9

Building an E-Commerce Website with Bootstrap

You took a look at the various attributes and features of Bootstrap in the earlier chapters. However, just learning the concepts is not enough, as you need to possess the skills to implement them practically. In this chapter, we will create an e-commerce website that will help you get to grips with web designing using Bootstrap.

Initially, we will build the parts step by step and increase the level of difficulty gradually so that you do not get overwhelmed with an information overload. We will first build the `ecommerce.html` page and then create the other pages such as `category.html`, `account.html`, and `product.html`, resulting in a responsive website.

> With Notepad, you cannot determine whether each `<div>` element has been closed, resulting in an incorrect output. In this example, we will be using Notepad++ as it provides syntax highlighting, word completion, and syntax folding, which will help you code better. Notepad++ is open source and can be downloaded for free from its official website at `http://notepad-plus-plus.org/`. It is a good practice to use Notepad++ or any advanced editor for your projects, as it streamlines your web designing experience.

Designing the ecommerce.html page

Create four web pages, namely `ecommerce.html`, `category.html`, `account.html`, and `product.html` for the e-commerce, product categories, account, and the product pages respectively.

Initially, we will look at `ecommerce.html`.

For now, let's paste the following basic Bootstrap code in the `ecommerce.html` file:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="utf-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <meta name="description" content="">
  <meta name="author" content="">

  <title>Bootstrap Store</title>

  <!-- Bootstrap Core CSS -->
  <link href="css/bootstrap.css" rel="stylesheet">

  <!-- Custom CSS -->
  <link href="style.css" rel="stylesheet">

  <!-- HTML5 Shim and Respond.js IE8 support of HTML5 elements and
media queries -->
  <!-- WARNING: Respond.js doesn't work if you view the page via
file:// -->
  <!--[if lt IE 9]>
    <script src="https://oss.maxcdn.com/libs/html5shiv/3.7.0/
html5shiv.js"></script>
    <script src="https://oss.maxcdn.com/libs/respond.js/1.4.2/respond.
min.js"></script>
  <![endif]-->
</head>

<body>
  <!-- jQuery Version 1.11.0 -->
  <script src="js/jquery-1.11.1.js"></script>

  <!-- Bootstrap Core JavaScript -->
  <script src="js/bootstrap.js"></script>
</body>
</html>
```

As you can see, we have also added `respond.min.js`, the jQuery file, and `HTML shiv` along with the Bootstrap files.

> Some CSS3 properties and HTML5 elements are not fully supported by the Internet Explorer 8 browser. Therefore, Internet Explorer 8 requires the use of `html5shiv` to correctly display those elements and `Respond.js` to enable media query support.

We will now create a navbar for the web page.

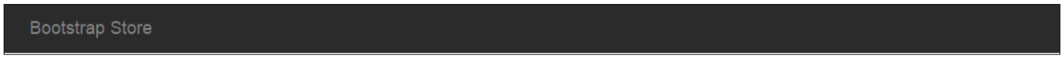
Similar to the process of creating a navbar in the preceding chapters, you define the navbar between the `<body>` tags. We define `navbar-brand` as `Bootstrap Store` and use the `.navbar-inverse` class, resulting in a black background and white text for the navbar:

```
<body>
<!-- Navigation -->
  <nav class="navbar-inverse" role="navigation">
    <div class="container-fluid">
      <!-- Brand and toggle get grouped for better mobile display -->
      <div class="navbar-header">
        <button type="button" class="navbar-toggle collapsed" data-
toggle="collapse" data-target="#bs-example-navbar-collapse-1">
          <span class="sr-only">Toggle navigation</span>
          <span class="icon-bar"></span>
          <span class="icon-bar"></span>
          <span class="icon-bar"></span>
        </button>
        <a class="navbar-brand" href="ecommerce.html">Bootstrap
Store</a>
      </div>

      <!-- Collect the nav links, forms, and other content for
toggling -->
      <div class="collapse navbar-collapse" id="bs-example-navbar-
collapse-1">
      </div><!-- /.navbar-collapse -->
    </div><!-- /.container-fluid -->
  </nav>
  <!-- jQuery Version 1.11.0 -->
  <script src="js/jquery-1.11.1.js"></script>

  <!-- Bootstrap Core JavaScript -->
  <script src="js/bootstrap.js"></script>
</body>
```

The output of the code on execution will be as follows:



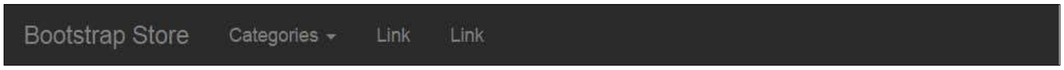
Further on, we will add the categories and navigation links to this `navbar`. The code has to be inserted after the `<!-- Collect the nav links, forms, and other content for toggling -->` comment and before the `<div>` element containing the `<!-- /.navbar-collapse -->` comment.

In the code, we define the **Categories** link in addition to the other navigation links. We create a dropdown for the **Categories** link, wherein we define the various types of products by their genre such as baby products, electronics, and shoes.

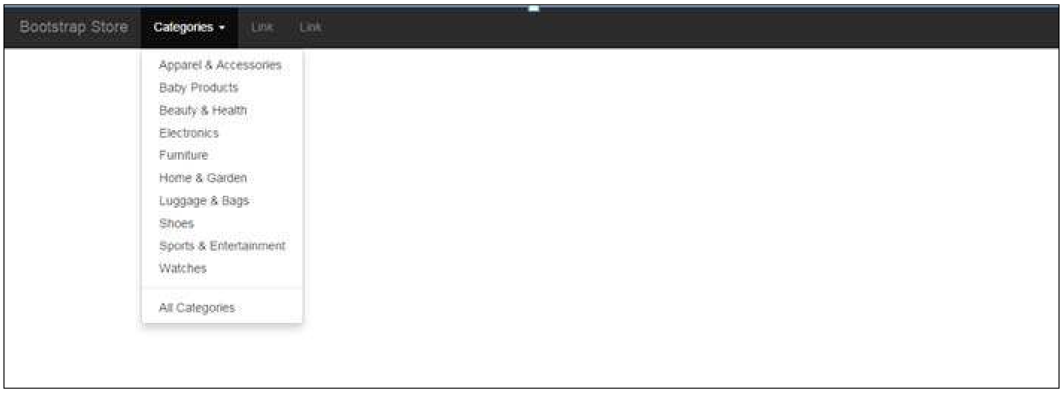
Take a look at the following code to understand it better:

```
<div class="collapse navbar-collapse" id="bs-example-navbar-
collapse-1">
<ul class="nav navbar-nav">
          <li class="dropdown">
            <a href="#"  class="active dropdown-toggle" data-
toggle="dropdown">Categories <span class="caret"></span></a>
            <ul class="dropdown-menu" role="menu">
              <li><a href="category.html">Apparel  &amp; Accessories</
a></li>
              <li><a href="category.html">Baby Products</a></li>
              <li><a href="category.html">Beauty  &amp; Health</a></
li>
              <li><a href="category.html">Electronics</a></li>
              <li><a href="category.html">Furniture</a></li>
              <li><a href="category.html">Home  &amp; Garden</a></li>
              <li><a href="category.html">Luggage  &amp; Bags</a></li>
              <li><a href="category.html">Shoes</a></li>
              <li><a href="category.html">Sports  &amp;
Entertainment</a></li>
              <li><a href="category.html">Watches</a></li>
              <li class="divider"></li>
              <li><a href="ecommerce.html">All Categories</a></li>
            </ul>
          </li>
          <li><a href="#">Link</a></li>
          <li><a href="#">Link</a></li>
        </ul>
```

The output of the code will be as follows:



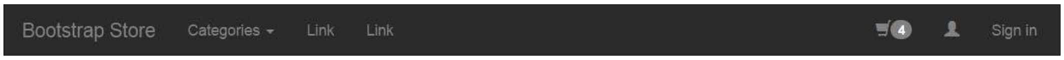
Thus, we have defined the **Categories** menu and the drop-down menu, which is visible on clicking the caret.



Now, we add the **Sign in** link, user account link, and the shopping cart with a badge to the right-hand side of the navbar and specify Glyphicons for them. Remember that this snippet needs to be added after the links defined prior to it and before the `<div>` element containing the `<!-- /.navbar-collapse -->` comment:

```
<ul class="nav navbar-nav navbar-right">
          <li><a href="#"><span class="badge pull-right">4</span><i
class="glyphicon glyphicon-shopping-cart"></i></a></li>
          <li><a href="account.html"><i class="glyphicon glyphicon-
user"></i></a></li>
          <li><a href="#" data-toggle="modal" data-
target="#myModal">Sign in</a></li>
        </ul>
```

The output of the added code on execution will result in the following screenshot:



For the **Sign in** part, we will create a modal that will be displayed by clicking on the **Sign in** link.