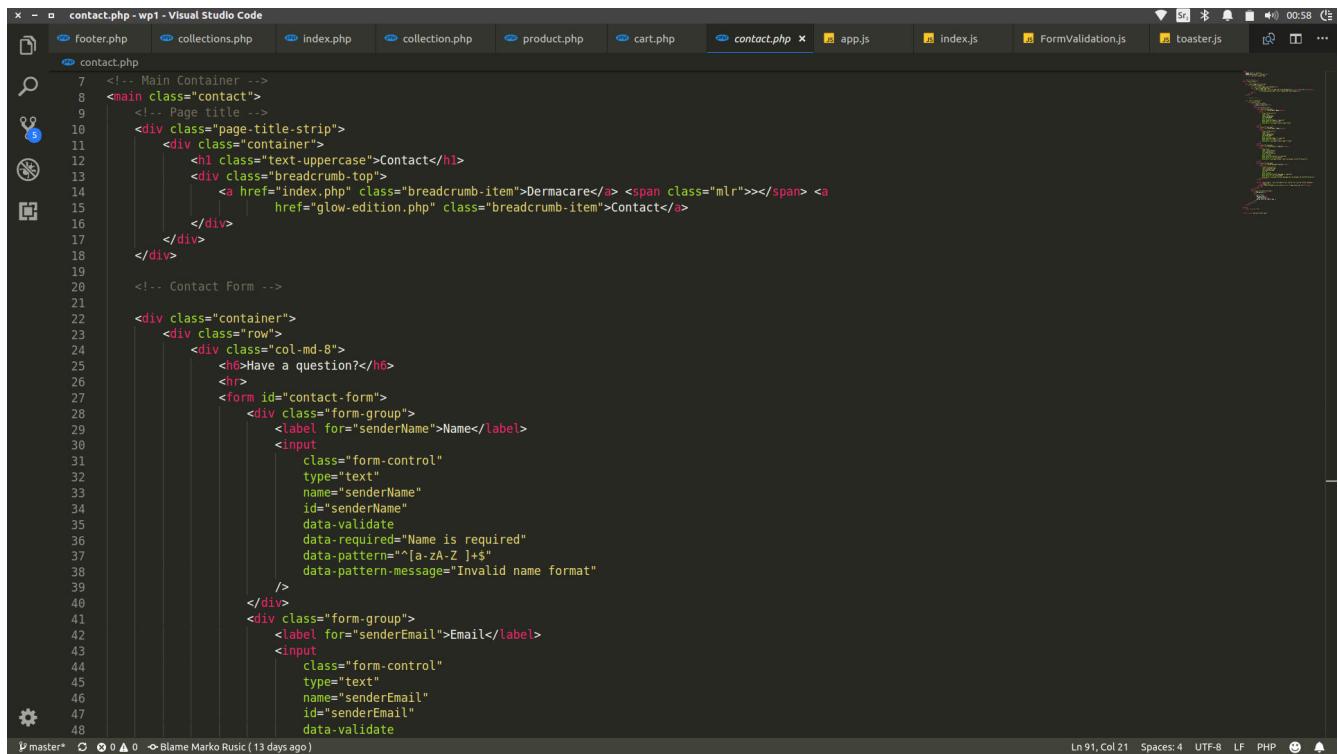
cart.php - wp1 - Visual Studio Code

```
80   data-validate
81   data-required="City is required"
82   data-pattern="^([a-zA-Z\u0080-\u024F]+(?:\.\s|-|\s|'))*[a-zA-Z\u0080-\u024F]*$"
83   data-pattern-message="Invalid city name"
84 </div>
85 <div class="form-group">
86   <label for="contact">Phone number</label>
87   <input
88     type="tel"
89     name="contact"
90     id="contact"
91     class="form-control"
92     data-validate
93     data-required="Phone number is required"
94     data-pattern="^[-+]?[()0-9][0-9]{1,4}[()0-9][-.\s/.0-9]*$"
95     data-pattern-message="Invalid phone number"
96   />
97 </div>
98 <div class="form-group">
99   <label for="additionalInfo">Additional info</label>
100  <textarea
101    class="form-control"
102    id="additionalInfo"
103    name="additionalInfo"
104    rows="3"
105  ></textarea>
106 </div>
107 <button type="submit" class="btn-derma btn-add-to-cart-look d-flex justify-content-between">
108   <span>Order</span>
109   <span class="btn-add-to-cart-plus"></span>
110 </button>
111 </form>
112 </div>
113 </div>
114 </div>
115 </div>
116 </div>
117 <?php include 'partials/footer.php' ?>
```

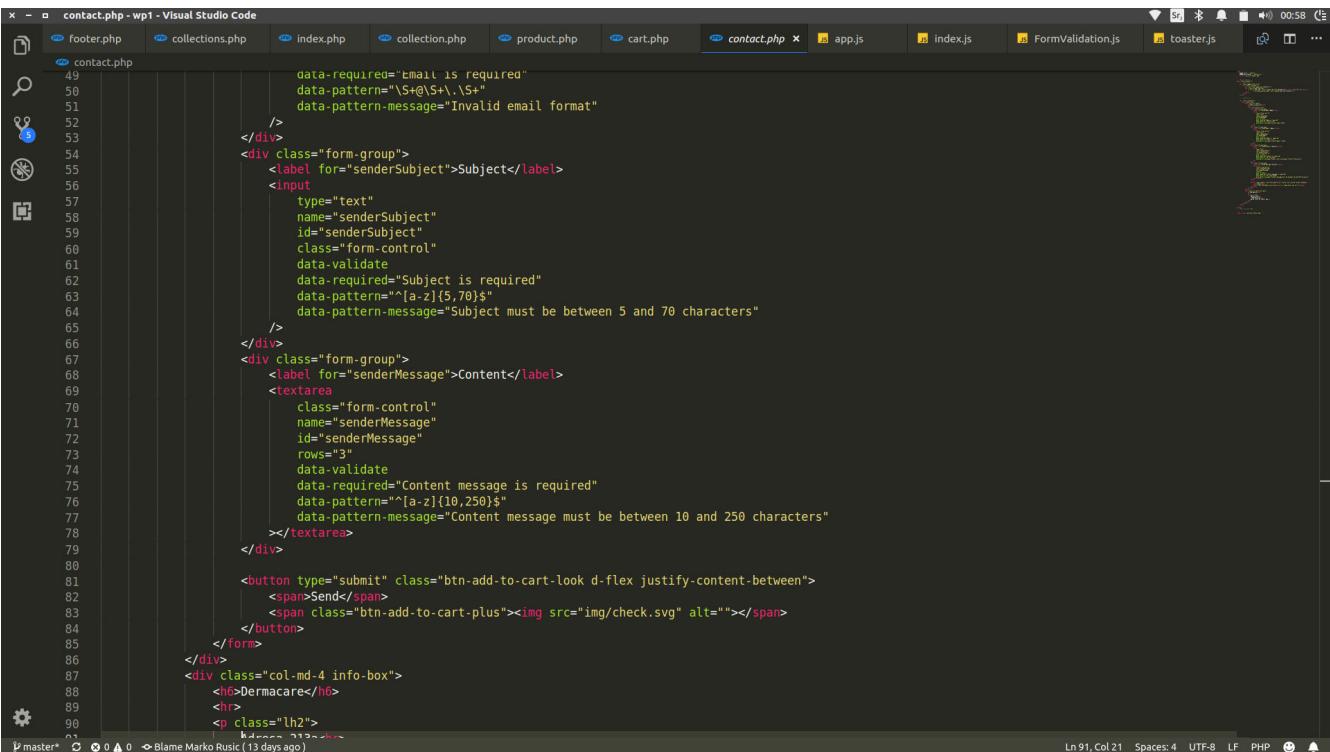
Ln 1, Col 1 Spaces: 4 UTF-8 LF PHP 🎉

## 3.6 Kontakt strana

Kontakt forma se dinamički validira kroz JavaScript na osnovu data atributa na svakom polju.



```
contact.php
7 <!-- Main Container -->
8 <main class="contact">
9   <!-- Page title -->
10  <div class="page-title-strip">
11    <div class="container">
12      <h1 class="text-uppercase">Contact</h1>
13      <div class="breadcrumb-top">
14        <a href="index.php" class="breadcrumb-item">Dermacare</a> <span class="mlr">></span> <a
15          href="glow-edition.php" class="breadcrumb-item">Contact</a>
16      </div>
17    </div>
18  </div>
19
20  <!-- Contact Form -->
21
22  <div class="container">
23    <div class="row">
24      <div class="col-md-8">
25        <h2>Have a question?</h2>
26        <br>
27        <form id="contact-form">
28          <div class="form-group">
29            <label for="senderName">Name</label>
30            <input
31              class="form-control"
32              type="text"
33              name="senderName"
34              id="senderName"
35              data-validate
36              data-required="Name is required"
37              data-pattern="^a-zA-Z ]+$"
38              data-pattern-message="Invalid name format"
39            />
40          </div>
41          <div class="form-group">
42            <label for="senderEmail">Email</label>
43            <input
44              class="form-control"
45              type="text"
46              name="senderEmail"
47              id="senderEmail"
48              data-validate
49            />
50          </div>
51          <div class="form-group">
52            <label for="senderSubject">Subject</label>
53            <input
54              type="text"
55              name="senderSubject"
56              id="senderSubject"
57              class="form-control"
58              data-validate
59              data-required="Subject is required"
60              data-pattern="[a-z]{5,70}$"
61              data-pattern-message="Subject must be between 5 and 70 characters"
62            />
63          </div>
64          <div class="form-group">
65            <label for="senderMessage">Content</label>
66            <textarea
67              class="form-control"
68              name="senderMessage"
69              id="senderMessage"
70              rows="3"
71              data-validate
72              data-required="Content message is required"
73              data-pattern="[a-z]{10,250}$"
74              data-pattern-message="Content message must be between 10 and 250 characters"
75            ></textarea>
76          </div>
77          <button type="submit" class="btn-add-to-cart-look d-flex justify-content-between">
78            <span>Send</span>
79            <span class="btn-add-to-cart-plus"></span>
80          </button>
81        </form>
82      </div>
83      <div class="col-md-4 info-box">
84        <h3>Dermacare</h3>
85        <hr>
86        <p class="lh2">
87          Address: 123 Main St, City, Country
88        </p>
89      </div>
90    </div>
91  </div>
92
93  <script src="js/contact.js" type="text/javascript"></script>
94
95  <script src="js/FormValidation.js" type="text/javascript"></script>
96
97  <script src="js/toaster.js" type="text/javascript"></script>
98
99
100 </body>
101 </html>
```



```
contact.php
49          data-required="Email is required"
50          data-pattern="\S+@\S+\.\S+"
51          data-pattern-message="Invalid email format"
52        />
53      </div>
54      <div class="form-group">
55        <label for="senderSubject">Subject</label>
56        <input
57          type="text"
58          name="senderSubject"
59          id="senderSubject"
60          class="form-control"
61          data-validate
62          data-required="Subject is required"
63          data-pattern="[a-z]{5,70}$"
64          data-pattern-message="Subject must be between 5 and 70 characters"
65        />
66      </div>
67      <div class="form-group">
68        <label for="senderMessage">Content</label>
69        <textarea
70          class="form-control"
71          name="senderMessage"
72          id="senderMessage"
73          rows="3"
74          data-validate
75          data-required="Content message is required"
76          data-pattern="[a-z]{10,250}$"
77          data-pattern-message="Content message must be between 10 and 250 characters"
78        ></textarea>
79      </div>
80      <button type="submit" class="btn-add-to-cart-look d-flex justify-content-between">
81        <span>Send</span>
82        <span class="btn-add-to-cart-plus"></span>
83      </button>
84    </form>
85  </div>
86  <div class="col-md-4 info-box">
87    <h3>Dermacare</h3>
88    <hr>
89    <p class="lh2">
90      Address: 123 Main St, City, Country
91    </p>
92  </div>
93
94  <script src="js/contact.js" type="text/javascript"></script>
95
96  <script src="js/FormValidation.js" type="text/javascript"></script>
97
98  <script src="js/toaster.js" type="text/javascript"></script>
99
100 </body>
101 </html>
```

```
contact.php - wp1 - Visual Studio Code
Footer.php collections.php index.php collection.php product.php cart.php contact.php app.js index.js FormValidation.js toaster.js ...
contact.php
77     ></textarea>
78     </div>
79
80     <button type="submit" class="btn-add-to-cart-look d-flex justify-content-between">
81         <span>Send</span>
82         <span class="btn-add-to-cart-plus"></span>
83     </button>
84
85     </form>
86 </div>
87 <div class="col-md-4 info-box">
88     <h6>Dermacare</h6>
89     <hr>
90     <p class="lh2">
91         Adresa 213a<br>
92         11000 Beograd<br>
93         marko.rusic.22.17@ict.edu.rs
94     </p>
95 </div>
96 </div>
97 </div>
98
99 </main>
100 <!-- Main Container END -->
101
102
103 <?php include 'partials/footer.php' ?>
104
105
```

### 3.7 Strana o autoru

about.php - wp1 - Visual Studio Code

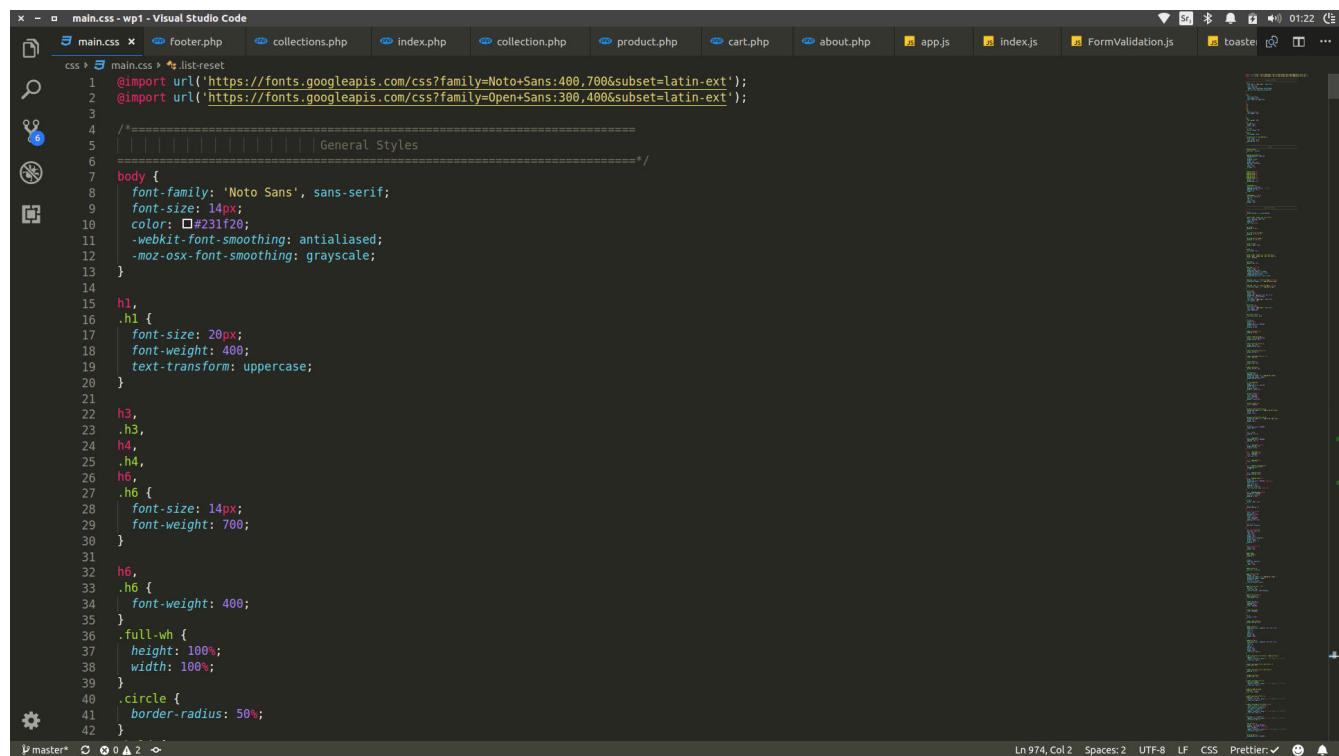
```
1 <?php
2     $page_title = 'About me';
3     $page_description = 'Imam 20 godina, iz Beograda sam, student druge godine Visoke ICT skole, smer Internet tehnologije. Aktivno se bavim web programiranjem od
4     include 'partials/header.php';
5 ?>
6
7     <main id="about-me">
8         <div class="container py-5">
9             <div class="content-wrapper flex-center-col">
10                 
11                 <p class="author-info">
12                     <span class="bold uc">Marko Rusic</span>
13                     </p><p><i class="fa fa-envelope-o" aria-hidden="true"></i> marko.rusic.22.17@ict.edu.rs</p>
14                     <p><i class="fa fa-id-card-o" aria-hidden="true"></i> Broj indeksa: 22/17</p>
15                 </p></p>
16                 <p class="text-center author-text"><?= $page_description ?> </p>
17                 <div class="author-more-info text-center mt-3">
18                     <p>Korisini linkovi:</p>
19                     <div class="useful-links">
20                         <a href="https://github.com/markorusic" class="btn btn-derma" style="width: 250px;" target="_blank">
21                             <i class="fa fa-github"></i> GitHub
22                         </a>
23                         <a href="https://codepen.io/markol23/" class="btn btn-derma" style="width: 250px;" target="_blank">
24                             <i class="fa fa-codepen"></i> Codepen
25                         </a>
26                     </div>
27                     <hr>
28                     <p>
29                         Repozitorijum ovog sajta: <a class="repo black" href="https://github.com/markorusic/wp1" target="_blank">https://github.com/markorusic/wp1
30                         </a>
31                     </p>
32                 </div>
33             </div>
34         </main>
35
36     <?php include 'partials/footer.php' ?>
```

master\* 0 0 ▲ 0 ↵ Ln 29, Col 1 Tab Size: 4 UTF-8 LF PHP ⚡

# 4. Stilizovanje HTML elemenata - CSS

CSS je zajednički za sve stranice na sajtu i sadržan je u main.css fajlu. Odvojen je po sledećim sekcijama.

## 4.1 Generalni stilovi



The screenshot shows the Visual Studio Code interface with the main.css file open. The code defines general styles for various HTML elements:

```
1 @import url('https://fonts.googleapis.com/css?family=Noto+Sans:400,700&subset=latin-ext');
2 @import url('https://fonts.googleapis.com/css?family=Open+Sans:300,400&subset=latin-ext');

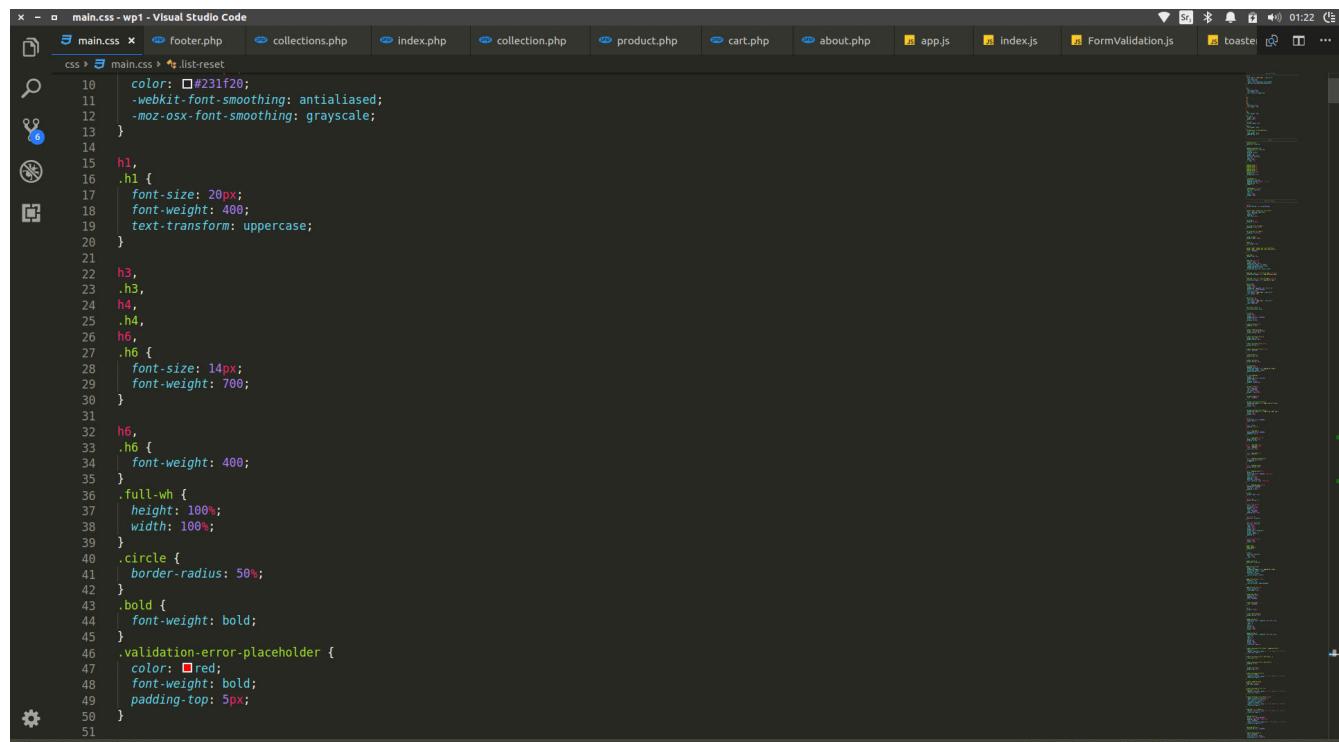
3 /*=====
4  *===== General Styles =====
5  *===== */
6 body {
7     font-family: 'Noto Sans', sans-serif;
8     font-size: 14px;
9     color: #231f20;
10    -webkit-font-smoothing: antialiased;
11    -moz-osx-font-smoothing: grayscale;
12 }

13 h1,
14 .h1 {
15     font-size: 20px;
16     font-weight: 400;
17     text-transform: uppercase;
18 }

19 h3,
20 .h3,
21 h4,
22 .h4,
23 h5,
24 .h5,
25 h6,
26 .h6 {
27     font-size: 14px;
28     font-weight: 700;
29 }

30 h6,
31 .h6 {
32     font-weight: 400;
33 }
34 .full-wh {
35     height: 100%;
36     width: 100%;
37 }
38 .circle {
39     border-radius: 50%;
40 }

41 .bold {
42     font-weight: bold;
43 }
44 .validation-error-placeholder {
45     color: red;
46     font-weight: bold;
47     padding-top: 5px;
48 }
49 
```



The screenshot shows the Visual Studio Code interface with the main.css file open, showing additional styles:

```
10 color: #231f20;
11 -webkit-font-smoothing: antialiased;
12 -moz-osx-font-smoothing: grayscale;
13 }

14 h1,
15 .h1 {
16     font-size: 20px;
17     font-weight: 400;
18     text-transform: uppercase;
19 }

20 h3,
21 .h3,
22 h4,
23 .h4,
24 h5,
25 .h5,
26 h6,
27 .h6 {
28     font-size: 14px;
29     font-weight: 700;
30 }

31 h6,
32 .h6 {
33     font-weight: 400;
34 }
35 .full-wh {
36     height: 100%;
37     width: 100%;
38 }
39 .circle {
40     border-radius: 50%;
41 }

42 .bold {
43     font-weight: bold;
44 }
45 .validation-error-placeholder {
46     color: red;
47     font-weight: bold;
48     padding-top: 5px;
49 }
50 
```

## 4.2 Layout

The screenshot shows the Visual Studio Code interface with the file `main.css` open. The code is organized into sections with comments. The `list-reset` section contains styles for the `.heading-wrap` class, which includes a pseudo-element `.heading-wrap::before` with a background color of `#231f20` and absolute positioning. It also includes styles for `h1` through `h6` headings with white backgrounds. The `videoWrapper` section sets a relative position and padding for a video container, while the `videoWrapper iframe` section sets absolute positions for the `iframe` within it.

```
52 /*=====
53 |===== Layout
54 =====*/
55 .heading-wrap {
56   position: relative;
57 }
58
59 .heading-wrap::before {
60   background-color: #231f20;
61   content: '';
62   display: block;
63   height: 1px;
64   margin-top: -1px;
65   position: absolute;
66   top: 50%;
67   width: 100%;
68   z-index: -1;
69 }
70
71 .heading-wrap h1,
72 .heading-wrap h2,
73 .heading-wrap h3,
74 .heading-wrap h4,
75 .heading-wrap h5,
76 .heading-wrap h6 {
77   margin: 0;
78   background: #fff;
79 }
80
81 .videoWrapper {
82   position: relative;
83   padding-bottom: 56.25%; /* 16:9 */
84   padding-top: 25px;
85   height: 0;
86 }
87
88 .videoWrapper iframe {
89   position: absolute;
90   top: 0;
91   left: 0;
92   width: 100%;
93   height: 100%;
```

## 4.3 Specifični stilovi

The screenshot shows the Visual Studio Code interface with the file `main.css` open. The code is organized into sections with comments. The `list-reset` section contains styles for the `header` element, including a solid border at the bottom. It then moves on to the `navbar-light` section, specifically targeting `.nav-link` elements. These styles include uppercase text transform, a color of `#231f20`, zero padding, and a font size of 14px. It also defines styles for `.nav-item` with padding, `.nav-item:first-of-type` with padding, and `.nav-item:last-of-type` with padding. The `.navbar-brand` section adds a right margin of 100px. The `.navbar` section sets a minimum height of 85px. The `list-reset` section concludes with styles for `.nav-link:focus` and `.nav-link:hover`, both changing the color to `#838088`. Finally, the `home_top` section adds a margin-top of 25px.

```
96 /*=====
97 |===== Specific Styles
98 =====*/
99
100 header {
101   border-bottom: 2px solid #8d7d0a;
102 }
103
104 .navbar-light .navbar-nav .nav-link {
105   text-transform: uppercase;
106   color: #231f20;
107   padding: 0;
108   font-size: 14px;
109 }
110
111 .nav-item {
112   padding: 0 25px;
113 }
114
115 .nav-item:first-of-type {
116   padding: 0 25px 0 0;
117 }
118
119 .nav-item:last-of-type {
120   padding: 0 0 0 25px;
121 }
122
123 .navbar-brand {
124   margin-right: 100px;
125 }
126
127 .navbar {
128   min-height: 85px;
129 }
130
131 .navbar-light .navbar-nav .nav-link:focus,
132 .navbar-light .navbar-nav .nav-link:hover {
133   color: #838088;
134 }
135
136 .home_top {
137   margin-top: 25px;
```

main.css - wp1 - Visual Studio Code

```
css > main.css > .list-reset
136 .home-top {
137   margin-top: 25px;
138 }
139
140 .home-top article {
141   height: 380px;
142   margin-bottom: 50px;
143   background-repeat: no-repeat;
144   -webkit-background-size: cover;
145   background-size: cover;
146   background-position: center center;
147 }
148
149 .home-top .row > div:first-of-type article {
150   background-image: url('/img/naslovna1.jpg');
151 }
152
153 .home-top .row > div:last-of-type article {
154   background-image: url('/img/about2.jpg');
155 }
156
157 .box-block {
158   width: 90px;
159   height: 90px;
160   background: #rgba(255, 255, 255, 0.9);
161   border: 2px solid #8d7d0a;
162   font-size: 20px;
163   font-family: 'Open Sans', sans-serif;
164   font-weight: 300;
165 }
166
167 .box-block a {
168   font-size: 20px;
169   font-family: 'Open Sans', sans-serif;
170   font-weight: 300;
171   color: #231f20;
172 }
173
174 .box-block a:hover {
175   text-decoration: none;
176 }
```

Ln 974, Col 2 Spaces: 2 UTF-8 LF CSS Prettier: ✓

main.css - wp1 - Visual Studio Code

```
css > main.css > .btn-derma
178 .hr-line {
179   width: 58px;
180   height: 1px;
181   background-color: #8d7d0a;
182   margin: 3px 0;
183   display: block;
184 }
185
186 .product-section h3 {
187   padding: 0 20px;
188 }
189
190 .product-image-preview {
191   border: 1px solid #231f20;
192   margin-bottom: 20px;
193 }
194
195 .product-preview-article {
196   margin-top: 25px;
197   margin-bottom: 20px;
198 }
199
200 .product-preview-article h6 {
201   margin-bottom: 0;
202 }
203
204 .product-preview-article h4 a {
205   color: #231f20;
206 }
207
208 .single-product {
209   margin-top: 40px;
210 }
211
212 .product-section {
213   margin-bottom: 50px;
214 }
215
216 .testimonials {
217   padding: 50px 0;
218   background-image: url('/img/pattern.jpg');
219   background-repeat: repeat;
220 }
```

Ln 993, Col 25 Spaces: 2 UTF-8 LF CSS Prettier: ✓

```
main.css - wp1 - Visual Studio Code
main.css footer.php collections.php index.php collection.php product.php cart.php about.php app.js index.js FormValidation.js toaste...
CSS > main.css > .btn-derma
216 .testimonials {
217   padding: 50px 0;
218   background-image: url('/img/pattern.jpg');
219   background-repeat: repeat;
220   margin-bottom: 40px;
221 }
222 .hr-line-heading {
223   height: 1px;
224   background-color: #231f20;
225   width: 100px;
226   display: block;
227   margin: 0 auto 30px;
228 }
229 .carousel-item p {
230   font-size: 19px;
231   color: #000000;
232   max-width: 770px;
233   margin: 0 auto 20px;
234 }
235 .carousel-item h6 {
236   color: #038088;
237 }
238 .carousel-control-prev-icon {
239   background-image: url('img/arrow-left.svg');
240   width: 22px;
241   height: 45px;
242 }
243 .carousel-control-next-icon {
244   background-image: url('img/arrow-right.svg');
245   width: 22px;
246   height: 45px;
247 }
248 .footer {
249   background-color: #838088;
250   color: #fff;
251 }
252 .footer .row {
253   padding: 45px 0;
254 }
255 .footer .copyright {
256   background-color: #000000;
257   padding: 10px 0;
258 }
259 .footer .copyright small {
260   margin-bottom: 5px;
261   display: block;
262 }
263 .footer .copyright small,
264 .footer .copyright p {
265   color: #038088;
266   font-size: 12px;
267 }
268 .footer .copyright a {
269   color: #cfcfcf;
270 }
271 .footer .heading-wrap:before {
272   background-color: #fff;
273   z-index: 0;
274 }
275 .footer .heading-wrap {
276   margin-bottom: 35px;
277 }
278 .footer .heading-wrap h5 {
279   margin: 0;
280   background-color: #038088 !important;
281   color: #fff;
282   font-size: 14px;
283   padding: 0 20px;
284   position: relative;
285   text-transform: none !important;
286 }
287 .footer .heading-wrap h5:after {
288   content: '';
289   position: absolute;
290   right: 0;
291   top: 50%;
292   width: 0;
293   height: 0;
294   border-top: 10px solid transparent;
295   border-bottom: 10px solid transparent;
296   border-left: 20px solid #038088;
297 }
298 .footer .heading-wrap h5:after {
299   content: '';
300   position: absolute;
301   left: 0;
302   top: 50%;
```

```
main.css - wp1 - Visual Studio Code
main.css footer.php collections.php index.php collection.php product.php cart.php about.php app.js index.js FormValidation.js toaste...
CSS > main.css > .btn-derma
259 .footer .row {
260   padding: 45px 0;
261 }
262 .footer .copyright {
263   background-color: #000000;
264   padding: 10px 0;
265 }
266 .footer .copyright small {
267   margin-bottom: 5px;
268   display: block;
269 }
270 .footer .copyright small,
271 .footer .copyright p {
272   color: #038088;
273   font-size: 12px;
274 }
275 .footer .copyright a {
276   color: #cfcfcf;
277 }
278 .footer .heading-wrap:before {
279   background-color: #fff;
280   z-index: 0;
281 }
282 .footer .heading-wrap {
283   margin-bottom: 35px;
284 }
285 .footer .heading-wrap h5 {
286   margin: 0;
287   background-color: #038088 !important;
288   color: #fff;
289   font-size: 14px;
290   padding: 0 20px;
291   position: relative;
292   text-transform: none !important;
293 }
294 .footer .heading-wrap h5:after {
295   content: '';
296   position: absolute;
297   right: 0;
298   top: 50%;
299   width: 0;
300   height: 0;
301   border-top: 10px solid transparent;
302   border-bottom: 10px solid transparent;
303   border-left: 20px solid #038088;
304 }
305 .footer .heading-wrap h5:after {
306   content: '';
307   position: absolute;
308   left: 0;
309   top: 50%;
```

```
main.css - wp1 - Visual Studio Code
main.css footer.php collections.php index.php collection.php product.php cart.php about.php app.js index.js FormValidation.js toaster ...
css > main.css > .btn-derma
312 footer p {
313   margin-bottom: 0;
314 }
315 footer form input {
316   width: 100%;
317   max-width: 370px;
318   height: 45px;
319   font-size: 14px;
320   color: #838088;
321   padding-left: 10px;
322 }
323 footer form {
324   position: relative;
325 }
326 footer form button {
327   position: absolute;
328   top: 10px;
329   right: 10px;
330   width: 34px;
331   height: 28px;
332   background: transparent;
333   border: none;
334   border-radius: 0;
335   padding: 0;
336 }
337 footer form button {
338   width: 34px;
339   height: 28px;
340 }
341 .modal-body,
342 .modal-header {
343   padding: 0;
344 }
345 .close {
346   position: absolute;
347   top: -20px;
348 }
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
Ln 993, Col 25 Spaces: 2 UTF-8 LF CSS Prettier: ✓ 🎉 🔍
```

```
main.css - wp1 - Visual Studio Code
main.css footer.php collections.php index.php collection.php product.php cart.php about.php app.js index.js FormValidation.js toaster ...
css > main.css > .btn-derma
356 .modal-content {
357   position: relative;
358 }
359
360 .page-title-strip {
361   height: 85px;
362   background-image: url('/img/pattern.jpg');
363   background-repeat: repeat;
364   display: flex;
365   flex-flow: column;
366   justify-content: center;
367 }
368
369 .page-title-strip > div {
370   display: flex;
371   flex-flow: row;
372   justify-content: space-between;
373 }
374
375 .page-title-strip h1 {
376   margin-bottom: 0;
377   line-height: 41px;
378 }
379
380 .breadcrumb-top {
381   background: #fff;
382   padding: 10px;
383   color: #838088;
384 }
385
386 .breadcrumb-item {
387   color: #838088;
388 }
389
390 .mlr {
391   margin: 0 10px;
392 }
393
394 .product-description {
395   margin-bottom: 50px;
396 }
397
Ln 993, Col 25 Spaces: 2 UTF-8 LF CSS Prettier: ✓ 🎉 🔍
```

main.css - wp1 - Visual Studio Code

```
css > main.css > *.btn-derma
399 .image-overlay {
400   background-color: #rgba(255, 255, 255, 0.8);
401   left: 0;
402   right: 0;
403   top: 0;
404   bottom: 0;
405   width: 100%;
406   height: 100%;
407 }
408
409 .image-overlay {
410   background-color: #rgba(255, 255, 255, 0.8);
411   left: 0;
412   right: 0;
413   top: 0;
414   bottom: 0;
415   width: 100%;
416   height: 100%;
417   transition: ease 1s;
418 }
419
420 .product-preview-article:hover .image-overlay {
421   opacity: 0;
422   -webkit-transition: ease 1s; /* For Safari 3.1 to 6.0 */
423   transition: ease 1s;
424 }
425
426 .product-preview-article .box-block a {
427   font-size: 17px;
428 }
429
430 .product-preview-article .box-block {
431   padding: 0 5px;
432 }
433
434 .category-section {
435   margin-top: 25px;
436 }
437
438 .product-preview-article {
439   overflow: hidden;
440   -webkit-transition: ease 1s; /* For Safari 3.1 to 6.0 */
441 }
```

master\* 0 0 ▲ 2 → Blame Marko Rusic (14 days ago) Ln 993, Col 25 Spaces: 2 UTF-8 LF CSS Prettier ✓

main.css - wp1 - Visual Studio Code

```
css > main.css > *.btn-derma
443 .product-image-preview,
444 .home-top article {
445   overflow: hidden;
446 }
447
448 .product-preview-article img,
449 .home-top article {
450   -webkit-transition: ease 1s; /* For Safari 3.1 to 6.0 */
451   transition: ease 1s;
452 }
453
454 .product-preview-article:hover img {
455   -webkit-transform: scale(1.1);
456   -moz-transform: scale(1.1);
457   -o-transform: scale(1.1);
458   transform: scale(1.1);
459   -webkit-transition: ease 1s; /* For Safari 3.1 to 6.0 */
460   transition: ease 1s;
461 }
462
463 .home-top article:hover {
464   -webkit-transition: ease 1s; /* For Safari 3.1 to 6.0 */
465   transition: ease 1s;
466 }
467
468 .load-more-btn {
469   border: 1px solid #838088;
470   padding: 10px 20px !important;
471   color: #838088;
472   -webkit-transition: ease 0.5s; /* For Safari 3.1 to 6.0 */
473   transition: ease 0.5s;
474 }
475
476 .lmb-wrap:before {
477   background-color: #838088;
478 }
479
480 .load-more-btn:hover {
481   color: #838088;
482   text-decoration: none;
483   background-color: #fbfbfb;
484 }
```

master\* 0 0 ▲ 2 → Blame Marko Rusic (14 days ago) Ln 993, Col 25 Spaces: 2 UTF-8 LF CSS Prettier ✓

main.css - wp1 - Visual Studio Code

```
css > main.css > *.btn-derma
486
487 .lmb-wrap {
488     margin-top: 30px;
489 }
490
491 .product-images-thumb {
492     background-repeat: no-repeat;
493     background-position: center center;
494     background-size: cover;
495     display: block;
496     padding-bottom: 100%;
497     position: relative;
498     cursor: pointer;
499 }
500
501 .input-number {
502     font-weight: 400;
503     color: #838088;
504     border: 1px solid #231f20;
505     width: 62px;
506     height: 40px;
507     margin-right: 10px;
508     text-align: center;
509     font-size: 16px;
510     padding: 0 5px;
511 }
512
513 input[type='number']::-webkit-inner-spin-button {
514     opacity: 1;
515 }
516
517 .btn-add-to-cart,
518 .btn-add-to-cart-look {
519     width: 270px;
520     height: 40px;
521     background-color: #231f20;
522     color: #fff;
523     border: none;
524     font-weight: 700;
525     text-transform: uppercase;
526     font-size: 14px;
527     padding-left: 15px;
528 }
```

master\* 0 0 2 → Blame Marko Rusic (14 days ago) Ln 993, Col 25 Spaces: 2 UTF-8 LF CSS Prettier ✓

main.css - wp1 - Visual Studio Code

```
css > main.css > *.btn-derma
549 .arrows-product a:first-of-type {
550     margin-right: 20px;
551 }
552
553 .arrows-product a img {
554     width: 10px;
555     height: auto;
556 }
557
558 .add-to-cart-block {
559     margin: 25px 0;
560 }
561
562 .shopping-cart-icon {
563     text-align: center;
564     width: 28px;
565     height: 32px;
566 }
567
568 .num-of-showed-products {
569     color: #838088;
570 }
571
572 .product-filter select {
573     width: 270px;
574     height: 35px;
575     padding-left: 15px;
576     color: #838088;
577     border-radius: 3px;
578     border: 1px solid #838088;
579     background-color: transparent;
580 }
581
582 .product-filter {
583     margin-top: 40px;
584     margin-bottom: 15px;
585 }
586
587 .preview-product-atc {
588     bottom: -100%;
589     left: 0;
590     right: 0;
```

master\* 0 0 2 → Blame Marko Rusic (14 days ago) Ln 993, Col 25 Spaces: 2 UTF-8 LF CSS Prettier ✓

main.css - wp1 - Visual Studio Code

```
css > main.css > media-filter a
```

```
587 .preview-product-atc {  
588     bottom: -100%;  
589     left: 0;  
590     right: 0;  
591     width: 100%;  
592     z-index: 9999;  
593     position: absolute;  
594     transition: bottom 0.5s ease;  
595 }  
596  
597 .product-preview-article:hover .preview-product-atc {  
598     bottom: 0;  
599 }  
600  
601 .media-filter {  
602     margin-top: 25px;  
603 }  
604  
605 .media-filter a {  
606     font-weight: 700;  
607     color: #838088;  
608     margin-right: 5px;  
609 }  
610  
611 .media-filter a:last-of-type {  
612     margin-right: 0;  
613     margin-left: 5px;  
614 }  
615  
616 .modal-backdrop {  
617     background-color: #ffffff;  
618     opacity: 0.8;  
619 }  
620  
621 .arr-left {  
622     left: 10px;  
623     top: 50%;  
624     transform: translateY(-50%);  
625 }  
626  
627 .arr-right {  
628     right: 10px;  
629 }
```

Ln 609, Col 2 Spaces:2 UTF-8 LF CSS Prettier:✓ 🎉

main.css - wp1 - Visual Studio Code

```
css > main.css > media-filter a
```

```
626 .arr-right {  
627     right: 10px;  
628     top: 50%;  
629     transform: translateY(-50%);  
630 }  
631  
632 .media-preview {  
633     width: 100%;  
634     padding-bottom: 133%;  
635     -webkit-background-size: cover;  
636     background-size: cover;  
637     background-position: center center;  
638 }  
639  
640 #map {  
641     height: 380px;  
642     margin-top: 20px;  
643     margin-bottom: 45px;  
644 }  
645  
646 .contact h6 {  
647     font-weight: 700;  
648     font-size: 16px;  
649 }  
650  
651 .contact hr {  
652     border-color: #231f20;  
653 }  
654  
655 .contact label {  
656     font-size: 14px;  
657 }  
658  
659 .contact .form-control {  
660     border-radius: 0;  
661     border-color: #cdcfd0;  
662 }  
663  
664 .contact form,  
665 .contact .info-box {  
666     margin-bottom: 60px;  
667 }
```

Ln 609, Col 2 Spaces:2 UTF-8 LF CSS Prettier:✓ 🎉

main.css - wp1 - Visual Studio Code

```
679 @media screen and (max-width: 600px) {  
680   table#cart tbody td.form-control {  
681     display: inline !important;  
682   }  
683  
684   .actions .btn {  
685     width: 36px;  
686     margin: 1.5em 0;  
687   }  
688  
689   .actions .btn-info {  
690     float: left;  
691   }  
692  
693   .actions .btn-danger {  
694     float: right;  
695   }  
696  
697   table#cart thead {  
698     display: none;  
699   }  
700  
701   table#cart tbody td {  
702     display: block;  
703     padding: 0.6rem;  
704   }  
705  
706   table#cart tbody tr td:first-child {  
707     background: #f4f4f4;  
708     color: #231f20;  
709   }  
710  
711   table#cart tbody td:before {  
712     content: attr(data-th);  
713     font-weight: bold;  
714     display: inline-block;  
715     width: 8rem;  
716   }  
717  
718   table#cart tfoot td {  
719     display: block;  
720   }  
721  
722 }  
723  
724 master* 0 0 0 0 0 0 Blame Marko Rusic (14 days ago)
```

Ln 680, Col 38 Spaces: 2 UTF-8 LF CSS Prettier: ✓

main.css - wp1 - Visual Studio Code

```
728 .card-product-img {  
729   width: 100%;  
730   padding-bottom: 100%;  
731   -webkit-background-size: cover;  
732   background-size: cover;  
733   background-position: center center;  
734   border: 1px solid #cdcfdf0;  
735 }  
736  
737 .table thead th {  
738   border-top: none;  
739   border-color: #cdcfdf0;  
740   font-size: 14px;  
741   font-weight: 400;  
742 }  
743  
744 .table td,  
745 .table th {  
746   border-color: #cdcfdf0;  
747   border-width: 2px;  
748 }  
749  
750 .table tbody tr:last-of-type {  
751   border-bottom: 1px solid #cdcfdf0;  
752 }  
753  
754 .hidden-xs {  
755   padding-right: 0;  
756 }  
757  
758 #cart h4,  
759 .table > tbody > tr > td {  
760   font-size: 14px;  
761   font-weight: 400;  
762   color: #8f9194;  
763 }  
764  
765 #cart button {  
766   border: 1px solid #cdcfdf0;  
767   border-radius: 0;  
768   padding: 10px 15px;  
769 }  
770  
771 }  
772  
773 master* 0 0 0 0 0 0 Blame Marko Rusic (15 days ago)
```

Ln 726, Col 2 Spaces: 2 UTF-8 LF CSS Prettier: ✓

main.css - wp1 - Visual Studio Code

```
770
771 #cart {
772   margin-bottom: 60px;
773 }
774
775 #orderModal .modal-dialog {
776   width: 600px;
777   max-width: 100%;
778 }
779
780 #orderModal .modal-body {
781   padding: 40px;
782 }
783
784 #orderModal .modal-content {
785   background-color: #rgba(0, 0, 0, 0.9);
786   color: #fff;
787 }
788
789 .modal-content {
790   border-radius: 0;
791 }
792
793 #orderModal .form-control {
794   border-radius: 0;
795   border-color: #cdcfad;
796 }
797
798 .modal-content {
799   border-radius: 0;
800 }
801
802 #orderModal .form-control {
803   border-radius: 0;
804   border-color: #cdcfad;
805 }
806
807 .shopping-cart {
808   margin-top: 30px;
809 }
810
811 .cart-sum {
```

master\* 0 0 1 → Blame Marko Rusic (15 days ago)

Ln 726, Col 2 Spaces: 2 UTF-8 LF CSS Prettier ✓

main.css - wp1 - Visual Studio Code

```
810
811 .cart-sum {
812   font-size: 16px;
813 }
814
815 #orderModal .btn-add-to-cart {
816   background-color: #8d7d0a;
817 }
818
819 #orderModal .modal-content {
820   border-color: #8d7d0a;
821 }
822
823 .s-cart .modal-backdrop {
824   background-color: #fffffe;
825   opacity: 0;
826 }
827
828 ::selection {
829   background: #d4c00b; /* WebKit/Blink Browsers */
830 }
831 ::moz-selection {
832   background: #d4c00b; /* Gecko Browsers */
833 }
834
835 .read-more-btn {
836   font-weight: 700;
837   color: #231f20;
838 }
839
840 /*=====
841 | Responsive
842 =====*/
843
844 .product-image-preview img {
845   display: block;
846   margin: auto;
847 }
848
849 @media only screen and (max-width: 768px) {
850   .page-title-strip {
851     height: auto;
```

master\* 0 0 1 → Blame Marko Rusic (15 days ago)

Ln 726, Col 2 Spaces: 2 UTF-8 LF CSS Prettier ✓

main.css - wp1 - Visual Studio Code

```
css > main.css > ...
840  ====== Responsive ======
841  ====== ====== ====== ======
842  ====== ====== ====== ======
843
844 .product-image-preview img {
845   display: block;
846   margin: auto;
847 }
848
849 @media only screen and (max-width: 768px) {
850   .page-title-strip {
851     height: auto;
852   }
853
854   .page-title-strip > div {
855     display: block;
856   }
857
858   .page-title-strip h1 {
859     margin-bottom: 10px;
860     margin-top: 10px;
861   }
862
863   .navbar-brand {
864     margin-right: 0;
865   }
866
867   footer .heading-wrap {
868     margin-bottom: 20px;
869     margin-top: 30px;
870   }
871
872   .modal-dialog {
873     margin-top: 50px;
874   }
875
876   .hidden-xs {
877     display: none;
878   }
879
880
881 @media only screen and (max-width: 991px) {
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
```

master\* 0 0 1 Blame Marko Rusic (15 days ago)

Ln 726, Col 2 Spaces: 2 UTF-8 LF CSS Prettier: ✓

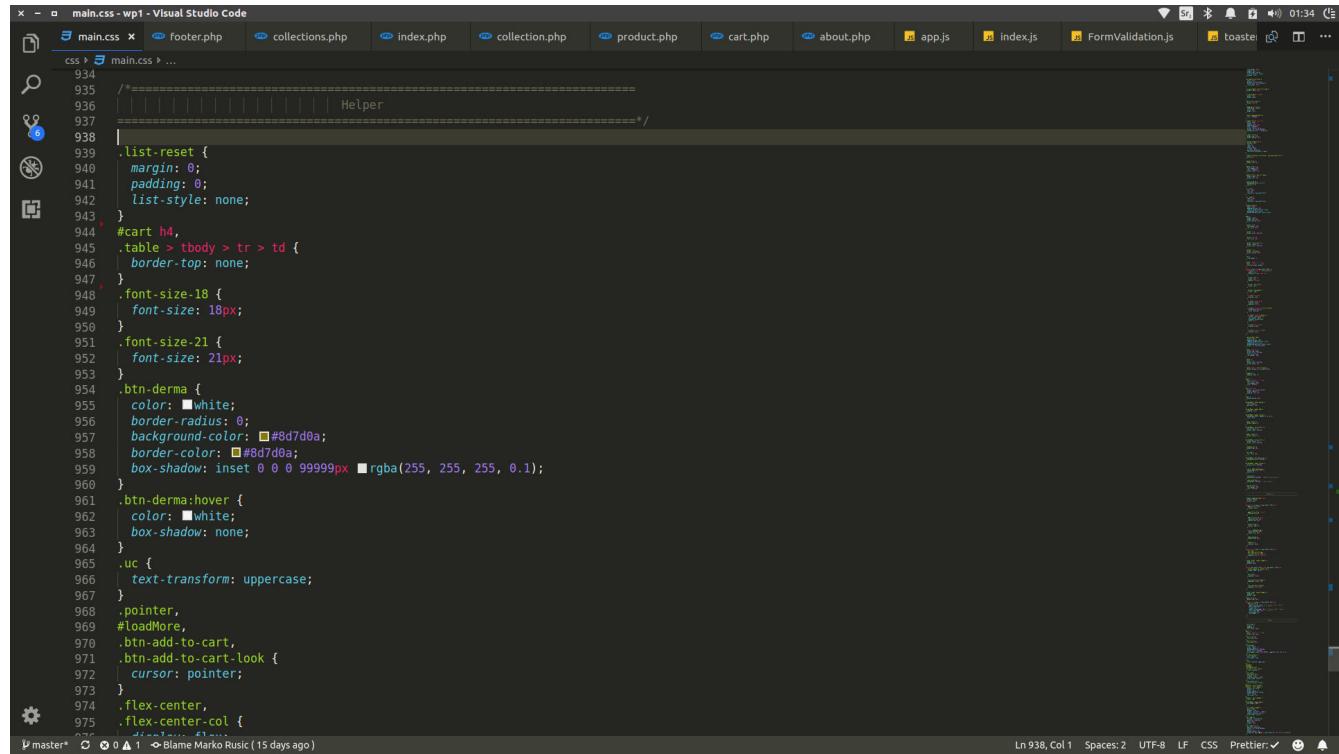
main.css - wp1 - Visual Studio Code

```
css > main.css > ...
881 @media only screen and (max-width: 991px) {
882   .nav-item,
883   .nav-item:last-of-type,
884   .nav-item:first-of-type {
885     padding: 10px 0 !important;
886   }
887
888
889 .navbar-light .navbar-toggler {
890   padding: 0;
891   border: none;
892 }
893
894 @media (min-width: 992px) and (max-width: 1200px) {
895   .navbar-expand-lg .navbar-nav {
896     align-items: center;
897   }
898
899   .nav-item {
900     padding: 0 20px;
901   }
902
903   .nav-item:first-of-type {
904     padding: 0 20px 0 0;
905   }
906
907   .nav-item:last-of-type {
908     padding: 0 0 0 20px;
909   }
910
911
912 .navbar-light .navbar-toggler {
913   padding: 0;
914   border: none;
915 }
916 .about-us-tet {
917   margin-top: 30px;
918 }
919 @media only screen and (min-width: 991px) {
920   .about-us-tet {
921     -webkit-column-count: 2; /* Chrome, Safari, Opera */
922     -moz-column-count: 2; /* Firefox */
923   }
924 }
```

master\* 0 0 1 Blame Marko Rusic (15 days ago)

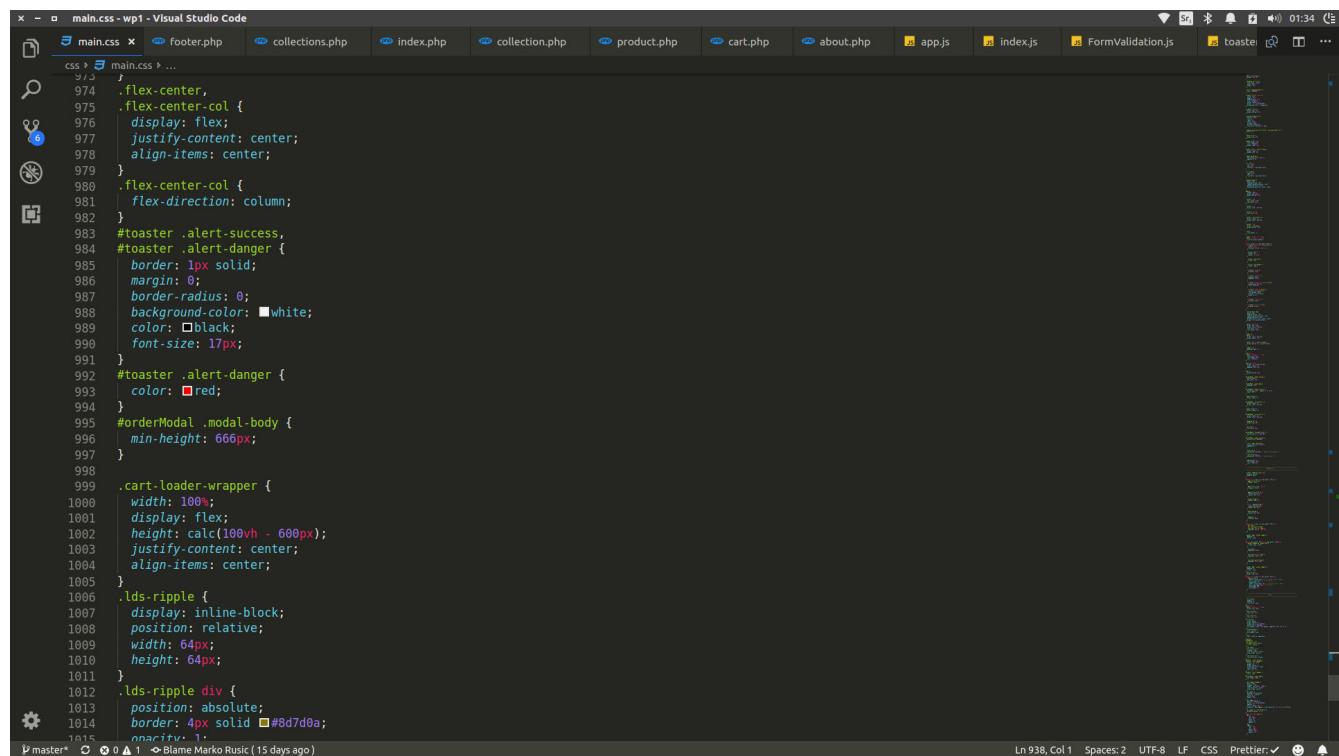
Ln 726, Col 2 Spaces: 2 UTF-8 LF CSS Prettier: ✓

## 4.4 CSS helperi



A screenshot of Visual Studio Code showing a file named `main.css`. The code is a CSS file containing various utility classes and styles. The top part of the code defines a `.list-reset` class for list items, a `#cart` class for a cart section, and several font size classes (`.font-size-18`, `.font-size-21`). Below these are styles for a button class `.btn-derma` and its hover state. Further down are styles for a class `.uc` (transform uppercase), pointer classes (`.pointer`, `#loadMore`, `.btn-add-to-cart`, `.btn-add-to-cart-look`), and flex center classes (`.flex-center`, `.flex-center-col`). The code is heavily commented with `/*===== Helper =====*/`. The status bar at the bottom shows the file is at line 938, column 1, with 2 spaces, using LF line endings, and CSS syntax.

```
/*=====
 |-----| Helper
 -----
 .list-reset {
 margin: 0;
 padding: 0;
 list-style: none;
 }
 #cart h4,
 .table > tbody > tr > td {
 border-top: none;
 }
 .font-size-18 {
 font-size: 18px;
 }
 .font-size-21 {
 font-size: 21px;
 }
 .btn-derma {
 color: white;
 border-radius: 0;
 background-color: #8d7d0a;
 border-color: #8d7d0a;
 box-shadow: inset 0 0 0 99999px rgba(255, 255, 255, 0.1);
 }
 .btn-derma:hover {
 color: white;
 box-shadow: none;
 }
 .uc {
 text-transform: uppercase;
 }
 .pointer,
 #loadMore,
 .btn-add-to-cart,
 .btn-add-to-cart-look {
 cursor: pointer;
 }
 .flex-center,
 .flex-center-col {
 display: flex;
 justify-content: center;
 align-items: center;
 }
 .flex-center-col {
 flex-direction: column;
 }
 #toaster .alert-success,
 #toaster .alert-danger {
 border: 1px solid;
 margin: 0;
 border-radius: 0;
 background-color: white;
 color: black;
 font-size: 17px;
 }
 #toaster .alert-danger {
 color: red;
 }
 #orderModal .modal-body {
 min-height: 666px;
 }
 .cart-loader-wrapper {
 width: 100%;
 display: flex;
 height: calc(100vh - 666px);
 justify-content: center;
 align-items: center;
 }
 .lds-ripple {
 display: inline-block;
 position: relative;
 width: 64px;
 height: 64px;
 }
 .lds-ripple div {
 position: absolute;
 border: 4px solid #8d7d0a;
 opacity: 1;
 }
```



A screenshot of Visual Studio Code showing the continuation of the `main.css` file. It includes definitions for the `.lds-ripple` class (a loading indicator) and its child `div` element, which has absolute positioning and a border radius of 50%. The code is annotated with comments explaining the styling. The status bar at the bottom shows the file is at line 938, column 1, with 2 spaces, using LF line endings, and CSS syntax.

```
.lds-ripple div {
 position: absolute;
 border: 4px solid #8d7d0a;
 opacity: 1;
}
```

x - main.css - wp1 - Visual Studio Code

main.css x footer.php collections.php index.php collection.php product.php cart.php about.php app.js index.js FormValidation.js toaste...

css > main.css > ...

```
998 .cart-loader-wrapper {  
999   width: 100%;  
1000  display: flex;  
1001  height: calc(100vh - 600px);  
1002  justify-content: center;  
1003  align-items: center;  
1004 }  
1005 .lds-ripple {  
1006  display: inline-block;  
1007  position: relative;  
1008  width: 64px;  
1009  height: 64px;  
1010 }  
1011 .lds-ripple div {  
1012  position: absolute;  
1013  border: 4px solid #8d7d0a;  
1014  opacity: 1;  
1015  border-radius: 50%;  
1016  animation: lds-ripple 1s cubic-bezier(0, 0.2, 0.8, 1) infinite;  
1017 }  
1018 .lds-ripple div:nth-child(2) {  
1019  animation-delay: -0.5s;  
1020 }  
1021 @keyframes lds-ripple {  
1022  0% {  
1023    top: 28px;  
1024    left: 28px;  
1025    width: 0;  
1026    height: 0;  
1027    opacity: 1;  
1028  }  
1029  100% {  
1030    top: -1px;  
1031    left: -1px;  
1032    width: 58px;  
1033    height: 58px;  
1034    opacity: 0;  
1035  }  
1036 }  
1037 }  
1038 }
```

master\* 0 ▲ 1 → Blame Marko Rusic (15 days ago)

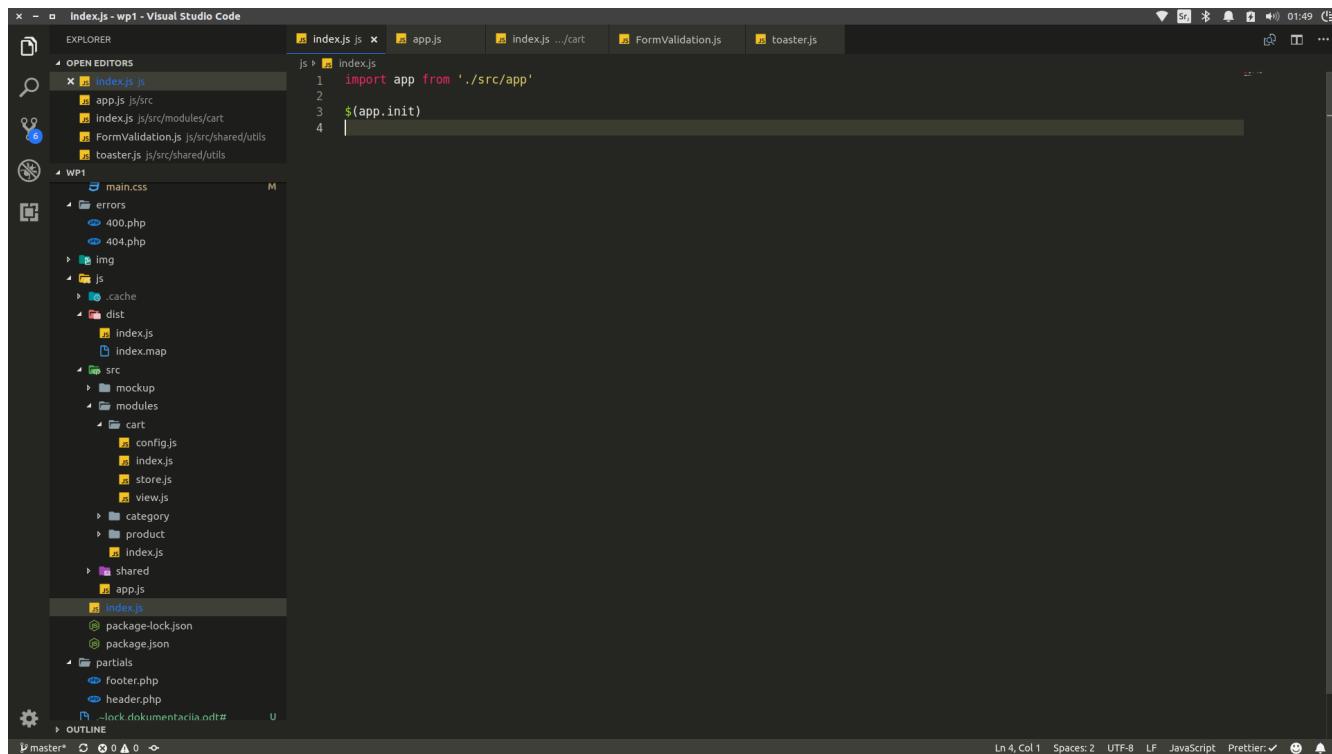
Ln 938, Col 1 Spaces: 2 UTF-8 LF CSS Prettier: ✓

# 5. JavaScript

JavaScript se transpajluje koristeći babel i bandluje u jedan produkcioni fajl js/dist/index.js, sa ciljem korišćenja svih funkcionalnosti ES6-ES7 sintakse i njihove omogućenosti u svim pretraživačima. Build tool koji u pozadini izvršava ceo taj proces je Parcel.

## 5.1 Entry fajl

Glavni entry fajl koji pokreće JS aplikaciju nalazi se u js/src/index.js fajlu i njegova svrha je da kada se ceo HTML sadržaj učita inicializuje aplikaciju.



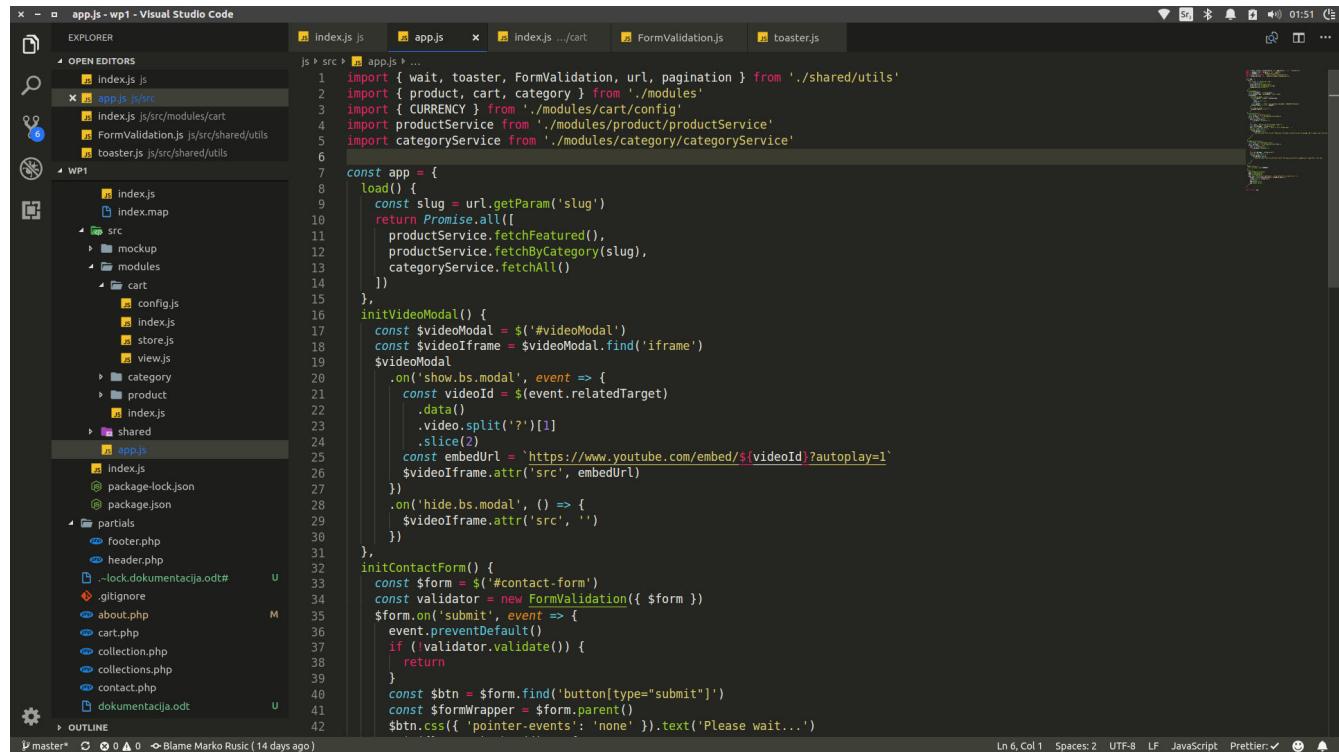
The screenshot shows the Visual Studio Code interface with the following details:

- File Explorer:** Shows the project structure. The root directory is "js". Inside "js", there are "src" and "dist" folders. "src" contains "errors", "404.php", "404.php", "img", ".cache", "dist", "src", "modules", and "cart". "src/cart" contains "config.js", "index.js", "store.js", and "view.js". "src/shared" contains "category", "product", and "index.js". "src/app" contains "app.js" and "index.js". "dist" contains "index.js" and "index.map".
- Editor:** The main editor tab is "index.js" under "js/src". The code in the editor is:

```
1 import app from './src/app'
2
3 $(app.init)
4 |
```
- Status Bar:** Shows "In 4, Col 1" and "Spaces: 2, LF, JavaScript, Prettier" along with other standard status bar icons.

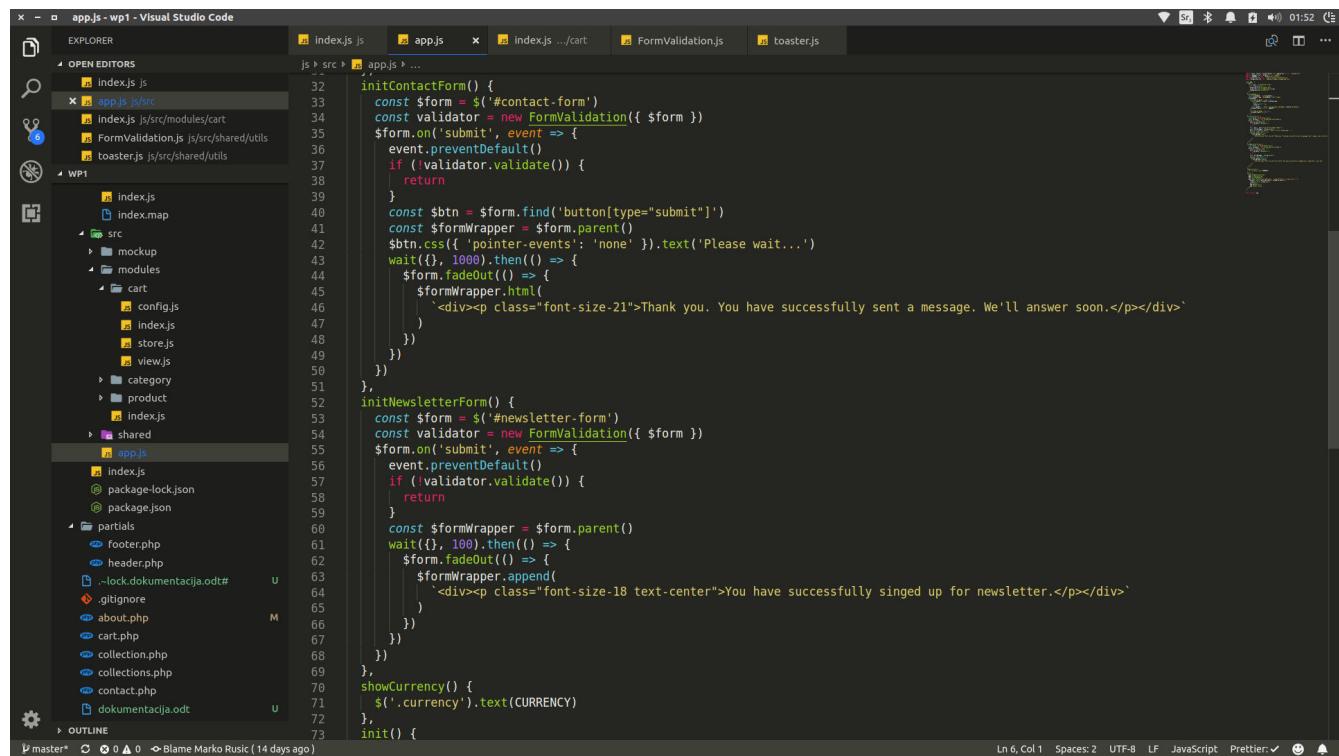
## 5.2 app.js

app.js fajl je zadužen za pokretanje aplikacije i svih njenih celina.



The screenshot shows the Visual Studio Code interface with the title bar "app.js - wp1 - Visual Studio Code". The Explorer sidebar on the left lists files like index.js, app.js, index.js (js/src), FormValidation.js, and toaster.js. The main editor area displays the app.js code:

```
1 import { wait, toaster, FormValidation, url, pagination } from './shared/utils'
2 import { product, cart, category } from './modules'
3 import { CURRENCY } from './modules/cart/config'
4 import productService from './modules/product/productService'
5 import categoryService from './modules/category/categoryService'
6
7 const app = {
8   load() {
9     const slug = url.getParam('slug')
10    return Promise.all([
11      productService.fetchFeatured(),
12      productService.fetchByCategory(slug),
13      categoryService.fetchAll()
14    ]),
15  },
16  initVideoModal() {
17    const $videoModal = $('#videoModal')
18    const $videoIframe = $videoModal.find('iframe')
19    $videoModal
20      .on('show.bs.modal', event => {
21        const videoId = $(event.relatedTarget)
22          .data()
23          .video.split('?')[1]
24        const embedUrl = `https://www.youtube.com/embed/${videoId}?autoplay=1`
25        $videoIframe.attr('src', embedUrl)
26      })
27      .on('hide.bs.modal', () => {
28        $videoIframe.attr('src', '')
29      })
30  },
31  initContactForm() {
32    const $form = $('#contact-form')
33    const validator = new FormValidation({ $form })
34    $form.on('submit', event => {
35      event.preventDefault()
36      if (!validator.validate()) {
37        return
38      }
39      const $btn = $form.find('button[type="submit"]')
40      const $formWrapper = $form.parent()
41      $btn.css({ 'pointer-events': 'none' }).text('Please wait...')
42    })
43  }
44},
```



The screenshot shows the Visual Studio Code interface with the title bar "app.js - wp1 - Visual Studio Code". The Explorer sidebar on the left lists files like index.js, app.js, index.js (js/src), FormValidation.js, and toaster.js. The main editor area displays the app.js code, which includes the code from the previous screenshot plus additional logic for handling contact form submissions and newsletter signups:

```
32
33  initContactForm() {
34    const $form = $('#contact-form')
35    const validator = new FormValidation({ $form })
36    $form.on('submit', event => {
37      event.preventDefault()
38      if (!validator.validate()) {
39        return
40      }
41      const $btn = $form.find('button[type="submit"]')
42      const $formWrapper = $form.parent()
43      $btn.css({ 'pointer-events': 'none' }).text('Please wait...')
44      wait({}, 1000).then(() => {
45        $form.fadeOut(() => {
46          $formWrapper.html(
47            '<div><p class="font-size-21">Thank you. You have successfully sent a message. We'll answer soon.</p></div>'
48        )
49      })
50    })
51  },
52  initNewsletterForm() {
53    const $form = $('#newsletter-form')
54    const validator = new FormValidation({ $form })
55    $form.on('submit', event => {
56      event.preventDefault()
57      if (!validator.validate()) {
58        return
59      }
60      const $formWrapper = $form.parent()
61      wait({}, 100).then(() => {
62        $form.fadeOut(() => {
63          $formWrapper.append(
64            '<div><p class="font-size-18 text-center">You have successfully signed up for newsletter.</p></div>'
65          )
66        })
67      })
68    })
69  },
70  showCurrency() {
71    $('.currency').text(CURRENCY)
72  },
73  init() {
```

init metoda initijalizuje osnovne module, mokapuje učitavanje podataka sa servera a potom pokreće i inicializaciju modula koji zavise od tih podataka.

```
const $form = $( '#newsletterForm' )
const validator = new FormValidation( { $form } )
$form.on('submit', event => {
  event.preventDefault()
  if (!validator.validate()) {
    return
  }
  const $formWrapper = $form.parent()
  wait(100).then(() => {
    $form.fadeOut(100) => {
      $formWrapper.append(
        `<div><p class="font-size-18 text-center">You have successfully signed up for newsletter.</p></div>`
      )
    })
  })
}, showCurrency() {
  $('.currency').text(CURRENCY)
}, init() {
  app.initNewsletterForm()
  app.initContactForm()
  app.initVideoModal()
  app.load().then(([featuredProducts, categoryProducts, categories]) => {
    product.init({ featuredProducts, categoryProducts })
    category.init({ categories })
    cart.init()
    pagination.init()
    app.showCurrency()
  })
}
}

export default app
```

## 5.3 Shared modul

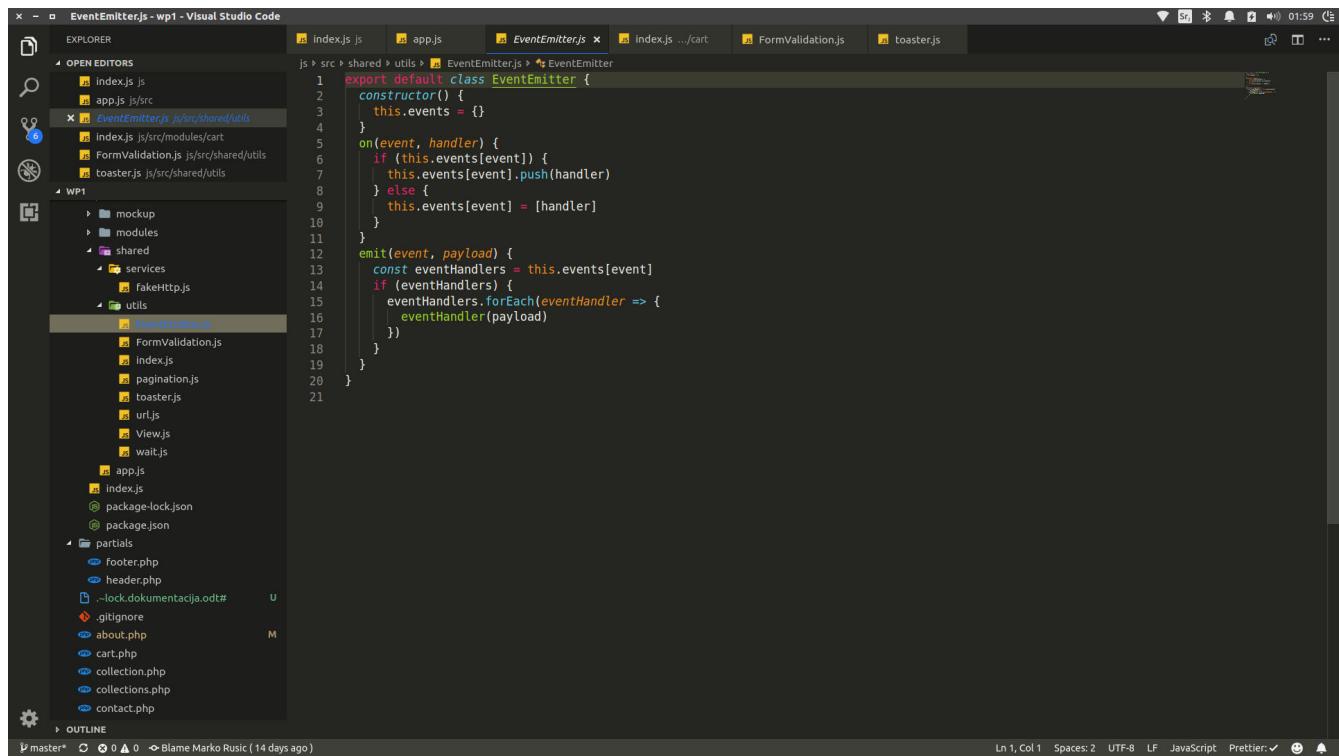
Sadrži pomoćne utility funkcije i klase, kao u servis za mokapovanje http zahteva.

```
x - markorusic@markorusic-Inspiron-5570: ~/skolica/wp
→ wp1 git:(master) X
→ wp1 git:(master) X tree js/src/shared
js/src/shared
└── services
    └── fakeHttp.js
utils
├── EventEmitter.js
├── FormValidation.js
├── index.js
├── pagination.js
├── toaster.js
├── url.js
└── View.js
wait.js

2 directories, 9 files
→ wp1 git:(master) X
```

### 5.3.1 EventEmmiter.js

Jednostavna implementacija EventEmitera, koristi se kao EventBus u modulu cart.

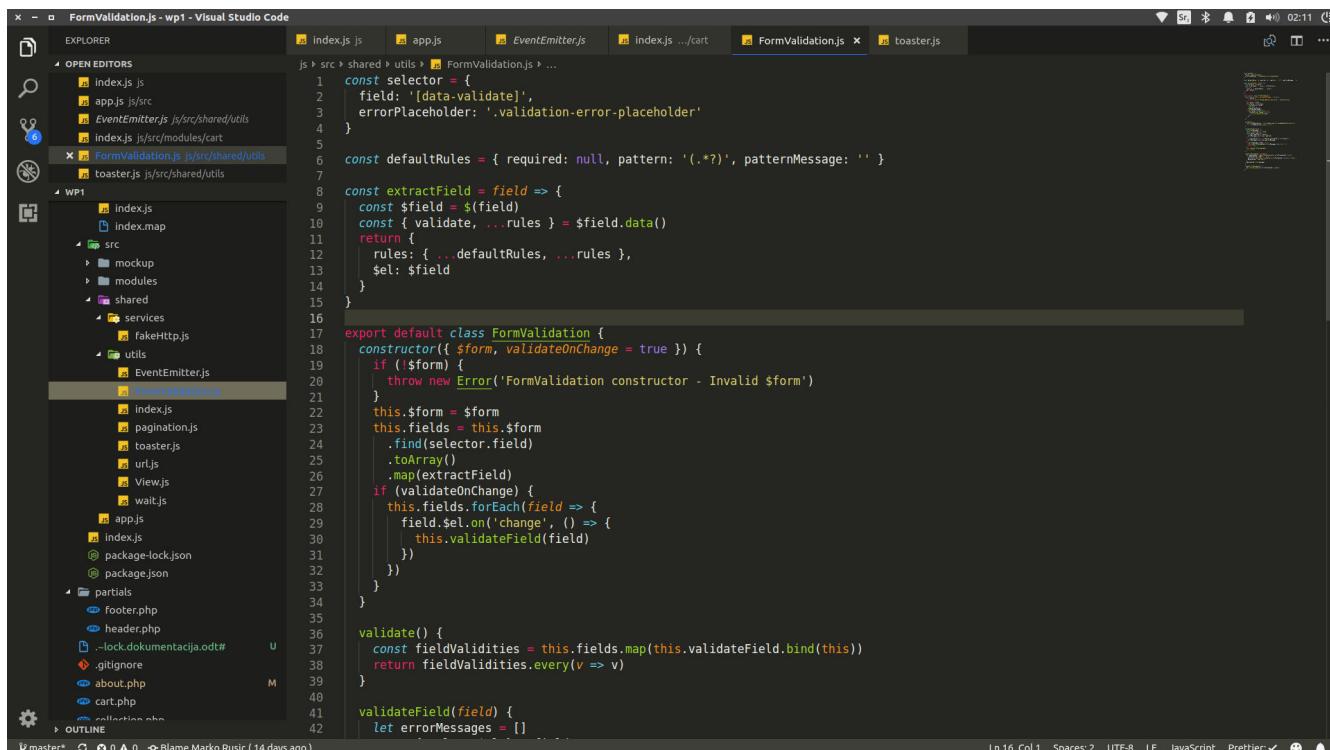


The screenshot shows the Visual Studio Code interface with the following details:

- Title Bar:** EventEmitter.js - wp1 - Visual Studio Code
- Editor:** The main area displays the code for `EventEmitter.js`. The code is a class definition with methods for adding event handlers and emitting events.
- Explorer:** On the left, the file tree shows the project structure:
  - `src` folder contains `shared`, `utils`, `index.js`, `app.js`, `FormValidation.js`, `toaster.js`, `mockup`, `modules`, `services`, and `utils`.
  - `utils` folder contains `EventEmitter.js`, `FormValidation.js`, `index.js`, `pagination.js`, `toaster.js`, `url.js`, `View.js`, and `wait.js`.
  - `app.js`, `index.js`, `package-lock.json`, and `package.json` are also listed.
- Bottom Status Bar:** master\* 0 0 0 Blame Marko Rusic (14 days ago) Ln 1, Col 1 Spaces: 2 UTF-8 LF JavaScript Prettier ✓

### 5.3.2 FormValidation.js

Klasa koja se koristi za validaciju svih formi i ispisivanje odgovarajućih grešaka. Svako polje forme koje treba da bude validirano mora da sadrži nekoliko data atributa od kojih je najbitniji pattern regularnog izraza po kom se određuje validnost tog polja.



```
const selector = {
  field: '[data-validate]',
  errorPlaceholder: '.validation-error-placeholder'
}

const defaultRules = { required: null, pattern: '(.*?)"', patternMessage: '' }

const extractField = field => {
  const $field = $(field)
  const { validate, ...rules } = $field.data()
  return {
    rules: { ...defaultRules, ...rules },
    $el: $field
  }
}

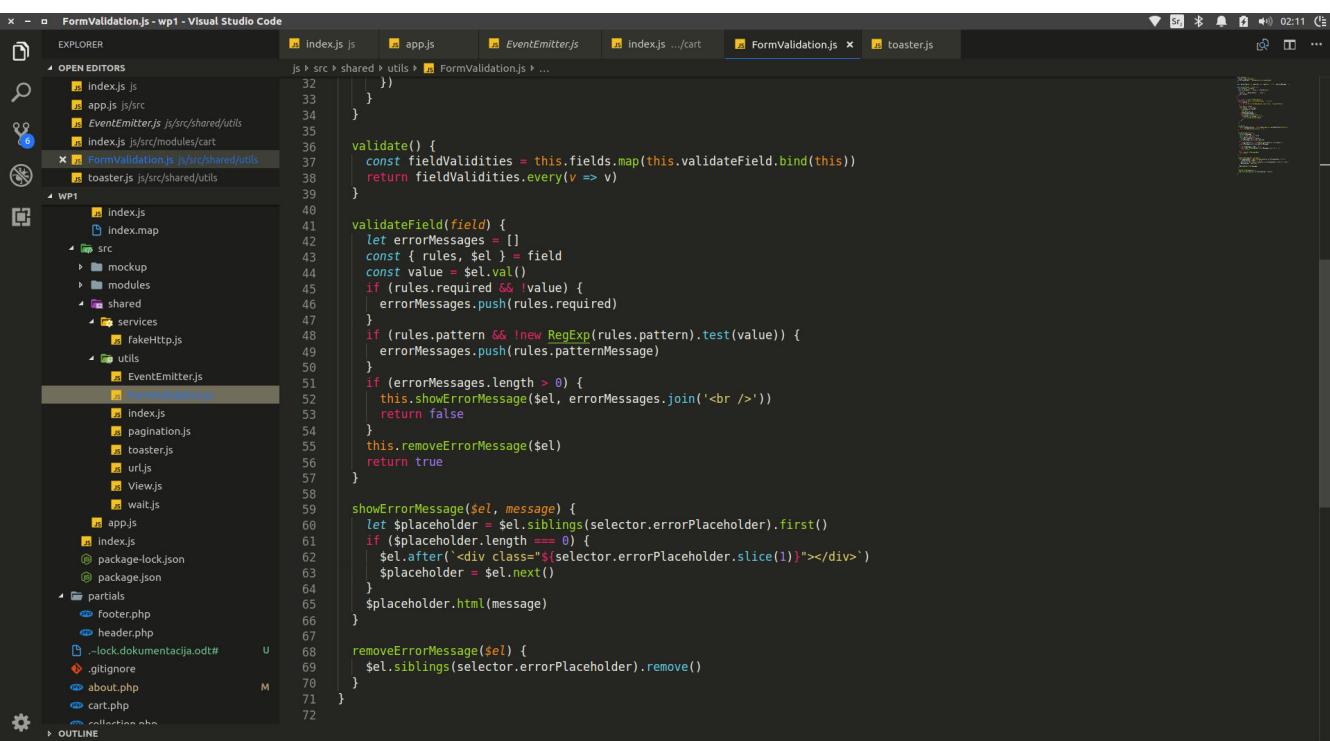
export default class FormValidation {
  constructor({ $form, validateOnChange = true }) {
    if (!$form) {
      throw new Error('FormValidation constructor - Invalid $form')
    }
    this.$form = $form
    this.fields = this.$form
      .find(selector.field)
      .toArray()
      .map(extractField)
    if (validateOnChange) {
      this.fields.forEach(field => {
        field.$el.on('change', () => {
          this.validateField(field)
        })
      })
    }
  }

  validate() {
    const fieldValidities = this.fields.map(this.validateField.bind(this))
    return fieldValidities.every(v => v)
  }

  validateField(field) {
    let errorMessages = []
    const { rules, $el } = field
    const value = $el.val()
    if (rules.required && !value) {
      errorMessages.push(rules.required)
    }
    if (rules.pattern && !new RegExp(rules.pattern).test(value)) {
      errorMessages.push(rules.patternMessage)
    }
    if (errorMessages.length > 0) {
      this.showErrorMessage($el, errorMessages.join('<br>'))
      return false
    }
    this.removeErrorMessage($el)
    return true
  }

  showErrorMessage($el, message) {
    let $placeholder = $el.siblings(selector.errorPlaceholder).first()
    if ($placeholder.length === 0) {
      $el.after(`<div class="${selector.errorPlaceholder.slice(1)}"></div>`)
      $placeholder = $el.next()
    }
    $placeholder.html(message)
  }

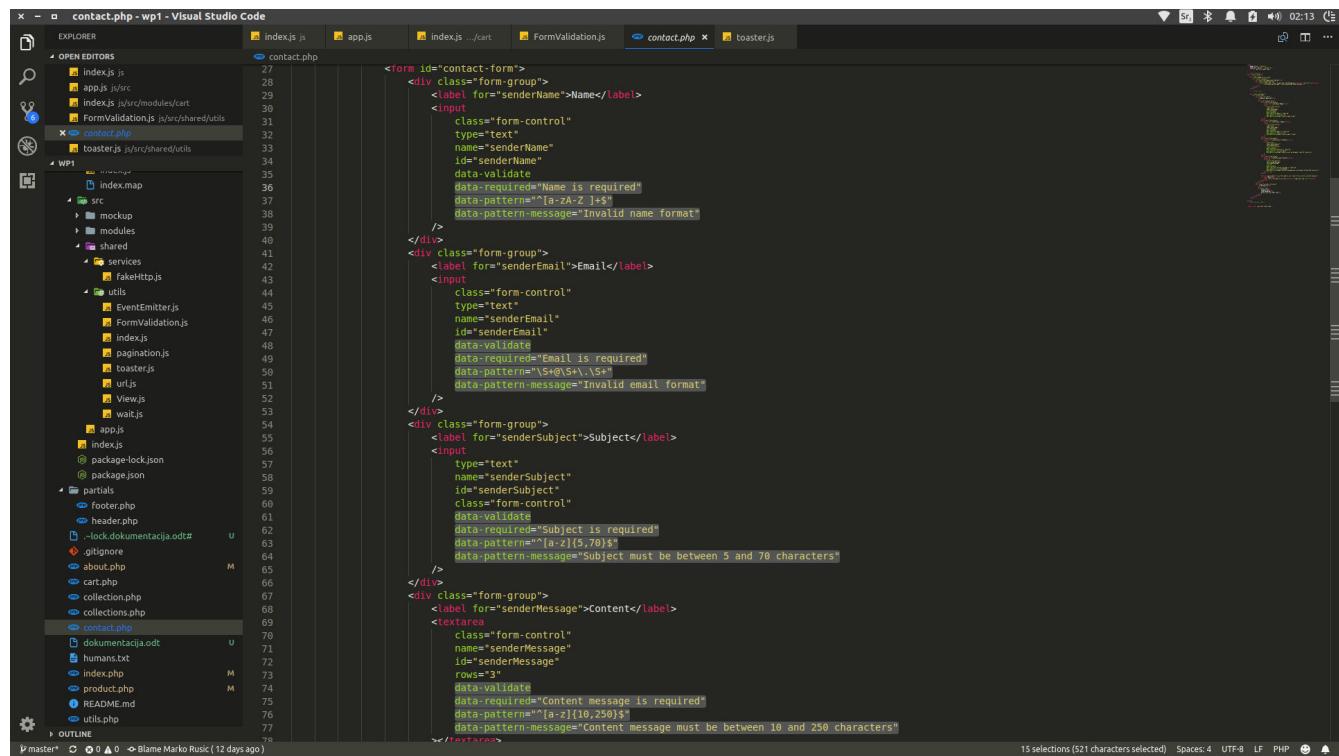
  removeErrorMessage($el) {
    $el.siblings(selector.errorPlaceholder).remove()
  }
}
```



```
showErrorMessage($el, message) {
  let $placeholder = $el.siblings(selector.errorPlaceholder).first()
  if ($placeholder.length === 0) {
    $el.after(`<div class="${selector.errorPlaceholder.slice(1)}"></div>`)
    $placeholder = $el.next()
  }
  $placeholder.html(message)
}

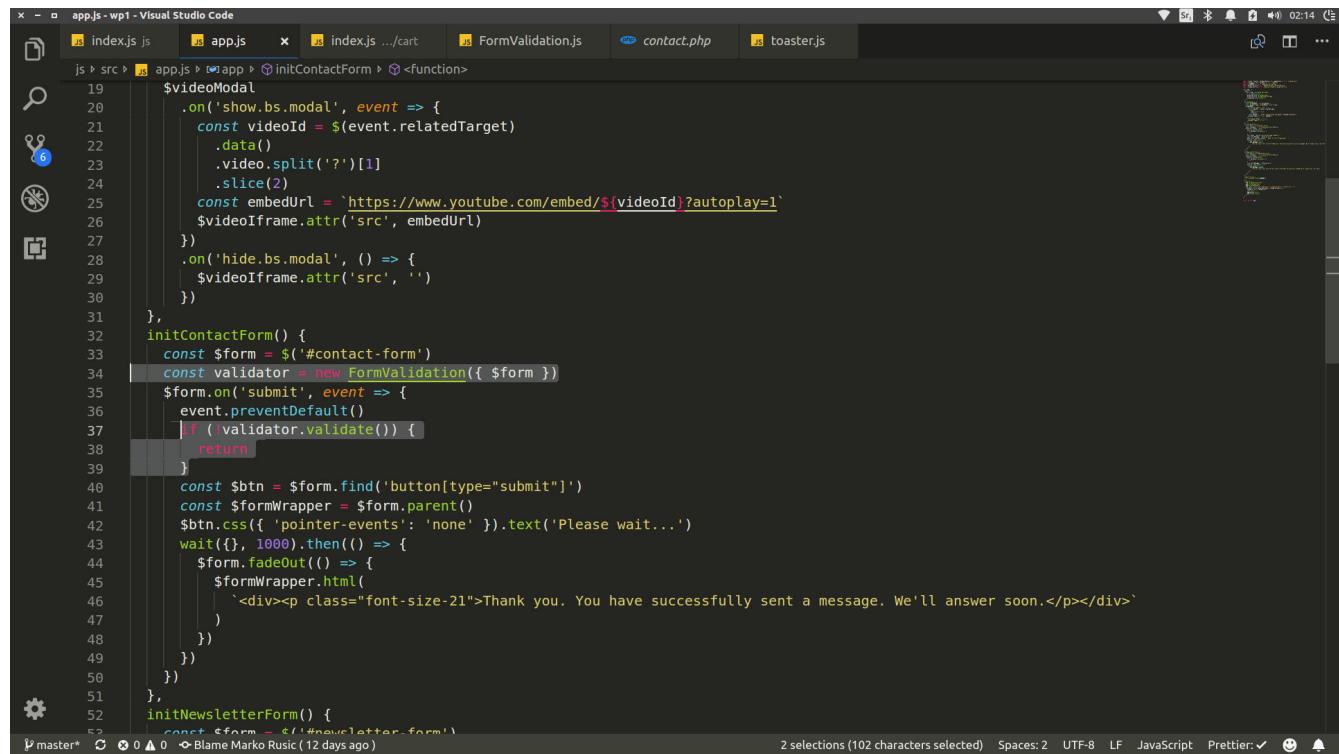
removeErrorMessage($el) {
  $el.siblings(selector.errorPlaceholder).remove()
}
```

## Primer korišćenja (Kontakt forma)



```
<form id="contact-form">
    <div class="form-group">
        <label for="senderName">Name</label>
        <input
            class="form-control"
            type="text"
            name="senderName"
            id="senderName"
            data-validate
            data-required="Name is required"
            data-pattern="^a-zA-Z ]+$"
            data-pattern-message="Invalid name format"
        />
    </div>
    <div class="form-group">
        <label for="senderEmail">Email</label>
        <input
            class="form-control"
            type="text"
            name="senderEmail"
            id="senderEmail"
            data-validate
            data-required="Email is required"
            data-pattern="^\w+@\w+\.\w+$"
            data-pattern-message="Invalid email format"
        />
    </div>
    <div class="form-group">
        <label for="senderSubject">Subject</label>
        <input
            type="text"
            name="senderSubject"
            id="senderSubject"
            class="form-control"
            data-validate
            data-required="Subject is required"
            data-pattern="^a-zA-Z ]{5,70}$"
            data-pattern-message="Subject must be between 5 and 70 characters"
        />
    </div>
    <div class="form-group">
        <label for="senderMessage">Content</label>
        <textarea
            class="form-control"
            name="senderMessage"
            id="senderMessage"
            rows="3"
            data-validate
            data-required="Content message is required"
            data-pattern="^a-zA-Z ]{10,250}$"
            data-pattern-message="Content message must be between 10 and 250 characters"
        />
    </div>

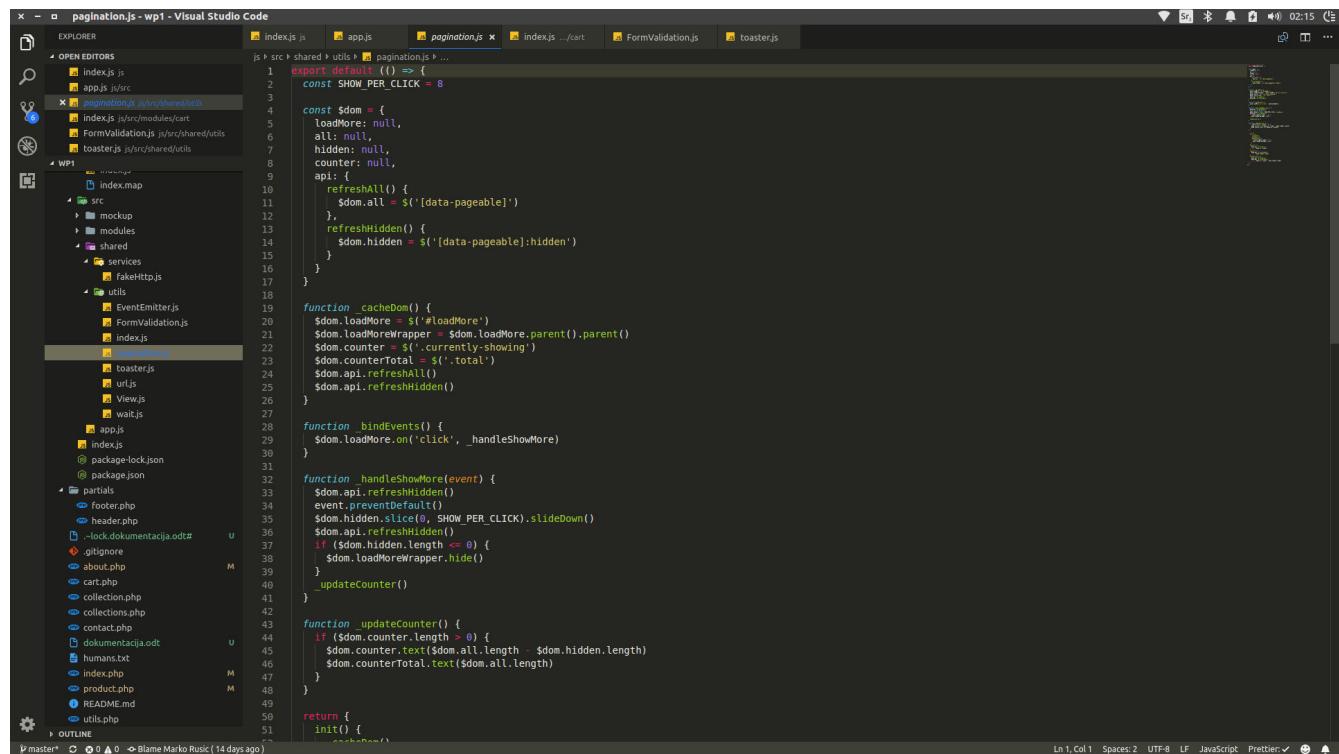
```



```
$videoModal
    .on('show.bs.modal', event => {
        const videoId = $(event.relatedTarget)
        .data()
        .video.split('?')[1]
        .slice(2)
        const embedUrl = `https://www.youtube.com/embed/${videoId}?autoplay=1`
        $videoIframe.attr('src', embedUrl)
    })
    .on('hide.bs.modal', () => {
        $videoIframe.attr('src', '')
    })
},
initContactForm() {
    const $form = $('#contact-form')
    const validator = new FormValidation({ $form })
    $form.on('submit', event => {
        event.preventDefault()
        if (!validator.validate()) {
            return
        }
        const $btn = $form.find('button[type="submit"]')
        const $formWrapper = $form.parent()
        $btn.css({ 'pointer-events': 'none' }).text('Please wait...')
        wait({}, 1000).then(() => {
            $form.fadeOut(() => {
                $formWrapper.html(
                    `<div><p class="font-size-21">Thank you. You have successfully sent a message. We'll answer soon.</p></div>`
                )
            })
        })
    })
},
initNewsletterForm() {
    const $form = $('#newsletter-form')
```

### 5.3.3 pagination.js

Koristi se za parcijalno prikazivanje proizvoda.



```
export default () => {
  const SHOW_PER_CLICK = 8

  const $dom = {
    loadMore: null,
    all: null,
    hidden: null,
    counter: null,
    api: {
      refreshAll() {
        $dom.all = $('[data-pageable]')
      },
      refreshHidden() {
        $dom.hidden = $('[data-pageable]:hidden')
      }
    }
  }

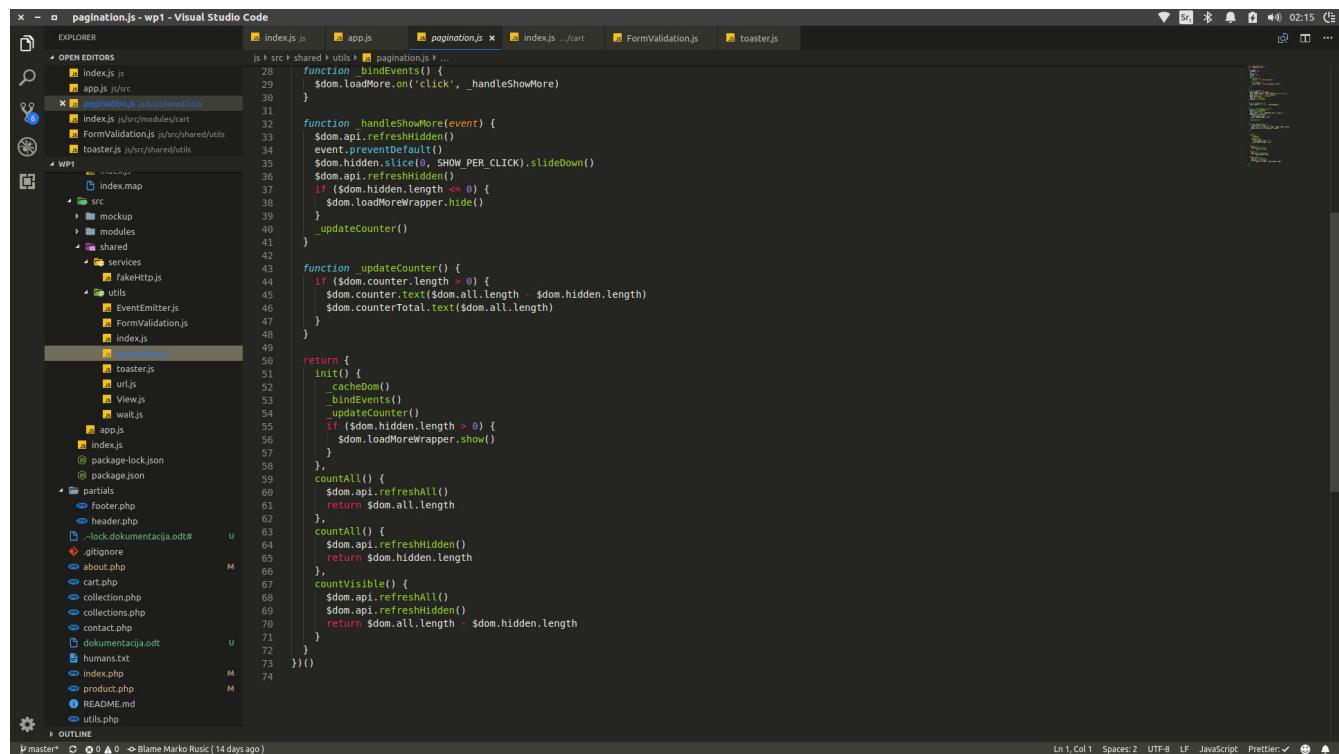
  function cacheDom() {
    $dom.loadMore = $('#loadMore')
    $dom.loadMoreWrapper = $dom.loadMore.parent().parent()
    $dom.counter = $('currently-showing')
    $dom.counterTotal = $('.total')
    $dom.api.refreshAll()
    $dom.api.refreshHidden()
  }

  function bindEvents() {
    $dom.loadMore.on('click', _handleShowMore)
  }

  function handleShowMore(event) {
    $dom.api.refreshHidden()
    event.preventDefault()
    $dom.hidden.slice(0, SHOW_PER_CLICK).slideDown()
    $dom.api.refreshHidden()
    if ($dom.hidden.length <= 0) {
      $dom.loadMoreWrapper.hide()
    }
    _updateCounter()
  }

  function updateCounter() {
    if ($dom.counter.length > 0) {
      $dom.counter.text($dom.all.length - $dom.hidden.length)
      $dom.counterTotal.text($dom.all.length)
    }
  }

  return {
    init() {
      cacheDom()
    },
    ...
  }
}
```



```
function bindEvents() {
  $dom.loadMore.on('click', _handleShowMore)
}

function handleShowMore(event) {
  $dom.api.refreshHidden()
  event.preventDefault()
  $dom.hidden.slice(0, SHOW_PER_CLICK).slideDown()
  $dom.api.refreshHidden()
  if ($dom.hidden.length <= 0) {
    $dom.loadMoreWrapper.hide()
  }
  _updateCounter()
}

function updateCounter() {
  if ($dom.counter.length > 0) {
    $dom.counter.text($dom.all.length - $dom.hidden.length)
    $dom.counterTotal.text($dom.all.length)
  }
}

return {
  init() {
    cacheDom()
    bindEvents()
    updateCounter()
    if ($dom.hidden.length > 0) {
      $dom.loadMoreWrapper.show()
    }
  },
  countAll() {
    $dom.api.refreshAll()
    return $dom.all.length
  },
  countAll() {
    $dom.api.refreshHidden()
    return $dom.hidden.length
  },
  countVisible() {
    $dom.api.refreshAll()
    $dom.api.refreshHidden()
    return $dom.all.length - $dom.hidden.length
  }
}
})()
```

## 5.3.4 toaster.js

jQuery plugin iskorišćen za notifikacije pri dodavanju proizvoda u korpu.

```
1 // Add toaster to array
2 toaster.push(..., this)
3 const indexOf = (function() {
4     // ...
5     if (typeof Array.prototype.indexOf !== 'function') {
6         Array.prototype.indexOf = function(searchElement, fromIndex) {
7             for (var i = fromIndex || 0, j = this.length; i < j; i += 1) {
8                 if (searchElement === undefined || searchElement === null) {
9                     if (this[i] === searchElement) {
10                         return i
11                     }
12                 } else if (this[i] === searchElement) {
13                     return i
14                 }
15             }
16             return -1
17         }
18     })
19 })([])
20 // ...
21 ;(function($, undefined) {
22     var toaster = {
23         getToaster: function() {
24             var toaster = $('#') + settings.toaster.id
25             if (toaster.length < 1) {
26                 toaster = $(settings.toaster.template)
27                     .attr('id', settings.toaster.id)
28                     .css(settings.toaster.css)
29                     .addClass(settings.toaster['class'])
30             }
31             if (
32                 settings.stylesheet &&
33                 $('link[href=' + settings.stylesheet + ']').length
34             ) {
35                 $('head').appendTo(
36                     $`<link rel="stylesheet" href="${settings.stylesheet}">`)
37             }
38             $(settings.toaster.container).append(toaster)
39         }
40         return toaster
41     },
42     notify: function(title, message, priority) {
43         var $toaster = this.getToaster()
44         var delimiter = title && message ? settings.toast.defaults.delimiter : ''
45         var $toast = $(`

` + title + `

`)
46         $toast.append(`<p>${message}</p>`)
47         $toast.append(`<span class="close">×</span>`)
48         $toaster.append($toast)
49     }
50 })(jQuery)
```

```
notify: function(title, message, priority) {
    var $toaster = $t('toaster')
    var delimiter = title + message ? settings.toast.defaults.delimiter : ''
    var $toast = $(`

${title}



${message}

`)
    $toast.append(`<div style="display:flex; justify-content:space-between; align-items:center; margin-bottom: 5px;">
        ${$t('close')}
        ${$t('copy')}
    </div>`)
    $toaster.append($toast)
    $toast.show()
    if (settings.debug && window.console) {
        console.log(toast)
    }
    $toaster.append(settings.toast.display($toast))
}

if (settings.donotdismiss.indexOf(priority) === -1) {
    var timeout =
        typeof settings.timeout === 'number'
            ? settings.timeout
            : typeof settings.timeout === 'object'
                ? priority in settings.timeout
                    ? settings.timeout[priority]
                    : 1500
                : setTimeout(function() {
                    settings.toast.remove($toast, function() {
                        $toast.remove()
                    })
                }, timeout)
}
var defaults = {
    toaster: {
        id: 'toaster',
        container: 'body',
        template: '<div></div>',
        class: 'toaster',
        css: {
            position: 'fixed',
            top: 0,
            left: 0,
            width: '100%',
            height: '100%',
            overflow: 'auto',
            scrollBehavior: 'smooth',
            transition: 'all 0.3s ease-in-out',
            backfaceVisibility: 'hidden',
            transform: 'translateZ(0)'
        }
    }
}
```

toaster.js - wp1 - Visual Studio Code

OPEN EDITORS

index.js.js app.js pagination.js index.js .../cart FormValidation.js toaster.js

```
var defaults = {
  toaster: {
    id: 'toaster',
    container: 'body',
    template: '<div></div>',
    class: 'toaster',
    css: {
      position: 'fixed',
      top: '75px',
      right: '10px',
      width: '300px',
      zIndex: 50000
    }
  },
  toast: {
    template: '<div class="alert alert-%priority% alert-dismissible" role="alert">' +
    '<button type="button" class="close" data-dismiss="alert">%times;</button>' +
    '<span aria-hidden="true">%&times;</span>' +
    '<span class="sr-only">Close</span>' +
    '</button>' +
    '<span class="title">%</span>%delimiter% <span class="message">%</span>' +
    '</div>',
    defaults: {
      title: '',
      priority: 'success',
      delimiter: ':'
    },
    css: {},
    cssm: {},
    csst: { fontWeight: 'bold' },
    fade: 'slow',
    display: function($toast) {
      return $toast.fadeIn(settings.toast.fade);
    },
    remove: function($toast, callback) {
      return $toast.animate(
        {
          opacity: '0',
          padding: '0px',
          margin: '0px',
          height: '0px'
        },
        {
          duration: settings.toast.fade,
        }
      );
    }
  }
};
```

Ln 105, Col 13 Spaces: 2 UTF-8 LF JavaScript Prettier: ✓

toaster.js - wp1 - Visual Studio Code

OPEN EDITORS

index.js.js app.js pagination.js index.js .../cart FormValidation.js toaster.js

```
remove: function($toast, callback) {
  return $toast.animate(
    {
      opacity: '0',
      padding: '0px',
      margin: '0px',
      height: '0px'
    },
    {
      duration: settings.toast.fade,
      complete: callback
    }
  );
}

debug: false,
timeout: 4500,
stylesheet: null,
donotdismiss: []

var settings = {};
$.extend(settings, defaults);

$.toaster = function(options) {
  if (typeof options === 'object') {
    if ('settings' in options) {
      settings = $.extend(true, settings, options.settings);
    } else {
      var values = Array.prototype.slice.call(arguments, 0);
      var labels = ['message', 'title', 'priority'];
      options = {};
      for (var i = 0, l = values.length; i < l; i += 1) {
        options[labels[i]] = values[i];
      }
    }
  }

  var title =
    'title' in options && typeof options.title === 'string'
    ? options.title
    : settings.toast.defaults.title;
  var message = 'message' in options ? options.message : null;
  var priority =
    'priority' in options && typeof options.priority === 'string'
    ? options.priority
    : settings.toast.defaults.priority;

  if (message === null) {
```

Ln 105, Col 13 Spaces: 2 UTF-8 LF JavaScript Prettier: ✓

```

145     },
146     debug: false,
147     timeout: 4500,
148     stylesheet: null,
149     dontDismiss: []
150   }
151 }
152
153 var settings = {};
154 $.extend(settings, defaults);
155
156 $.toaster = function(options) {
157   if (typeof options === 'object') {
158     if ('settings' in options) {
159       settings = $.extend(true, settings, options.settings)
160     } else {
161       var values = Array.prototype.slice.call(arguments, 0)
162       var labels = ['message', 'title', 'priority']
163       options = {}
164       for (var i = 0, l = values.length; i < l; i++) {
165         options[labels[i]] = values[i]
166       }
167     }
168   }
169
170   var title =
171     'title' in options && typeof options.title === 'string'
172     ? options.title
173     : settings.toast.defaults.title
174   var message = 'message' in options ? options.message : null
175   var priority =
176     'priority' in options && typeof options.priority === 'string'
177     ? options.priority
178     : settings.toast.defaults.priority
179
180   if (message !== null) {
181     toasting.notify(title, message, priority)
182   }
183
184   $.toaster.reset = function() {
185     settings = {};
186     $.extend(settings, defaults)
187   }
188
189 })(jQuery)
190
191

```

## 5.3.4 url.js

Pomoćni objekat za parsiranje i čitanje URL parametara.

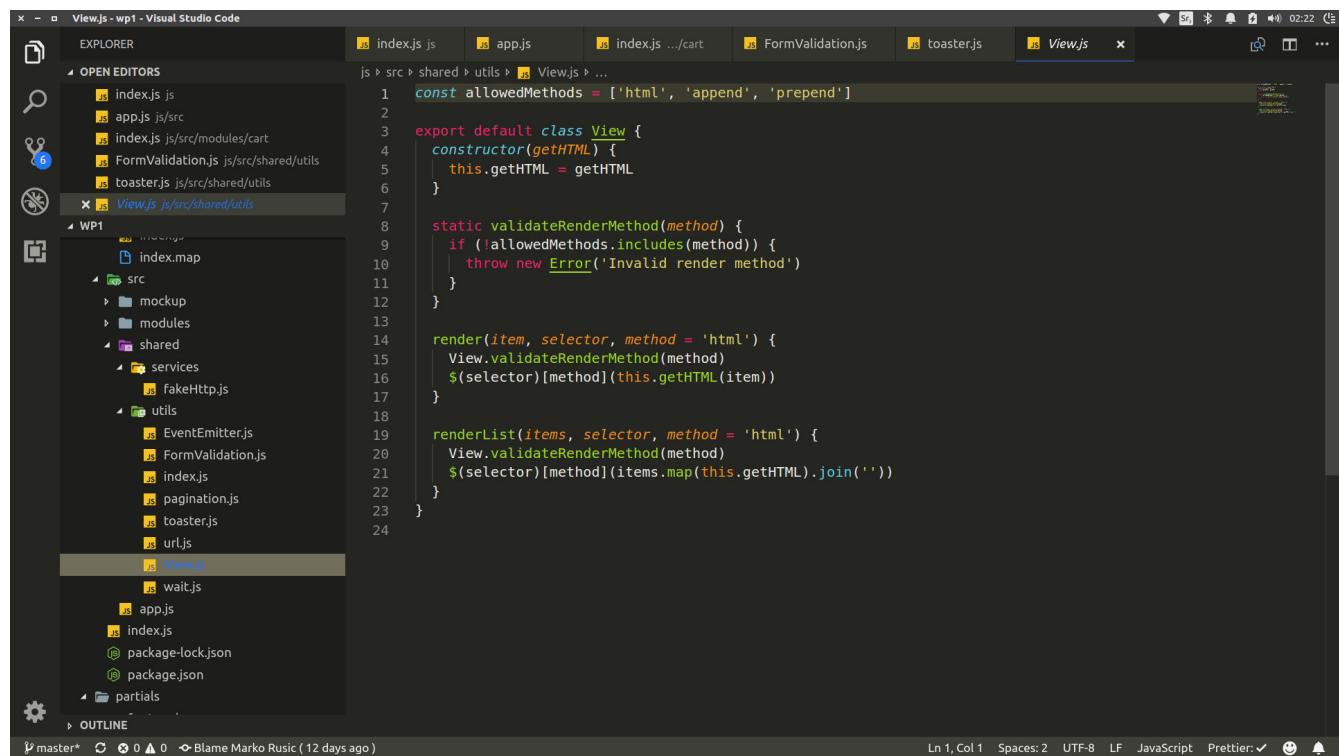
```

1  export default {
2   getParams() {
3     const search = location.search.substring(1)
4     try {
5       return JSON.parse(
6         `{"` +
7           decodeURI(search)
8             .replace(/\//g, '\\\\')
9             .replace(/\&/g, ',')
10            .replace(/=/g, '":') +
11          `"}`
12        )
13     } catch {
14       return {}
15     }
16   },
17   getParam(param) {
18     return this.getParams()[param]
19   }
20 }
21

```

### 5.3.5 View.js

Klasa koja se koristi za prikazivanje(render) podatka ili list podataka u DOM.



The screenshot shows the Visual Studio Code interface with the title bar "View.js - wp1 - Visual Studio Code". The left sidebar displays the project structure under "EXPLORER". The main editor area shows the code for "View.js". The status bar at the bottom indicates "Ln 1, Col 1" and "Spaces: 2".

```
const allowedMethods = ['html', 'append', 'prepend']

export default class View {
  constructor(getHTML) {
    this.getHTML = getHTML
  }

  static validateRenderMethod(method) {
    if (!allowedMethods.includes(method)) {
      throw new Error('Invalid render method')
    }
  }

  render(item, selector, method = 'html') {
    View.validateRenderMethod(method)
    $(selector)[method](this.getHTML(item))
  }

  renderList(items, selector, method = 'html') {
    View.validateRenderMethod(method)
    $(selector)[method](items.map(this.getHTML).join(''))
  }
}
```

### 5.3.6 wait.js

The screenshot shows the Visual Studio Code interface with the title bar "walt.js - wp1 - Visual Studio Code". The Explorer sidebar on the left shows a project structure under "WP1" with folders like "src", "mockup", "modules", and "shared". Inside "shared", there's a "services" folder containing "fakeHttp.js" and a "utils" folder containing "EventEmitter.js", "FormValidation.js", "index.js", "pagination.js", "toaster.js", "url.js", "View.js", and "wait.js". The "wait.js" file is currently selected and open in the editor tab. The code in the editor is:

```
1 export default (data, time = 0) =>
2   new Promise(resolve => {
3     setTimeout(() => {
4       resolve(data)
5     }, time)
6   })
7
```

The status bar at the bottom shows "Ln 1, Col 1" and "Spaces: 2".

### 5.3.7 fakeHttp.js

Objekat koji mokapuje http zahteve i pruza sličan interfejs kao popularna axios biblioteka.

The screenshot shows the Visual Studio Code interface with the title bar "fakeHttp.js - wp1 - Visual Studio Code". The Explorer sidebar on the left shows a project structure under "WP1" with folders like "src", "mockup", "modules", and "shared". Inside "shared", there's a "services" folder containing "fakeHttp.js" and a "utils" folder containing "EventEmitter.js", "FormValidation.js", "index.js", "pagination.js", "toaster.js", "url.js", "View.js", and "wait.js". The "fakeHttp.js" file is currently selected and open in the editor tab. The code in the editor is:

```
1 const fakeRequest = (...args) =>
2   new Promise(resolve => {
3     setTimeout(() => {
4       resolve(...args)
5     }, 500)
6   })
7
8 export default {
9   get: fakeRequest,
10  post: fakeRequest,
11  put: fakeRequest,
12  delete: fakeRequest
13 }
14
```

The status bar at the bottom shows "Ln 14, Col 1" and "Spaces: 2".

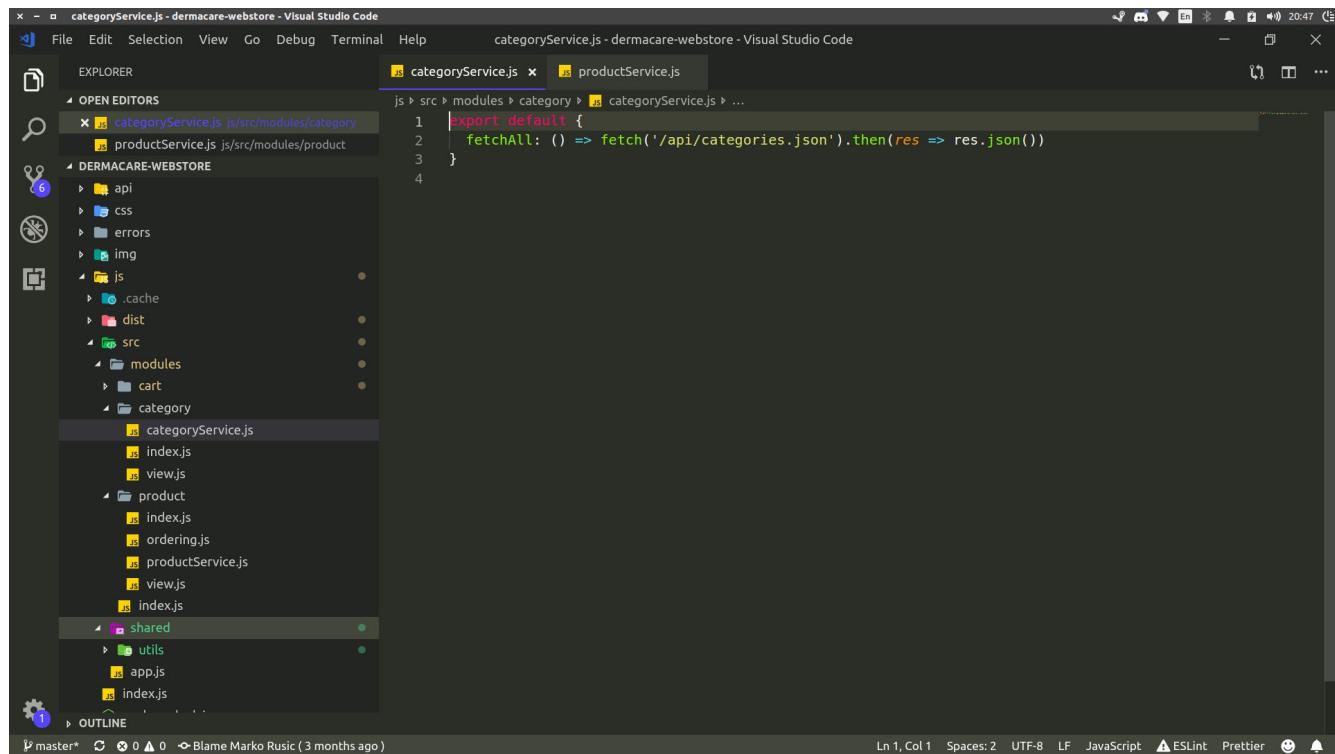
## 5.4 Category modul

Dinamički prikazuje kategorije(kolekcije) na collections.php strani.

```
x - □ markorusic@markorusic-Inspiron-5570: ~/skolica/wp/wp
→ horizont_app git:(TvProgramScreen) X cd
→ ~
→ ~ cd skolica/wp/wp1/
→ wp1 git:(master) X tree js/src/modules/category
js/src/modules/category
└── categoryService.js
└── index.js
└── view.js
0 directories, 3 files
→ wp1 git:(master) X █
```

## 5.4.1 categoryService.js

Servis koji povlači podatke iz JSON fajla na serveru korišćenjem fetch api-ja.



## 5.4.2 view.js

Implementacija shared View kalse.

The screenshot shows the Visual Studio Code interface with the following details:

- Title Bar:** view.js - wp1 - Visual Studio Code
- File Explorer (Left):** Shows the project structure:
  - WP1
  - src
    - cache
    - dist
    - index.js
    - index.map
    - mockup
    - modules
      - cart
      - category
        - categoryService.js
        - index.js
        - view.js
    - product
    - index.js
    - shared
      - app.js
      - index.js
    - package-lock.json
    - package.json
  - partials
- Code Editor (Right):** The file `view.js` is open, showing the following code:

```
1 import View from '..../shared/utils/View'
2
3 export default new View(
4   { id, name, slug, main_photo }) => `
5     <div class="col-6 col-md-4 col-lg-3">
6       <article class="product-preview-article">
7         <div class="position-relative product-image-preview d-flex justify-content-center align-items-center">
8           
9           <div class="image-overlay position-absolute"></div>
10          <div class="position-absolute box-block d-flex justify-content-center align-items-center">
11            <a class="text-center full-wh flex-center-col" href="collection.php?slug=${slug}">
12              ${name}
13            </a>
14          </div>
15        </div>
16        <div class="text-center">
17          <h4><a href="collection.php?slug=${slug}">${name}</a></h4>
18        </div>
19      </article>
20    </div>
21  )
22 }
```
- Status Bar (Bottom):** master\* 0 0 0 → Blame Marko Rusic ( 14 days ago ) Ln 1, Col 1 Spaces: 4 UTF-8 LF JavaScript Prettier:✓

### 5.4.3 index.js

Entry fajl Category modula.

The screenshot shows the Visual Studio Code interface with the following details:

- Title Bar:** Index.js - wp1 - Visual Studio Code
- File Explorer (Left):** Shows the project structure under 'WP1'. Key folders include 'src' (containing 'dist', 'mockup', 'modules', 'cart', and 'category'), 'shared' (containing 'app.js'), and 'partials'. Files like 'index.js', 'view.js', 'categoryService.js', and 'index.map' are visible.
- Code Editor (Right):** Displays the content of 'index.js' located at 'src/modules/category'. The code imports 'view' from './view' and defines an 'init' function that calls 'view.renderList(categories, '#collection-row')'.
- Status Bar (Bottom):** Shows the file path 'js > src > modules > category > index.js > init', line 'Ln 6, Col 49 (14 selected)', and other settings like 'Spaces: 2', 'UTF-8', 'LF', 'JavaScript', and 'Prettier'.

## 5.5 Product modul

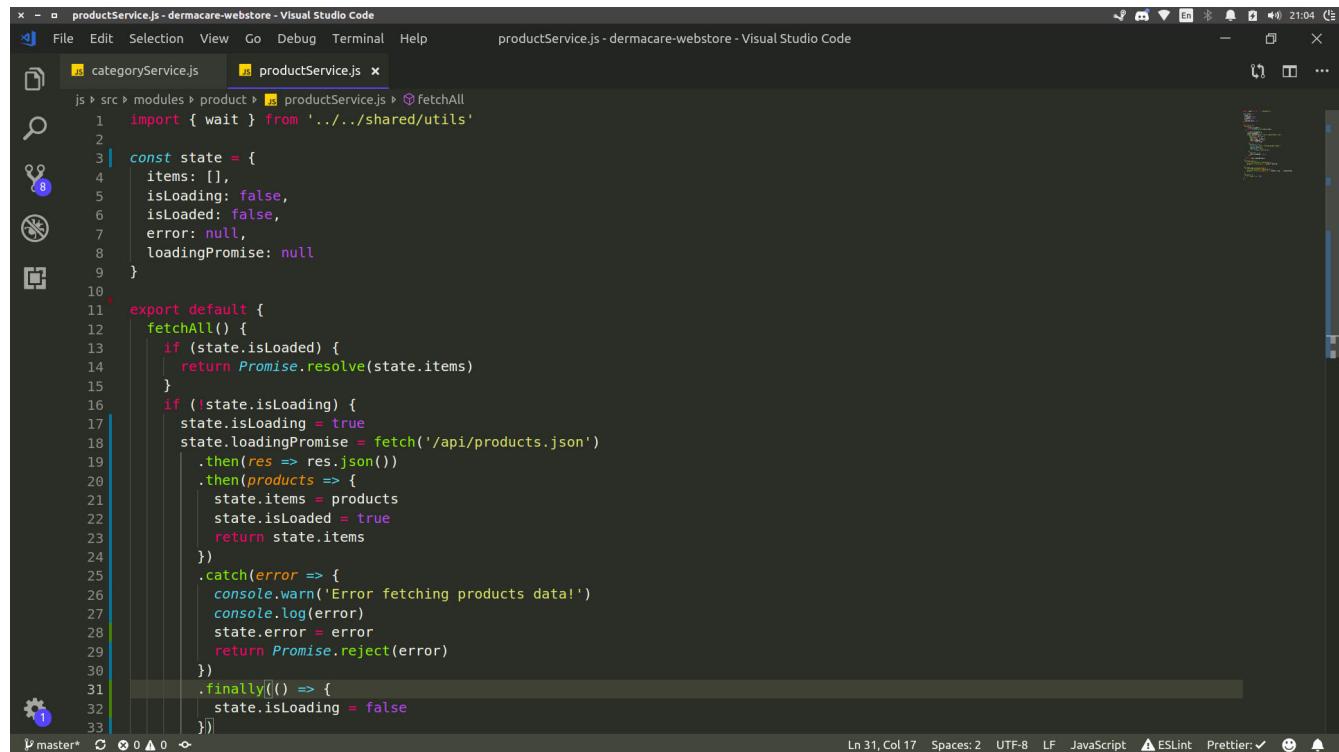
Dinamiči prikazuje proizvode početnoj stani (index.php) i strani pojedinačne kolekcije (collection.php), takodje je zadužen za sortiranje i parcijalni prikaz proizvoda.

```
→ wp1 git:(master) ✘ tree js/src/modules/product
js/src/modules/product
└── index.js
    ├── ordering.js
    ├── productService.js
    └── view.js

0 directories, 4 files
→ wp1 git:(master) ✘
```

## 5.5.1 productService.js

Servis koji povlači podatke o proizvodima iz JSON fajla sa servera.

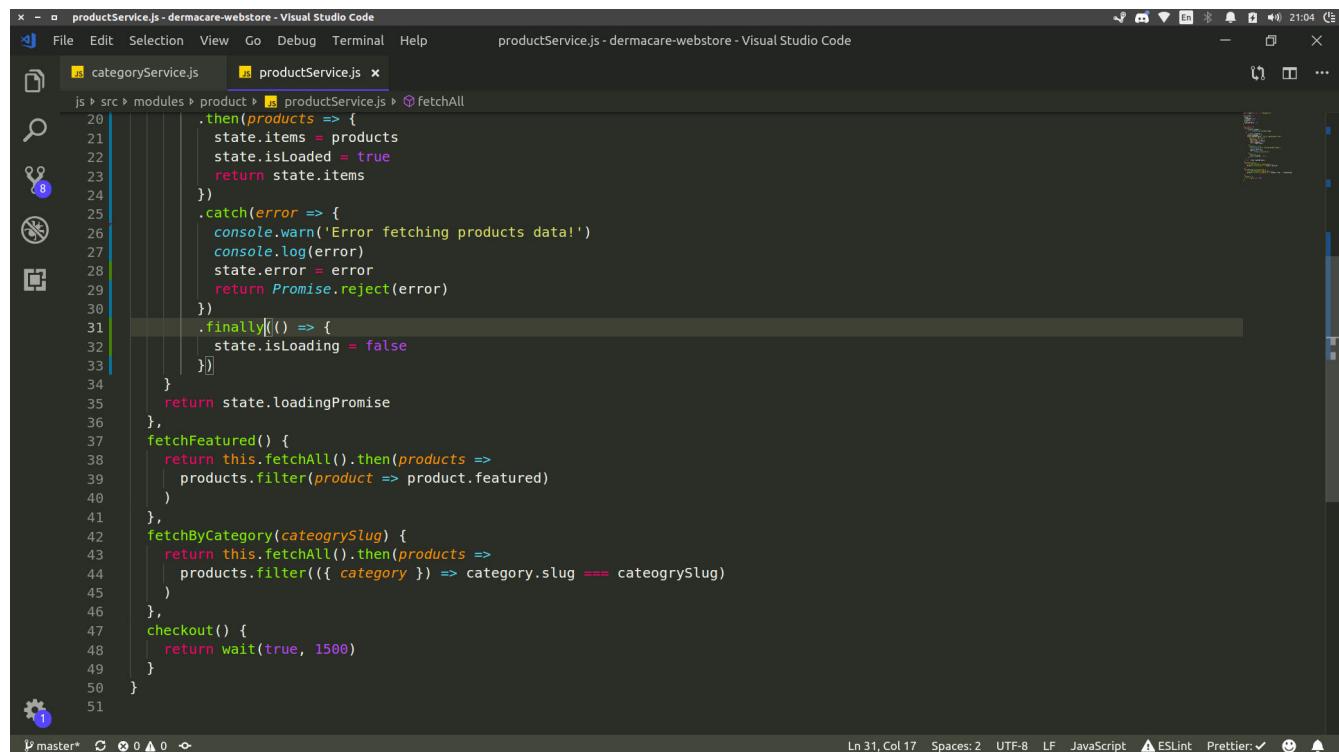


The screenshot shows the Visual Studio Code interface with the productService.js file open. The code defines a state object and a fetchAll function that uses a Promise to fetch products from a JSON file. It includes error handling and a finally block to set isLoading to false.

```
const state = {
  items: [],
  isLoading: false,
  isLoaded: false,
  error: null,
  loadingPromise: null
}

export default {
  fetchAll() {
    if (state.isLoaded) {
      return Promise.resolve(state.items)
    }

    if (!state.isLoading) {
      state.isLoading = true
      state.loadingPromise = fetch('/api/products.json')
        .then(res => res.json())
        .then(products => {
          state.items = products
          state.isLoaded = true
          return state.items
        })
        .catch(error => {
          console.warn('Error fetching products data!')
          console.log(error)
          state.error = error
          return Promise.reject(error)
        })
        .finally(() => {
          state.isLoading = false
        })
    }
  }
}
```



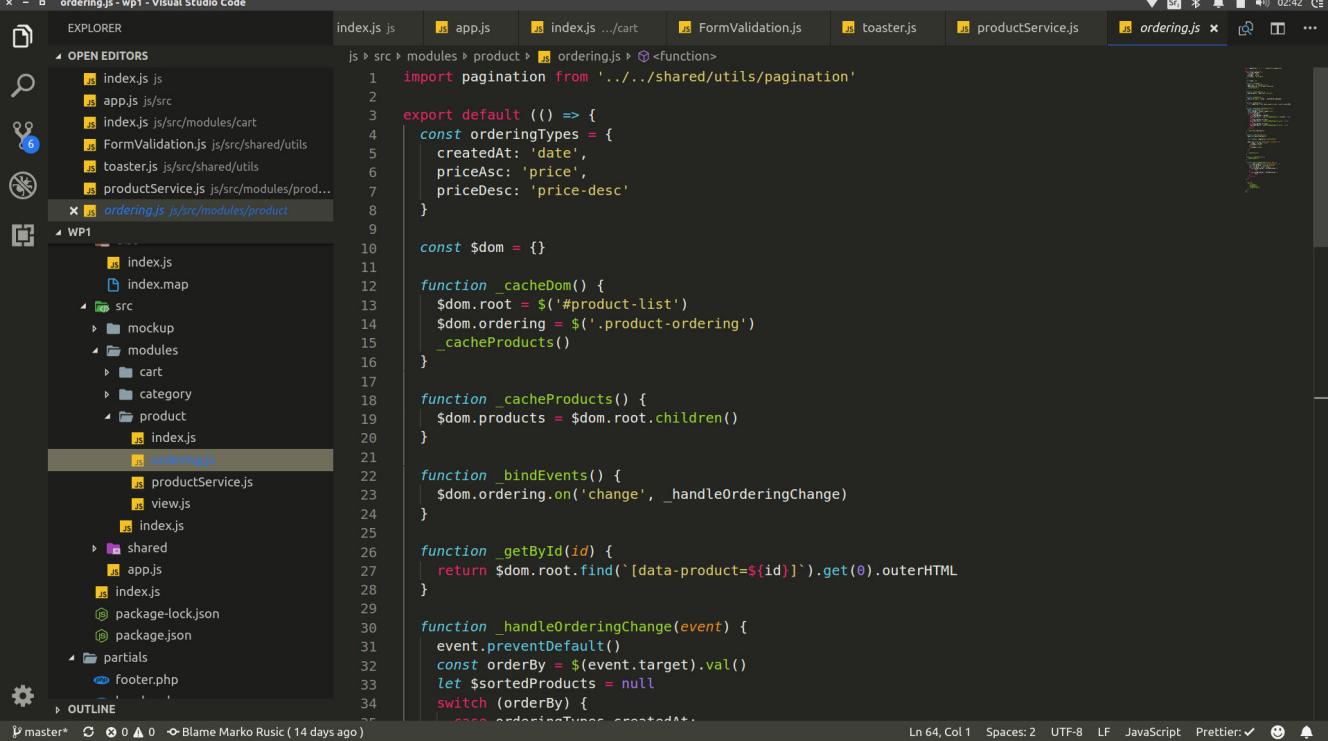
The screenshot shows the Visual Studio Code interface with the productService.js file open. The code has been expanded to include fetchFeatured, fetchByCategory, and checkout methods. The fetchAll method now includes a filter for featured products and a filter for a specific category. The checkout method returns a promise that waits 1500ms.

```
.then(products => {
  state.items = products
  state.isLoaded = true
  return state.items
})
.catch(error => {
  console.warn('Error fetching products data!')
  console.log(error)
  state.error = error
  return Promise.reject(error)
})
.finally(() => {
  state.isLoading = false
})

return state.loadingPromise
},
fetchFeatured() {
  return this.fetchAll().then(products =>
    products.filter(product => product.featured)
  )
},
fetchByCategory(categorySlug) {
  return this.fetchAll().then(products =>
    products.filter({ category }) => category.slug === categorySlug
  )
},
checkout() {
  return wait(true, 1500)
}
}
```

## 5.5.2 ordering.js

Sortiranje proizvoda po izabranom polju.



The screenshot shows the Visual Studio Code interface with the file `ordering.js` open in the editor. The code implements a sorting logic for products based on user input. It includes functions for caching the DOM, binding events, and handling ordering changes. The `orderingTypes` object defines three sorting criteria: `createdAt`, `priceAsc`, and `priceDesc`.

```
import pagination from '../../../../../shared/utils/pagination'

export default () => {
  const orderingTypes = {
    createdAt: 'date',
    priceAsc: 'price',
    priceDesc: 'price-desc'
  }

  const $dom = {}

  function _cacheDom() {
    $dom.root = $('#product-list')
    $dom.ordering = $('.product-ordering')
    _cacheProducts()
  }

  function _cacheProducts() {
    $dom.products = $dom.root.children()
  }

  function _bindEvents() {
    $dom.ordering.on('change', _handleOrderingChange)
  }

  function _getById(id) {
    return $dom.root.find(`[data-product=${id}]`).get(0).outerHTML
  }

  function _handleOrderingChange(event) {
    event.preventDefault()
    const orderBy = $(event.target).val()
    let $sortedProducts = null
    switch (orderBy) {
      case orderingTypes.createdAt:
        $sortedProducts = _getSortedDomElements('createdAt', true)
        break
      case orderingTypes.priceAsc:
        $sortedProducts = _getSortedDomElements('price', false)
        break
      case orderingTypes.priceDesc:
        $sortedProducts = _getSortedDomElements('price', true)
        break
      default:
    }
    _render($sortedProducts)
  }

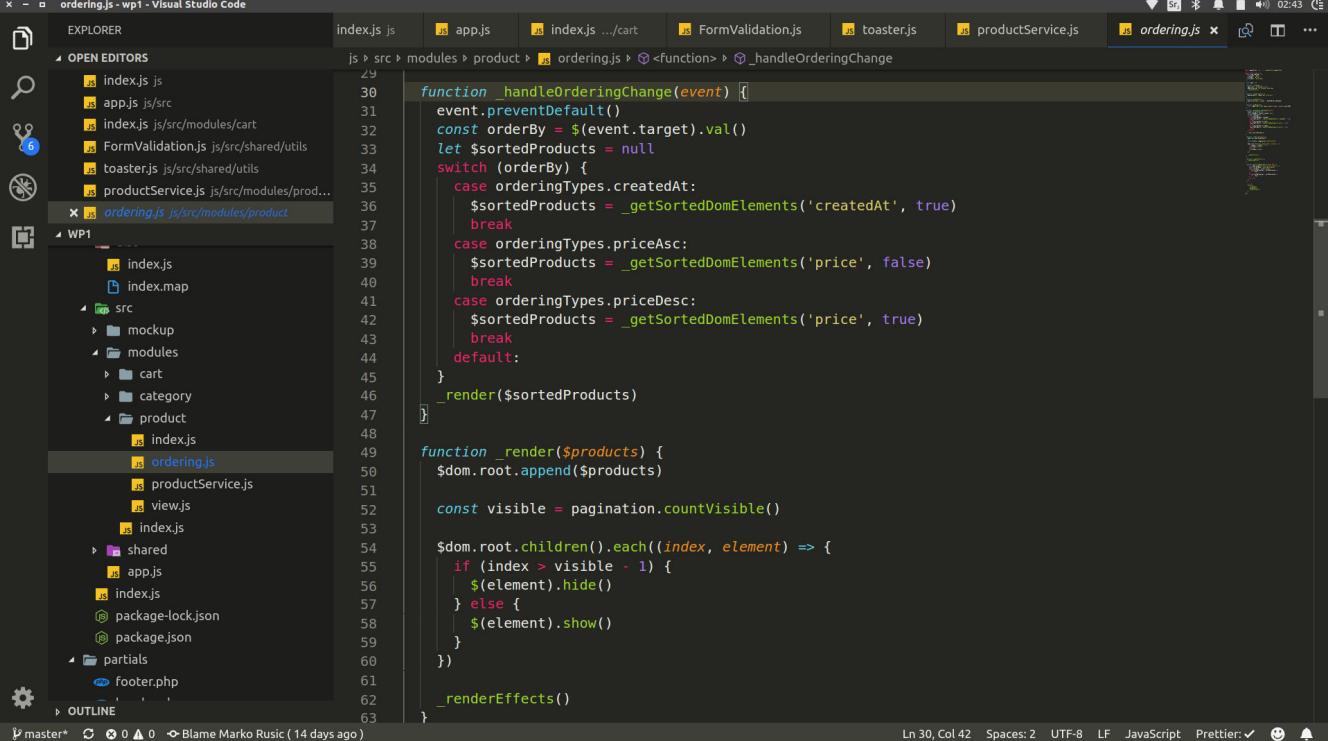
  function _render($products) {
    $dom.root.append($products)

    const visible = pagination.countVisible()

    $dom.root.children().each((index, element) => {
      if (index > visible - 1) {
        $(element).hide()
      } else {
        $(element).show()
      }
    })

    _renderEffects()
  }

  _cacheDom()
  _bindEvents()
}
```



The screenshot shows the Visual Studio Code interface with the file `ordering.js` open in the editor. The code has been extended to include logic for rendering effects after the products are sorted. This involves hiding elements beyond the current visible range and then showing them again. The `_renderEffects()` function is responsible for this behavior.

```
function _handleOrderingChange(event) {
  event.preventDefault()
  const orderBy = $(event.target).val()
  let $sortedProducts = null
  switch (orderBy) {
    case orderingTypes.createdAt:
      $sortedProducts = _getSortedDomElements('createdAt', true)
      break
    case orderingTypes.priceAsc:
      $sortedProducts = _getSortedDomElements('price', false)
      break
    case orderingTypes.priceDesc:
      $sortedProducts = _getSortedDomElements('price', true)
      break
    default:
  }
  _render($sortedProducts)
}

function _render($products) {
  $dom.root.append($products)

  const visible = pagination.countVisible()

  $dom.root.children().each((index, element) => {
    if (index > visible - 1) {
      $(element).hide()
    } else {
      $(element).show()
    }
  })

  _renderEffects()
}
```

```
ordering.js - wp1 - Visual Studio Code
EXPLORER OPEN EDITORS index.js.js app.js index.js.../cart FormValidation.js toaster.js productService.js ordering.js
js > src > modules > product > ordering.js > <Function> > _handleOrderingChange
index.js.js 62 _renderEffects()
app.js.js 63 }
index.js.js 64
index.js.js 65 function _renderEffects() {
index.js.js 66 _cacheProducts()
index.js.js 67 }
index.js.js 68
index.js.js 69 function _getSortedDomElements(prop, desc) {
index.js.js 70 return $dom.products.sort((current, next) => {
index.js.js 71   const currentData = $(current).data()
index.js.js 72   const nextData = $(next).data()
index.js.js 73   if (currentData[prop] < nextData[prop]) {
index.js.js 74     return desc ? 1 : -1
index.js.js 75   }
index.js.js 76   if (currentData[prop] > nextData[prop]) {
index.js.js 77     return desc ? -1 : 1
index.js.js 78   }
index.js.js 79   return 0
index.js.js 80 })
index.js.js 81
index.js.js 82 return {
index.js.js 83   init() {
index.js.js 84     _cacheDom()
index.js.js 85     _bindEvents()
index.js.js 86     _renderEffects()
index.js.js 87   }
index.js.js 88 }
index.js.js 89 }()
index.js.js 90 }()
index.js.js 91 }
```

Ln 30, Col 42 Spaces: 2 UTF-8 LF JavaScript Prettier: ✓

### 5.5.3 view.js

Implementacija shared View kalse.

```

import { CURRENCY } from '../cart/config'
import View from '../../shared/utils/View'

export default new View(
  ({ id, name, price, slug, main_photo, category, created_at }, index) => `
    <div class="col-6 col-md-4 col-lg-3" data-pageable data-product-id="${id}" data-price="$${price}" data-created-at="$${created_at}" style="display:$${index > 7 ? 'none' : 'block'}">
      <article class="product-preview-article">
        <div class="product-image-preview-position-relative">
          <a href="/product.php?slug=${slug}">
            
          </a>
          <button data-product-id="${id}" class="btn-add-to-cart d-flex justify-content-between preview-product-etc">
            <span>Add to cart</span>
            <span class="btn-add-to-cart-plus"></span>
          </button>
        </div>
        <div class="d-flex flex-column justify-content-md-between flex-md-row">
          <div>
            <h4><a href="/product.php?slug=${slug}">$${name}</a></h4>
            <h4>$${price} ${CURRENCY}</h4>
          </div>
          ${category &&
            ((() =>
              <h6 class="pb-1">
                <a style="color: inherit;" href="/collection.php?slug=${
                  category.slug
                }">
                  ${category.name}
                </a>
              </h6>
            ))()
          }
        </div>
      </article>
    </div>
  )
)

```

Blame Marko Rusic (14 days ago)

Ln 12, Col 33 Spaces: 2 UTF-8 LF JavaScript Prettier: ✓

## 5.5.4 index.js

Entry fajl Category modula.

```

import productOrdering from './ordering'
import view from './view'

export default [
  view,
  init({ featuredProducts, categoryProducts }) {
    view.renderList(featuredProducts.slice(0, 4), '#home-product-group-1 .row')
    view.renderList(featuredProducts.slice(4, 8), '#home-product-group-2 .row')
    view.renderList(categoryProducts, '#product-list')
    productOrdering.init()
  }
]

```

Blame Marko Rusic (14 days ago)

Ln 13, Col 1 Spaces: 2 UTF-8 LF JavaScript Prettier: ✓

## 5.6 Cart modul

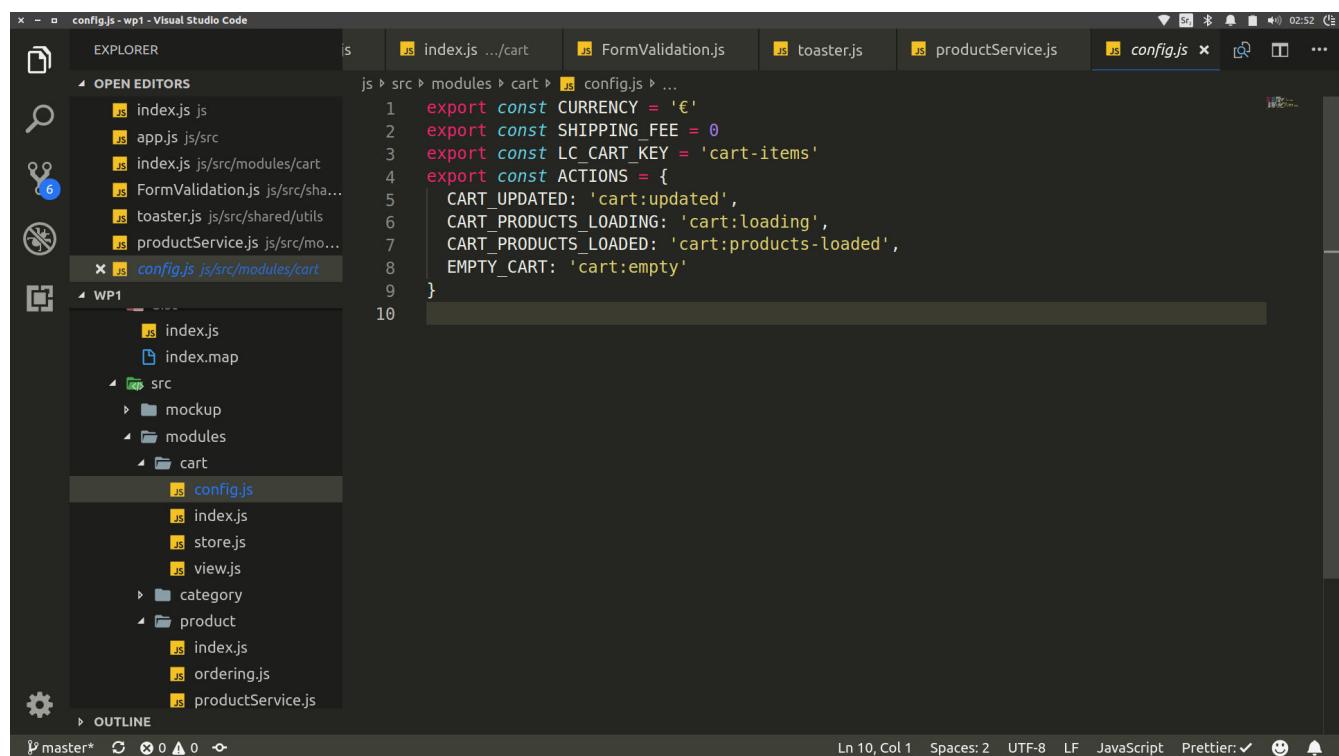
Dinamički prikazuje, i manipuliše sadržajem korpe i čuva ga u LocalStorage-u. Takodje je zadužen za finalni korak u kupovini, tj Checkout formu.

```
→ wp1 git:(master) ✘ tree js/src/modules/cart
js/src/modules/cart
├── config.js
└── index.js
├── store.js
└── view.js

0 directories, 4 files
→ wp1 git:(master) ✘
```

### 5.6.1 config.js

Konfiguracioni fajl sa parametrizovanim vrednostima konstanti.



The screenshot shows the Visual Studio Code interface with the following details:

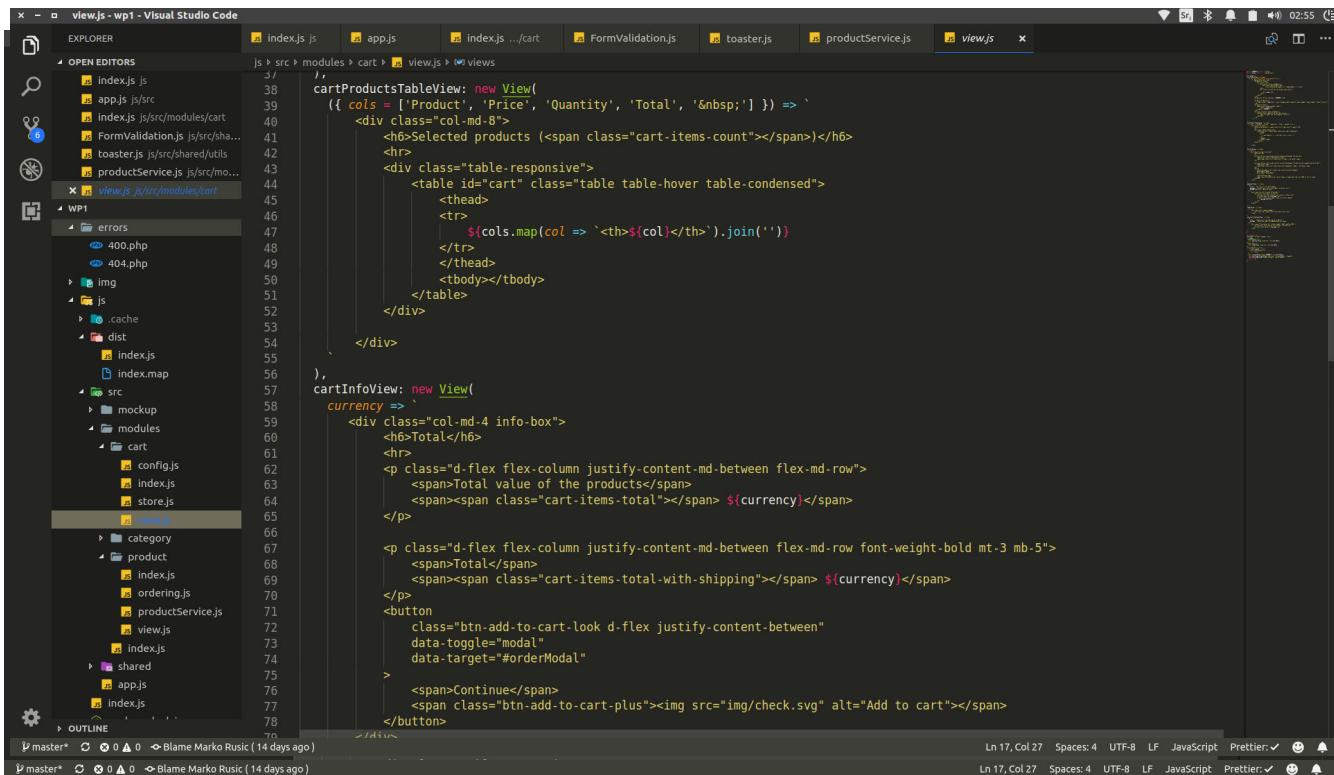
- File Explorer:** Shows the project structure:
  - WP1
  - src
  - modules
  - cart
  - config.js
  - index.js
  - store.js
  - view.js
  - category
  - product
  - index.js
  - ordering.js
  - productService.js
- Editor:** The file `config.js` is open in the editor tab. The code content is as follows:

```
1 export const CURRENCY = '€'
2 export const SHIPPING_FEE = 0
3 export const LC_CART_KEY = 'cart-items'
4 export const ACTIONS = {
5   CART_UPDATED: 'cart:updated',
6   CART_PRODUCTS_LOADING: 'cart:loading',
7   CART_PRODUCTS_LOADED: 'cart:products-loaded',
8   EMPTY_CART: 'cart:empty'
9 }
```

The status bar at the bottom indicates: master\*, 0:0 ▲0, Ln 10, Col 1, Spaces: 2, UTF-8, LF, JavaScript, Prettier: ✓.

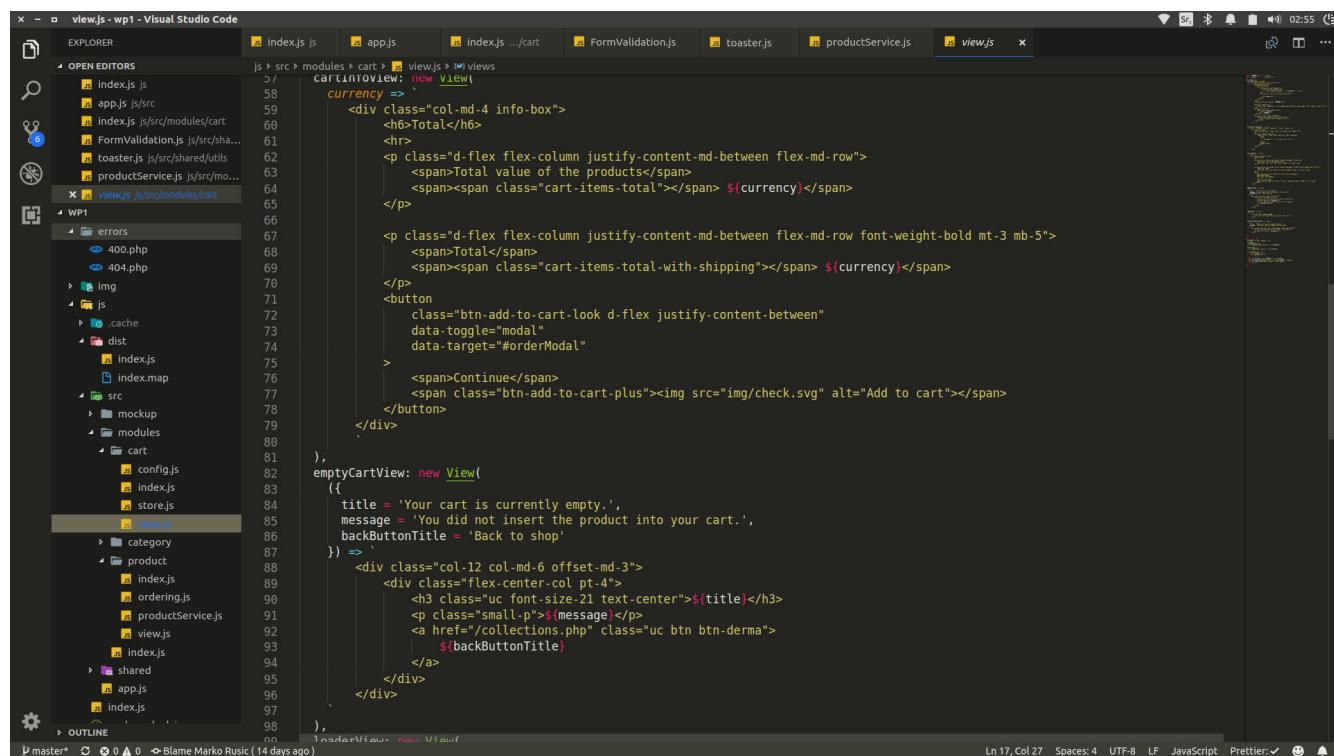
## 5.6.2 view.js

Sadrži sve delove korpe koji se dinamički prikazuju i logiku njihovog prikazivanja.



The screenshot shows the Visual Studio Code interface with the file 'view.js' open. The code is a JavaScript file containing logic for rendering a shopping cart. It includes functions for rendering the total value of items in the cart and handling the addition of items to the cart. The code uses template literals and ES6 syntax. The VS Code status bar at the bottom indicates the file is 17 lines long, 27 columns wide, and uses spaces for indentation.

```
js > src > modules > cart > view.js > views
    ...
    cartProductsTableView: new View(
      ({ cols = ['Product', 'Price', 'Quantity', 'Total', ' '] }) => `
        <div class="col-md-8">
          <h6>Selected products <span class="cart-items-count"></span></h6>
          <hr>
          <div class="table-responsive">
            <table id="cart" class="table table-hover table-condensed">
              <thead>
                <tr>
                  ${cols.map(col => `<th>${col}</th>`).join('')}
                </tr>
              </thead>
              <tbody></tbody>
            </table>
          </div>
        `,
        cartInfoView: new View(
          currency => `
            <div class="col-md-4 info-box">
              <h6>Total</h6>
              <br>
              <p class="d-flex flex-column justify-content-md-between flex-md-row">
                <span>Total value of the products</span>
                <span><span class="cart-items-total">${currency}</span> ${currency}</span>
              </p>
              <p class="d-flex flex-column justify-content-md-between flex-md-row font-weight-bold mt-3 mb-5">
                <span>Total</span>
                <span><span class="cart-items-total-with-shipping">${currency}</span> ${currency}</span>
              </p>
              <button
                class="btn-add-to-cart-look d-flex justify-content-between"
                data-toggle="modal"
                data-target="#orderModal"
              >
                <span>Continue</span>
                <span class="btn-add-to-cart-plus"></span>
              </button>
            </div>
          `,
        ),
        emptyCartView: new View(
          () => `
            <div class="col-12 col-md-6 offset-md-3">
              <div class="flex-center-col pt-4">
                <h3 class="uc font-size-21 text-center">${title}</h3>
                <p class="small-p">${message}</p>
                <a href="/collections.php" class="uc btn btn-derma">
                  ${backButtonTitle}
                </a>
              </div>
            </div>
          `,
        ),
      )
    ),
  
```



The screenshot shows the Visual Studio Code interface with the file 'view.js' open. This part of the code defines a 'emptyCartView' function that displays a message about an empty cart and a link to go back to the shop. The code uses template literals and ES6 syntax. The VS Code status bar at the bottom indicates the file is 17 lines long, 27 columns wide, and uses spaces for indentation.

```
js > src > modules > cart > view.js > views
    ...
    emptyCartView: new View(
      () => `
        <div class="col-12 col-md-6 offset-md-3">
          <div class="flex-center-col pt-4">
            <h3 class="uc font-size-21 text-center">${title}</h3>
            <p class="small-p">${message}</p>
            <a href="/collections.php" class="uc btn btn-derma">
              ${backButtonTitle}
            </a>
          </div>
        </div>
      ),
    ),
  
```

```
1 // ...
2 // ...
3 // ...
4 // ...
5 // ...
6 // ...
7 // ...
8 // ...
9 // ...
10 // ...
11 // ...
12 // ...
13 // ...
14 // ...
15 // ...
16 // ...
17 // ...
18 // ...
19 // ...
20 // ...
21 // ...
22 // ...
23 // ...
24 // ...
25 // ...
26 // ...
27 // ...
28 // ...
29 // ...
30 // ...
31 // ...
32 // ...
33 // ...
34 // ...
35 // ...
36 // ...
37 // ...
38 // ...
39 // ...
40 // ...
41 // ...
42 // ...
43 // ...
44 // ...
45 // ...
46 // ...
47 // ...
48 // ...
49 // ...
50 // ...
51 // ...
52 // ...
53 // ...
54 // ...
55 // ...
56 // ...
57 // ...
58 // ...
59 // ...
60 // ...
61 // ...
62 // ...
63 // ...
64 // ...
65 // ...
66 // ...
67 // ...
68 // ...
69 // ...
70 // ...
71 // ...
72 // ...
73 // ...
74 // ...
75 // ...
76 // ...
77 // ...
78 // ...
79 // ...
80 // ...
81 // ...
82 // ...
83 // ...
84 // ...
85 // ...
86 // ...
87 // ...
88 // ...
89 // ...
90 // ...
91 // ...
92 // ...
93 // ...
94 // ...
95 // ...
96 // ...
97 // ...
98 // ...
99 // ...
100 // ...
101 // ...
102 // ...
103 // ...
104 // ...
105 // ...
106 // ...
107 // ...
108 // ...
109 // ...
110 // ...
111 // ...
112 // ...
113 // ...
114 // ...
115 // ...
116 // ...
117 // ...
118 // ...
119 // ...
120 // ...
121 // ...
122 // ...
123 // ...
124 // ...
125 // ...
126 // ...
127 // ...
128 // ...
129 // ...
130 // ...
131 // ...
132 // ...
133 // ...
134 // ...
135 // ...
136 // ...
137 // ...
138 // ...
139 // ...
```

### 5.6.3 store.js

Čuva i manipuliše podacima iz korpe. Implementira shared EventEmitter klasu u svrhu koršćenja EvenBus paterna.

The screenshot shows the Visual Studio Code interface with the file `store.js` open in the editor. The code implements a shopping cart logic using an event bus and a local storage service. It includes methods for initializing the cart from local storage, fetching products, and updating the cart with new items or quantities.

```
import productService from '../product/productService';
import EventEmiter from '../shared/utils/EventEmitter';
import { ACTIONS, LC_CART_KEY, SHIPPING_FEE } from './config';

export default {
  eventBus: new EventEmitter(),
  cart: [],
  initCart() {
    let cartItems = [];
    try {
      cartItems = JSON.parse(localStorage.getItem(LC_CART_KEY)) || [];
    } catch (e) {}

    if (cartItems.length === 0) {
      this.eventBus.emit(ACTIONS.CART_PRODUCTS_LOADED, cartItems);
      this.setItems(cartItems);
      return;
    }

    const cartItemsIds = cartItems.map(item => item.id);

    this.eventBus.emit(ACTIONS.CART_PRODUCTS_LOADING);

    productService.fetchAll().then(products => {
      const productIds = products.map(({ id }) => id);
      cartItems = cartItems.filter(item => productIds.includes(item.id))
        .map((item, index, cart) => {
          const product = products.find(p => p.id === item.id);
          return {
            ...item,
            ...product
          };
        });
      this.eventBus.emit(ACTIONS.CART_PRODUCTS_LOADED, cartItems);
      this.setItems(cartItems);
    }, 1500);
  },
  setItems(cart) {
    this.cart = [...cart];
    localStorage.setItem(LC_CART_KEY, JSON.stringify(this.cart));
    this.eventBus.emit(ACTIONS.CART_UPDATED, this.cart);
  },
  getMiniCart() {
    return this.cart.map(({ id, quantity }) => ({ id, quantity }));
  },
  countItems() {
    return this.cart.reduce(
      (currentValue, currentItem) => currentValue + currentItem.quantity,
      0
    );
  },
  total() {
    return (
      this.cart.reduce(
        (currentValue, currentItem) =>
          currentValue + currentItem.price * currentItem.quantity,
        0
      ) + SHIPPING_FEE
    );
  },
  totalByProduct(id) {
    const item = this.cart.find(item => item.id === id);
    if (item) {
      return item.price * item.quantity;
    }
    return 0;
  },
  add({ id, quantity }, sumQuantities = true) {
    const cartItem = this.cart.find(product => product.id === id);
    if (!cartItem) {
      this.setItems([...this.cart, { id, quantity }]);
    } else {
      if (sumQuantities) {
        cartItem.quantity += quantity;
      } else {
        cartItem.quantity = quantity;
      }
      this.setItems(this.cart.map(item => (item.id === id ? cartItem : item)));
    }
  }
};
```

This screenshot is identical to the first one, showing the `store.js` file in the WPI project within Visual Studio Code. The code is the same, implementing a shopping cart logic using an event bus and a local storage service.

```
import productService from '../product/productService';
import EventEmiter from '../shared/utils/EventEmitter';
import { ACTIONS, LC_CART_KEY, SHIPPING_FEE } from './config';

export default {
  eventBus: new EventEmitter(),
  cart: [],
  initCart() {
    let cartItems = [];
    try {
      cartItems = JSON.parse(localStorage.getItem(LC_CART_KEY)) || [];
    } catch (e) {}

    if (cartItems.length === 0) {
      this.eventBus.emit(ACTIONS.CART_PRODUCTS_LOADED, cartItems);
      this.setItems(cartItems);
      return;
    }

    const cartItemsIds = cartItems.map(item => item.id);

    this.eventBus.emit(ACTIONS.CART_PRODUCTS_LOADING);

    productService.fetchAll().then(products => {
      const productIds = products.map(({ id }) => id);
      cartItems = cartItems.filter(item => productIds.includes(item.id))
        .map((item, index, cart) => {
          const product = products.find(p => p.id === item.id);
          return {
            ...item,
            ...product
          };
        });
      this.eventBus.emit(ACTIONS.CART_PRODUCTS_LOADED, cartItems);
      this.setItems(cartItems);
    }, 1500);
  },
  setItems(cart) {
    this.cart = [...cart];
    localStorage.setItem(LC_CART_KEY, JSON.stringify(this.cart));
    this.eventBus.emit(ACTIONS.CART_UPDATED, this.cart);
  },
  getMiniCart() {
    return this.cart.map(({ id, quantity }) => ({ id, quantity }));
  },
  countItems() {
    return this.cart.reduce(
      (currentValue, currentItem) => currentValue + currentItem.quantity,
      0
    );
  },
  total() {
    return (
      this.cart.reduce(
        (currentValue, currentItem) =>
          currentValue + currentItem.price * currentItem.quantity,
        0
      ) + SHIPPING_FEE
    );
  },
  totalByProduct(id) {
    const item = this.cart.find(item => item.id === id);
    if (item) {
      return item.price * item.quantity;
    }
    return 0;
  },
  add({ id, quantity }, sumQuantities = true) {
    const cartItem = this.cart.find(product => product.id === id);
    if (!cartItem) {
      this.setItems([...this.cart, { id, quantity }]);
    } else {
      if (sumQuantities) {
        cartItem.quantity += quantity;
      } else {
        cartItem.quantity = quantity;
      }
      this.setItems(this.cart.map(item => (item.id === id ? cartItem : item)));
    }
  }
};
```

The screenshot shows the Visual Studio Code interface with the following details:

- Editor Tab Bar:** index.js, app.js, index.js ../cart, FormValidation.js, toaster.js, productService.js, store.js (highlighted).
- Editor Content:** The code for `store.js` is displayed, showing methods like `add`, `remove`, and `clear` which interact with a `cart` object.
- Explorer Panel:** Shows the project structure under `WP1`. Key folders include `errors`, `img`, `js`, `dist`, and `src`. Under `src`, there are `modules` (containing `cart`, `category`, `product`, `shared`), `index.js`, `index.map`, and `view.js`.
- Bottom Status Bar:** master\*, 0 ▲ 0, Blame Marko Rusic (14 days ago), Ln 4, Col 1, Spaces: 2, UTF-8, LF, JavaScript, Prettier: ✓.

#### 5.6.4 index.js

Entry fajl modula. Objedinjuje funkcionalnosti svih prethodnih delova ovog modula i kontroliše celokupnu funkcionalnost korpe, njenog prikazivanja, i njenih akcija.

Index.js - wp1 - Visual Studio Code

```
OPEN EDITORS
index.js.js
app.js
index.js.../cart
FormValidation.js
toaster.js
productService.js
store.js

WPI
errors
400.php
404.php
img
js
.cache
dist
index.js
index.map
src
mockup
modules
cart
config.js
index.js
store.js
view.js
category
product
index.js
ordering.js
productService.js
view.js
index.js
shared
app.js
index.js

OUTLINE

Ln 25, Col 24  Spaces:2  UTF-8  LF  JavaScript  Prettier:✓  ⚡  🔔
```

index.js

```
1 import { FormValidation } from '../shared/utils'
2 import productService from './product/productService'
3 import store from './store'
4 import view from './view'
5 import { ACTIONS, SHIPPING_FEE, LC_CART_KEY } from './config'
6
7 const { CART_PRODUCTS_LOADING, CART_PRODUCTS_LOADED, CART_UPDATED } = ACTIONS
8
9 export default () => {
10   const isCartPage = window.location.pathname === '/cart.php'
11
12   let checkoutFormValidator = null
13   let checkoutInProgress = false
14   const $dom = {}
15
16   function initCart() {
17     store.eventBus.on(CART_PRODUCTS_LOADING, () => {
18       if (isCartPage) {
19         view.renderLoader()
20       }
21     })
22     store.eventBus.on(CART_PRODUCTS_LOADED, cart => {
23       if (isCartPage) {
24         view.initialRender(cart)
25         cacheCartDom()
26         checkoutFormValidator = new FormValidation({ $form: $dom.checkoutForm })
27         bindCartEvents()
28       }
29     })
30     store.eventBus.on(CART_UPDATED, cart => {
31       $dom.cartItemsCount.text(store.countItems())
32       if (isCartPage) {
33         if (cart.length === 0) {
34           view.renderEmptyCart()
35         }
36         const total = parseFloat(store.total()).toFixed(2)
37         const totalWithFee = parseFloat(total + SHIPPING_FEE).toFixed(2)
38         $dom.cartTotal.text(total)
39         $dom.cartTotalWithShipping.text(totalWithFee)
40       }
41     })
42     store.initCart()
43   }
44
45   function cacheCartDom() {
46     $dom.addToCart = $('.btn-add-to-cart')
47     $dom.addToCartQuantity = $('#add-to-cart-quantity')
48     $dom.cartItemsCount = $('.cart-items-count')
49   }
50
51   function cacheCartDom() {
52     $dom.cartTotal = $('.cart-items-total')
53     $dom.cartTotalWithShipping = $('.cart-items-total-with-shipping')
54     $dom.removeFromCart = $('.remove-from-cart')
55     $dom.changeQuantity = $('.change-product-quantity')
56     $dom.cartItemsCount = $('.cart-items-count')
57     $dom.checkoutForm = $('#checkout-form')
58     $dom.orderModal = $('#orderModal')
59   }
60
61   function bindEvents() {
62     $dom.addToCart.on('click', _handleAddToCart)
63   }
64
65   function bindCartEvents() {
66     $dom.removeFromCart.on('click', handleRemoveFromCart)
67     $dom.changeQuantity.on('change keydown', handleQuantityChange)
68     $dom.checkoutForm.on('submit', handleCheckout)
69   }
70
71   function handleAddToCart(event) {
72     event.preventDefault()
73     const $btn = $(event.target).closest('[data-product-id]')
74     const id = $btn.data().productId
75     let quantity = 1
76
77     if (!id) {
78       return
79     }
80
81     if ($dom.addToCartQuantity.length > 0) {
```

Index.js - wp1 - Visual Studio Code

```
OPEN EDITORS
index.js.js
app.js
index.js.../cart
FormValidation.js
toaster.js
productService.js
store.js

WPI
errors
400.php
404.php
img
js
.cache
dist
index.js
index.map
src
mockup
modules
cart
config.js
index.js
store.js
view.js
category
product
index.js
ordering.js
productService.js
view.js
index.js
shared
app.js
index.js

OUTLINE

Ln 25, Col 24  Spaces:2  UTF-8  LF  JavaScript  Prettier:✓  ⚡  🔔
```

index.js

```
40   }
41   store.initCart()
42
43   function cacheCartDom() {
44     $dom.addToCart = $('.btn-add-to-cart')
45     $dom.addToCartQuantity = $('#add-to-cart-quantity')
46     $dom.cartItemsCount = $('.cart-items-count')
47   }
48
49   function cacheCartDom() {
50     $dom.cartTotal = $('.cart-items-total')
51     $dom.cartTotalWithShipping = $('.cart-items-total-with-shipping')
52     $dom.removeFromCart = $('.remove-from-cart')
53     $dom.changeQuantity = $('.change-product-quantity')
54     $dom.cartItemsCount = $('.cart-items-count')
55     $dom.checkoutForm = $('#checkout-form')
56     $dom.orderModal = $('#orderModal')
57   }
58
59   function bindEvents() {
60     $dom.addToCart.on('click', _handleAddToCart)
61   }
62
63   function bindCartEvents() {
64     $dom.removeFromCart.on('click', handleRemoveFromCart)
65     $dom.changeQuantity.on('change keydown', handleQuantityChange)
66     $dom.checkoutForm.on('submit', handleCheckout)
67   }
68
69   function handleAddToCart(event) {
70     event.preventDefault()
71     const $btn = $(event.target).closest('[data-product-id]')
72     const id = $btn.data().productId
73     let quantity = 1
74
75     if (!id) {
76       return
77     }
78
79     if ($dom.addToCartQuantity.length > 0) {
```

Index.js - wp1 - Visual Studio Code

```
js > src > modules > cart > index.js > <function> > _initCart
    81 if ($dom.addToCartQuantity.length > 0) {
    82     quantity = parseInt($dom.addToCartQuantity.val())
    83 }
    84
    85 store.add({ id, quantity })
    86 $toaster('Successfully added to <a href="/cart.php">cart</a>!')
    87 }
    88
    89 function _handleRemoveFromCart(event) {
    90     event.preventDefault()
    91     const $el = $(event.target)
    92     const $product = $el.closest('[data-product-id]')
    93     store.remove($product.data().productId)
    94     $product.fadeOut()
    95 }
    96
    97 function _handleQuantityChange(event) {
    98     event.preventDefault()
    99     const $el = $(event.target)
   100    const $product = $el.closest('[data-product-id]')
   101    const product = $product.data()
   102    const quantity = parseInt($el.val())
   103    if (isNaN(quantity)) {
   104        return
   105    }
   106    store.add(
   107        {
   108            id: product.productId,
   109            quantity
   110        },
   111        false
   112    )
   113    $product
   114        .find('.single-product-total')
   115        .text(store.totalByProduct(product.productId))
   116 }
   117
   118 function _handleCheckout(event) {
   119     event.preventDefault()
   120     if (checkoutInProgress) {
   121         return
   122     }
   123     if (!checkoutFormValidator.validate()) {
   124         return
   125     }
   126     checkoutInProgress = true
   127     const $form = $(event.target)
   128     const $btn = $form.find('button[type="submit"]')
   129     const $modalBody = $form.parent()
   130     const btnTextBefore = $btn.text()
   131     $btn.css({ 'pointer-events': 'none' }).text('Please wait...')
   132     productService
   133         .checkout()
   134             .then(() => {
   135                 localStorage.removeItem(LC_CART_KEY)
   136                 $form.fadeOut() => {
   137                     $modalBody.addClass('flex-center-col')
   138                     view.successfulCheckoutView.render({}, $modalBody)
   139                 }
   140             })
   141             $dom.orderModal.on('hide.bs.modal', () => store.clear())
   142             checkoutInProgress = false
   143         })
   144             .catch(() => {
   145                 checkoutInProgress = false
   146                 $btn.css({ 'pointer-events': 'auto' }).text(btnTextBefore)
   147                 alert('An error has occurred!')
   148             })
   149
   150     return {
   151         init() {
   152             _cacheDom()
   153             _initCart()
   154             _bindEvents()
   155         }
   156     }()
   157 }()
```

Ln 25, Col 24 Spaces:2 UTF-8 LF JavaScript Prettier:✓

Index.js - wp1 - Visual Studio Code

```
js > src > modules > cart > index.js > <function> > _initCart
    117 function _handleCheckout(event) {
    118     event.preventDefault()
    119     if (checkoutInProgress) {
    120         return
    121     }
    122     if (!checkoutFormValidator.validate()) {
    123         return
    124     }
    125     checkoutInProgress = true
    126     const $form = $(event.target)
    127     const $btn = $form.find('button[type="submit"]')
    128     const $modalBody = $form.parent()
    129     const btnTextBefore = $btn.text()
    130     $btn.css({ 'pointer-events': 'none' }).text('Please wait...')
    131     productService
    132         .checkout()
    133             .then(() => {
    134                 localStorage.removeItem(LC_CART_KEY)
    135                 $form.fadeOut() => {
    136                     $modalBody.addClass('flex-center-col')
    137                     view.successfulCheckoutView.render({}, $modalBody)
    138                 }
    139             })
    140             $dom.orderModal.on('hide.bs.modal', () => store.clear())
    141             checkoutInProgress = false
    142         })
    143             .catch(() => {
    144                 checkoutInProgress = false
    145                 $btn.css({ 'pointer-events': 'auto' }).text(btnTextBefore)
    146                 alert('An error has occurred!')
    147             })
    148
    149     return {
    150         init() {
    151             _cacheDom()
    152             _initCart()
    153             _bindEvents()
    154         }
    155     }()
    156 }()
```

Ln 25, Col 24 Spaces:2 UTF-8 LF JavaScript Prettier:✓