



Radosław Gryczan

Bootstrap

TWORZENIE WŁASNYCH STYLÓW GRAFICZNYCH

Helion 

Wszelkie prawa zastrzeżone. Nieautoryzowane rozpowszechnianie całości lub fragmentu niniejszej publikacji w jakiegokolwiek postaci jest zabronione. Wykonywanie kopii metodą kserograficzną, fotograficzną, a także kopiowanie książki na nośniku filmowym, magnetycznym lub innym powoduje naruszenie praw autorskich niniejszej publikacji.

Wszystkie znaki występujące w tekście są zastrzeżonymi znakami firmowymi bądź towarowymi ich właścicieli.

Autor oraz Wydawnictwo HELION dołożyli wszelkich starań, by zawarte w tej książce informacje były kompletne i rzetelne. Nie biorą jednak żadnej odpowiedzialności ani za ich wykorzystanie, ani za związane z tym ewentualne naruszenie praw patentowych lub autorskich. Autor oraz Wydawnictwo HELION nie ponoszą również żadnej odpowiedzialności za ewentualne szkody wynikłe z wykorzystania informacji zawartych w książce.

Redaktor prowadzący: Małgorzata Kulik

Projekt okładki: Studio Gravite / Olsztyn
Obarek, Pokoński, Pazdrijowski, Zaprucki

Wydawnictwo HELION
ul. Kościuszki 1c, 44-100 GLIWICE
tel. 32 231 22 19, 32 230 98 63
e-mail: helion@helion.pl
WWW: <http://helion.pl> (księgarnia internetowa, katalog książek)

Drogi Czytelniku!

Jeżeli chcesz ocenić tę książkę, zajrzyj pod adres

<http://helion.pl/user/opinie/bootgr>

Możesz tam wpisać swoje uwagi, spostrzeżenia, recenzję.

Kody źródłowe wybranych przykładów dostępne są pod adresem:

<ftp://ftp.helion.pl/przyklady/bootgr.zip>

ISBN: 978-83-246-9255-2

Copyright © Helion 2017

Printed in Poland.

- [Kup książkę](#)
- [Poleć książkę](#)
- [Oceń książkę](#)

- [Księgarnia internetowa](#)
- [Lubię to! » Nasza społeczność](#)

Spis treści

| | |
|-------------------------------------------------------------------------------------------------|-----------|
| Słowo wstępne | 9 |
| Rozdział 1. Najważniejsze informacje — tytułem przypomnienia | 13 |
| Bootstrap: siatka, czyli grid | 14 |
| Narzędzia deweloperskie | 15 |
| Semantyka kodu | 16 |
| Outline kodu HTML | 17 |
| Block Element Modifier | 18 |
| Progressive Enhancement | 20 |
| Dostępność stron WWW | 21 |
| Podsumowanie rozdziału | 23 |
| Rozdział 2. Bootstrap i jego zastosowanie w praktyce — przykłady stron WWW | 25 |
| Jak rozpoznać, że witrynę napisano w Bootstrapie? | 26 |
| Datanyze Insider | 26 |
| builtwith.com | 27 |
| Badanie źródła strony | 27 |
| Strony napisane za pomocą Bootstrapa | 28 |
| Musicam Sacram | 28 |
| rozwal.to | 30 |
| nba.com | 31 |
| gfycat.com | 32 |
| fontawesome.io | 33 |
| Portale z projektami utworzonymi w Bootstrapie | 33 |
| Bootstrap Expo | 34 |
| Built with Bootstrap | 34 |
| Rozdział 3. Dodatkowe, bardzo przydatne zasoby związane z Bootstrapem | 37 |
| Edytor kodu | 37 |
| Emmet | 40 |
| Snippety | 42 |

| | |
|--------------------------------------------------------------------------------|------------|
| Minimapa kodu | 44 |
| Polskie znaki w Atomie | 45 |
| Dodatkowe materiały dotyczące pisania stron | 47 |
| Bootswatch | 47 |
| Kolorystyka | 50 |
| Stałe fragmenty kodu | 50 |
| Dokumentacja Bootstrapa | 51 |
| Kody HTML oraz CSS — przydatne adresy | 55 |
| Zdjęcia stockowe oraz placeholderzy | 57 |
| Podsumowanie | 59 |
| ZADANIA | 61 |
| Rozdział 4. Zaczynamy własny projekt w Bootstrapie — plan witryny | 63 |
| Obsługa programu graficznego | 67 |
| Praca nad szablonem | 71 |
| Nawigacja oraz strona wejściowa | 71 |
| Część strony „Jacy jesteśmy?” | 75 |
| Część strony „Galeria” | 79 |
| Część strony „Zespół” | 83 |
| Część strony „Osiągnięcia i nagrody” | 87 |
| Część strony „Poznaj opinie” | 92 |
| Ostatnia część strony — stopka | 96 |
| Podsumowanie pracy nad szablonem | 97 |
| Praca ogólna nad szablonem | 98 |
| ZADANIA | 101 |
| Rozdział 5. Wydajność strony a narzędzia Bootstrapa | 103 |
| Optymalizacja grafiki | 104 |
| Optymalizacja obrazów za pomocą programu GIMP | 104 |
| Optymalizacja obrazów za pomocą narzędzi online | 105 |
| Kompatybilność z różnymi przeglądarkami | 106 |
| Badanie optymalizacji strony za pomocą narzędzi deweloperskich | 108 |
| Szybkość pobierania elementów | 108 |
| Kody HTML oraz CSS witryny | 109 |
| Konsola JS na stronie | 110 |
| Treści na stronie | 110 |
| Testy optymalizujące stronę | 111 |
| Narzędzia zewnętrzne | 112 |
| YSlow | 112 |
| PageSpeed Insights | 113 |
| Badanie strony w wersji dla urządzeń mobilnych | 115 |
| SEO — RWD a subdomena m. | 116 |

| | |
|------------------------------------------------------------|------------|
| Ładowanie się plików | 117 |
| Modyfikacja gotowych stylów Bootstrapa | 118 |
| Minifikacja plików CSS — generatory | 122 |
| Content Delivery Network (CDN) | 123 |
| Kolejność ładowania się plików | 127 |
| ZADANIA | 128 |
| Rozdział 6. Pisanie strony w języku HTML | 129 |
| Szablony HTML | 130 |
| Modyfikacja pliku HTML | 134 |
| Ekran powitalny | 136 |
| Sekcja „Jacy jesteśmy?” | 139 |
| Galeria strony | 145 |
| Prezentacja zespołu (karuzela) | 150 |
| Osiągnięcia i nagrody | 155 |
| „Co mówią” — poznaj opinie | 160 |
| Stopka strony | 164 |
| Rozdział 7. Własne style CSS | |
| — dostosowywanie stylów do potrzeb strony | 169 |
| Style resetujące oraz normalizujące | 170 |
| Stylizowanie strony WWW | 171 |
| Czcionki | 173 |
| Stylizowanie klas ogólnych | 173 |
| Stylizowanie poszczególnych części strony | 179 |
| Dostosowanie strony do potrzeb urządzeń mobilnych | 209 |
| ZADANIA | 216 |
| Rozdział 8. Podsumowanie, wnioski i garść refleksji | |
| o pisaniu responsywnych stron bez Bootstrapa | 217 |
| Skończyliśmy... i co dalej? | 221 |
| Inne frameworki CSS | 222 |
| Foundation | 222 |
| Semantic UI | 222 |
| Material Design Lite | 223 |
| Skorowidz | 225 |

Część strony „Poznaj opinie”

Przejdźmy zatem do treści dostępnych po wybraniu menu *Co mówią*. Chodzi tu z pewnością o publikację opinii klientów. Jeszcze zanim omówię projekt graficzny strony, zastanówmy się, dlaczego warto umieszczać takie treści na stronie.

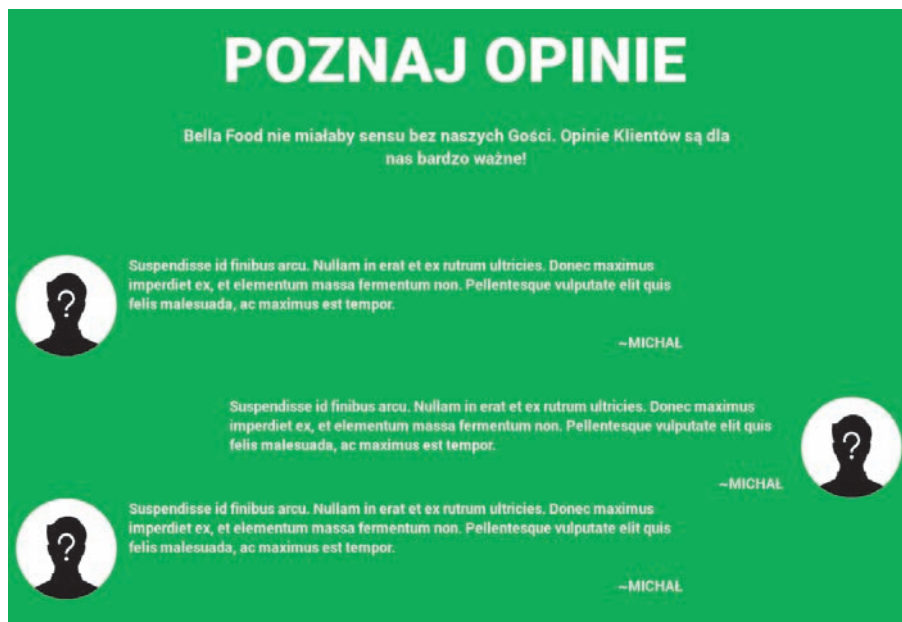
Przede wszystkim w ten sposób restauracja udowadnia, że jest odwiedzana przez gości. Recenzje umieszczone na stronie nie biorą się znikąd, a jeśli są one zawsze pozytywne, najpewniej wpłyną również na opinie innych osób. Oczywiście klienci muszą mieć na uwadze to, że prezentowane komentarze mogą być manipulowane i dlatego są one bardzo wygodnym sposobem promowania produktów lub miejsc. Jednakże na treść opinii twórcy strony nie ma wpływu, ponieważ jest to opinia klienta. Z tego względu na stronie mogą pokazać się wpisy, które nie przysparzają chwały profesjonalnej witrynie WWW.

Sekcja prezentująca opinie gości jest nieco luźniejszą częścią witryny, dlatego bardzo często pomaga uratować klimat strony. Jeśli treści witryny są na bardzo wysokim poziomie, jej powaga może spowodować pewien dyskomfort użytkownika. Między innymi dlatego ta część witryny ma zostać pokazana na końcu — chodzi o to, by w pamięci użytkownika pozostało tylko dobre wrażenie wywołane właśnie przez nią.

Po tych refleksjach możemy wreszcie napisać parę słów na temat restauracji. Ewentualna zmiana treści na prawdziwe opinie będzie zależeć od klienta. W tej chwili naszym celem jest utworzenie takich recenzji, które dla każdego potencjalnego konsumenta będą zachęcające. Najpierw zajrzyjmy do projektu strony, aby sprawdzić, jak wygląda zakładka z prezentacjami komentarzy gości (rysunek 4.29).

Jak widać, strona z opiniami klientów nie została zaprojektowana w jakiś specjalny sposób. Wygląda na to, że grafik znudził się projektowaniem szczegółów. Jest to jednak dobre rozwiązanie, jeśli chodzi o UX, ponieważ znany schemat jest o wiele bardziej przyjazny dla oglądającego niż kolejny nowy element. Zastanówmy się teraz nad treścią opinii.

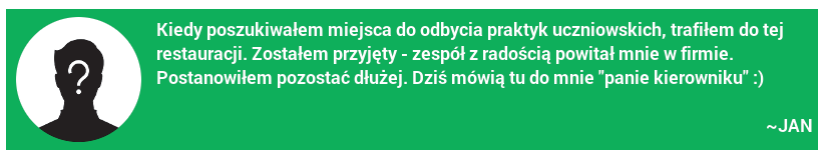
Mamy do uzupełnienia trzy pola, do których należałoby przypisać grafiki (mają one przedstawiać sylwetki komentujących). Dobrym pomysłem będzie przeznaczenie jednego z nich na element lekko humorystyczny. Przez to uzyskamy skrócenie dystansu do użytkownika. Tym zabawnym elementem



Rysunek 4.29. Tekst w nagłówku jest w porządku, możemy go więc pozostawić

nie może być jednak pierwsza opinia — później użytkownik przeczyta dwa poważniejsze komentarze. Nie może to być również druga recenzja. Najodpowiedniejszym miejscem dla takiego wesołego akcentu będzie ostatnia opinia — zacznijmy właśnie od niej.

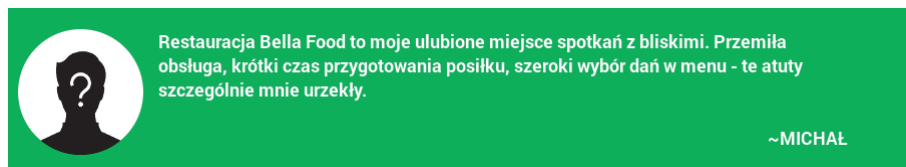
Aby zwiększyć szanse na skrócenie dystansu do użytkownika, warto wzbudzić w nim pozytywne emocje. Dlatego też zastosujemy tekst związany z pracownikami restauracji. Postawmy na osobę, którą już znamy ze slajdów. Warto przy tym pamiętać, do jakiej grupy wiekowej się zwracamy. W tym przypadku będą to raczej osoby młode. Napijemy zatem opinię, którą mógłby wygłosić Jan Nowak — mężczyzna pracujący w lokalu od dobrych kilku lat (rysunek 4.30).



Rysunek 4.30. Trzecia opinia

Taka treść dobrze się prezentuje. Dzięki takiej opinii dowiadujemy się, że w restauracji pracują młodzi ludzie, a to rzeczywiście ma duży wpływ na jej funkcjonowanie. Wróćmy więc do pierwszej opinii. Jak już wcześniej

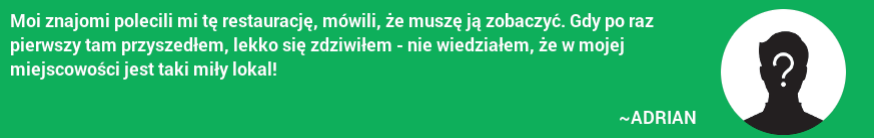
wspomniałem, jeżeli manipulujemy opiniami, warto podkreślać w nich właściwości, które uważamy za cenne. W przypadku branży gastronomicznej może to być szybkość obsługi, miły i fachowy personel czy odpowiednio skomponowane menu. Pierwsza opinia przybierze zatem taką formę jak na rysunku 4.31.



Rysunek 4.31. Taka treść buduje zaufanie do restauracji

Taka recenzja z pewnością wzbogaci zawartość merytoryczną strony i wzbudzi zaufanie klienta.

Drugą opinię skonstruujemy podobnie, jednak większą rolę powierzmy w niej bliskim naszego zadowolonego gościa. Mogli oni np. odwiedzić restaurację wcześniej i mu ją polecić. Dodatkowy przekaz, który chcemy w ten sposób skierować do klienta, brzmi: *najwyraźniej Bella Food jest porządnym lokalem, skoro się ją poleca znajomym*. Sama opinia może być zakończona wyrazem pozytywnego zaskoczenia (rysunek 4.32).



Moi znajomi polecili mi tę restaurację, mówili, że muszę ją zobaczyć. Gdy po raz pierwszy tam przyszedłem, lekko się zdziwiłem - nie wiedziałem, że w mojej miejscowości jest taki miły lokal!

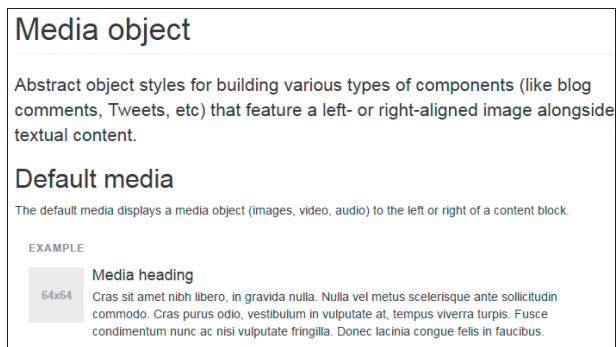
~ADRIAN



Rysunek 4.32. Opinia, którą umieścimy na miejscu drugim

Całość prezentuje się naprawdę dobrze. Napisaaliśmy kilkanaście linijek tekstu, aby podkreślić wiele cennych zalet, które przyczynią się do częstszych wizyt klientów. Teraz warto się zastanowić, jak zapisać tę stronę w kodzie HTML. W praktyce taka opinia jest cytatem z wypowiedzi konkretnej osoby. Właściwym semantycznie znacznikiem do zapisywania cytatów jest błąd `>quote`. Posiada on atrybut `cite`, pozwalający na odnoszenie się do witryny, z której pochodzi dana treść. Nie wykorzystamy go w naszej sytuacji, ponieważ w rzeczywistości źródłem cytatów będzie strona, na której opublikujemy wypowiedzi. Takie odwołania byłyby trochę nielogiczne. Rodzi się jednak kolejne pytanie: czy Bootstrap jest w stanie ułatwić nam pracę nad tą częścią kodu? Odpowiedź brzmi: tak, jest w stanie (rysunek 4.33).

Rysunek 4.33.
*Bootstrap jest
 potężnym
 narzędziem
 o wielkich
 możliwościach*



W praktyce naszym zadaniem będzie wyłącznie przebudowa strony ze schematu blokowego na semantyczny. Nie jest to trudne, ponieważ już wiemy, jakie znaczniki do tego wykorzystamy. Jeżeli jednak dokładniej spojrzymy na strukturę tego elementu, zauważymy część opisaną jako *Media heading*. Projekt strony wykonany przez grafika nie zawiera tego elementu, ale możemy go użyć w celu określenia postaci osoby, która wyraża opinię o restauracji. Dzięki odpowiedniej stylizacji osiągniemy zamierzony efekt.

Prawdopodobnie prezentacja opinii jest ostatnim elementem, który mogliśmy zmodyfikować jako osoby zajmujące się projektem witryny od strony jej wartości merytorycznej. Warto w tym miejscu podsumować nasze dokonania.

Utworzonym treściom nadaliśmy charakter dostosowany do potrzeb. Zarówno projektantowi, jak i klientowi (właścicielowi restauracji) zależy na doborze odpowiednich komentarzy. Jeżeli potencjalny gość restauracji trafi na jej stronę i będzie mile zaskoczony poziomem prezentowanych treści, wzrosną szanse na to, że odwiedzi on lokal również w rzeczywistości. Okazuje się więc, że interesy projektanta, klienta i konsumenta są zbieżne.

Poświęciliśmy bardzo dużo uwagi zagadnieniom odpowiedniego przygotowania treści pod kątem merytorycznym. Przyczyna jest prosta: to właśnie treść buduje stronę, ponieważ jest ze swojej natury nośnikiem informacji. W praktyce większość elementów graficznych pełni głównie funkcje ozdobne. Elementy te poprawiają odbiór strony przez użytkownika, ale nie są czynnikiem warunkującym jej użyteczność. Przydatność strony wynika z informacji, jakie prezentuje użytkownikom. W tym miejscu warto przywołać przykład karuzeli. Najczęściej jest ona rodzajem galerii służącej do wyświetlania zdjęć, niemniej bez trudu możemy w karuzeli zaprezentować zdjęcie wraz z opisem. Taka praktyka jest jednak bardzo zła, ponieważ w razie problemów z załadowaniem plików graficznych nie możemy przekazać użytkownikowi żadnych informacji. Tych kłopotów unikniemy, jeśli teksty umieścimy poza karuzelą: nawet jeżeli zdjęcia nie zostaną wyświetlone,

użytkownik otrzyma informację tekstową. Teksty na ogół są przystępnym dla użytkownika sposobem przekazywania informacji. Można je zredagować tak, aby były zrozumiałe i spójne w treści oraz formie. Jeśli tematyka każdej sekcji witryny jest powiązana z tematyką innych sekcji, łatwiej utrzymać konsekwentny i logiczny przekaz informacyjny całego projektu.

Warto zatem pamiętać o znaczeniu odpowiedniego przygotowania treści strony. Zdarzają się przypadki, że właśnie opracowaniu zawartości stron trzeba poświęcić o wiele więcej czasu niż pisaniu samego kodu źródłowego. Taka sytuacja jest często spotykana w przypadku doświadczonych twórców stron, którzy ukończyli już setki projektów. Potrafią oni bez trudu zadbać o atrakcyjny wygląd strony, a przecież wygląd ma za zadanie tylko wzbudzić u oglądającego zaciekawienie — ważniejszym zadaniem jest dostarczenie informacji, a to już zależy od treści strony.

Ostatnia część strony — stopka

Pozostało nam już tylko omówienie ostatniej części strony — stopki. Stopka zawiera informacje o możliwych sposobach nawiązania kontaktu z restauracją (rysunek 4.34).



Rysunek 4.34. Część strony „Kontakt”

Jak widać, nie ma tutaj niczego szczególnego. Jedynym elementem nieco wyłamującym się z konwencji przyjętej w całym projekcie jest mapa — nie została ona dostosowana do określonej szerokości tak jak inne składowe strony. Prezentuje się jednak ładnie i pozwala na ukazanie większego fragmentu terenu bez zmiany skali. Poniżej mapy mamy adres, numer telefonu oraz adres e-mail. Być może zastanawiasz się, czy to nie jest zbyt mało. Brakuje tutaj elementów związanych z mediami społecznościowymi, takich jak odnośniki do serwisów typu Facebook czy YouTube. Warto jednak pamiętać, że niewielka restauracja nie musi utrzymywać profili na kilkunastu portalach społecznościowych, nawet jeśli strona na Facebooku jest wygodnym miejscem do prezentowania różnych treści. Uznajmy jednak, że restauracja nie korzysta z tego rodzaju możliwości — właśnie dlatego w tej części strony nie uwzględniliśmy żadnych łączy do tego typu portali.

Aby ułatwić sobie pracę nad stopką, do napisania kodu mapy wykorzystamy narzędzia oferowane przez Google Maps — po prostu udostępnimy fragment mapy. W przypadku adresu użyjemy znacznika `address`, który pozwoli na nadanie tej części strony wartości semantycznej. Jest to zresztą jedna z nielicznych sytuacji, w których można zastosować znacznik wymuszający łamanie linii (czyli znacznik `br`).

Łatwo zauważyć, że ta część strony jest ostatnią ze wszystkich również z powodu zawartych w niej treści. W naturalny sposób można potraktować ją jako stopkę, ponieważ dane kontaktowe umieszcza się najczęściej właśnie tam. Uznajmy zatem, że dane kontaktowe nie będą umieszczone w elemencie zawartym w znaczniku `main`, a znajdą się po prostu w znaczniku `footer` całej strony internetowej.

Podsumowanie pracy nad szablonem

W ten sposób zaprojektowaliśmy całą stronę WWW, zarówno jej treści, jak i (po części) kod strukturalny. Aby ułatwić sobie wykonanie kolejnych zadań, zbierzmy wszystkie uwagi, jakie nam się nasunęły do tej pory:

- ◆ Zakładka *Start* jest do usunięcia, a jej funkcjonalność przeniesiona zostanie do logo restauracji.
- ◆ Obrazek powitalny otrzyma rozmiar wynoszący 100% wysokości ekranu.
- ◆ W miarę możliwości należy wykorzystać efekty graficzne dostarczane przez CSS3 (takie jak jasność, stopień rozmycia, kontrast itd.).

- ◆ Dla galerii wersja *full screen* jest niezalecana, a do budowy samej galerii zostanie wykorzystane narzędzie Bootstrap Image Gallery.
- ◆ Karuzela posłuży do zaprezentowania zespołu pracowników restauracji.
- ◆ Należy pomyśleć nad lekkim przeredagowaniem tekstów.
- ◆ W przypadku podstrony o nagrodach i osiągnięciach należy:
 - ◆ usunąć kolorowe tła ikon oznaczających poszczególne nagrody;
 - ◆ zmniejszyć rozmiary ikon;
 - ◆ zapewnić graficzne wyróżnienie elementu, który w danej chwili jest aktywny;
 - ◆ zastosować Font Awesome;
 - ◆ wykorzystać możliwości zakładek (Tabs) z Bootstrapa.
- ◆ Sekcja *O nas* zostanie oparta na elemencie *Media Object*, który jest natywny w Bootstrapie.
- ◆ W stopce strony wykorzystamy mapę z Google Maps.

Skoro określiliśmy już bardzo powierzchownie treści strony, warto zastanowić się nad ogólnym projektem witryny. Jest to bardzo ważne, ponieważ jeśli wcześniej przyjmimy pewne założenia dotyczące projektu i jego wykonania, oszczędzimy sobie niepotrzebnych rozterek podczas samego pisania. Po zapoznaniu się z kształtem strony mamy optymalny moment do podjęcia tego rodzaju decyzji.

Praca ogólna nad szablonem

Technologiami, które idą współmiernie z pisaniem kodu HTML, są zarówno arkusze CSS, jak i skrypty JavaScript. W tej książce skupimy się na pierwszej z nich. Obecnie funkcjonują setki rozwiązań usprawniających tworzenie front-endów aplikacji. Doskonałym przykładem takiego narzędzia jest właśnie Bootstrap, który zapewnia responsywność tworzonych elementów. Dzięki temu projektant może się już tylko skupić na projektowaniu charakterystycznych części strony. Tak zwane preprocesory zdecydowanie ułatwiają pisanie kodu stylizującego stronę. Preprocesorami zastosowanymi w Bootstrapie są Less i Sass (rysunek 4.35).

Rysunek 4.35.

Bootstrap korzysta z Less i Sass



Preprocessors

Bootstrap ships with vanilla CSS, but its source code utilizes the two most popular CSS preprocessors, [Less](#) and [Sass](#). Quickly get started with precompiled CSS or build on the source.

W praktyce można się łatwo przekonać, że preprocesory rzeczywiście ułatwiają pisanie kodu CSS. Co prawda nawet bez tego możemy w CSS korzystać ze zmiennych, ale nie jest to tak wygodne jak przy użyciu preprocesorów, które wstępnie przetwarzają arkusze kaskadowe. Co więcej, kod związany z dziedziczeniem również jest łatwiejszy do napisania:

```
// Declaring variables
@border-width: 1px;
@red: #842210;

// Using variables
div.header-title {
    border: @border-width solid @red;

    .description{
        border-bottom: @border-width+2 dashed #eee;
    }
}
```

Powyższy krótki kod został napisany za pomocą preprocesora Less. Jak można go rozumieć? Jak widać, najpierw zdefiniowano zmienne o różnych typach wartości. Jedna z nich przechowuje wartość wyrażoną w pikselach, druga zaś — kolor w zapisie szesnastkowym. Tak na marginesie: nazwa zmiennej `red` może być myląca, gdyż zmienna ta wcale nie musi nieść informacji o kolorze czerwonym. Niemniej zastosowanie zmiennych nieco zwiększa możliwości programisty. Sam preprocesor potrafi odróżniać działania matematyczne od np. nazwy zmiennej. Bez przerw pomiędzy liniami kodu oraz komentarzy opisany kod zajmuje osiem wierszy. Jak obszerny może być ten sam kod napisany wyłącznie w CSS?

```
div.header-title {
    border: 1px solid #842210;
}
div.header-title .description {
    border-bottom: 3px dashed #eeeeee;
}
```

Jak widać, kod zajął sześć linii. Czy w takim razie Less nie sprawia, że kod jest dłuższy? Otóż niekoniecznie. Na pewno natomiast ułatwia pisanie kodu. W przypadku pierwszego fragmentu kodu mogliśmy dodać wartość do zmiennej, a w przypadku drugiego musieliśmy ją jawnie napisać. Ewentualna zmiana szerokości obramowania z 3px na 4px również byłaby bardziej skomplikowana. Weźmy też pod uwagę, że zaprezentowany tu przykładowy kod nie jest długi, dlatego nie pozwala na pokazanie tego, że można wykorzystać zmienne kilkakrotnie. Wyobraźmy sobie, że chcielibyśmy, aby krawędź elementów w stylach miała o jeden piksel więcej. W Less taką zmianę naniesiemy bardzo szybko, przez zwiększenie wartości zmiennej o jeden, jeśli zaś operujemy bezpośrednio na stylach, musimy przeglądać cały kod i ręcznie zmieniać poszczególne wartości. Jak widać, Less może okazać się przydatny. Pamiętajmy jednak, że w praktyce i tak otrzymujemy kod CSS. W efekcie interpretacji plików o formacie `.less` uzyskujemy kod w postaci normalnego arkusza stylów. Należy sobie zatem zadać bardzo ważne pytanie: czy nasza strona jest na tyle obszerna, że wymaga zastosowania dodatkowego narzędzia? Jeśli mówimy o stronie restauracji Bella Food, odpowiedź brzmi: nie. W praktyce jedynie zwolnilibyśmy tworzenie strony. Natomiast jeśli chodziłoby o duży portal, gdzie poprawki trzeba wprowadzać w bardzo szybkim tempie, wykorzystanie preprocesora byłoby bardzo dobrym rozwiązaniem. Nasz projekt jest jednak produktem skończonym, w którego przypadku korekty wynikające z błędów lub ewentualnych aktualizacji będą łatwe do wprowadzenia. Nie zmienia to jednak faktu, że zapoznanie się z tematyką dotyczącą preprocesorów poszerza wiedzę o kodzie CSS, dlatego warto poznać co najmniej jeden z takich preprocesorów.

To także jeden z powodów, dla których nie traktujemy preprocesorów jako narzędzi do bezpośredniego stosowania. Nie każdy zna preprocesory i umie je wykorzystywać. Wyjaśnienie zasad ich używania nie jest celem tej książki. Poza tym równie dobrze można pisać zwyczajny kod CSS.

Drugim ważnym zagadnieniem, o którym już wspominałem, jest pisanie nazw zgodnych z Block Element Modifier. Bootstrap został napisany w dużym stopniu z poszanowaniem tej konwencji. Znowu warto sobie zadać pytanie, czy nasz projekt jest wystarczająco duży, aby stosować tę metodologię. Spójrzmy po raz drugi na fragment kodu:

```
.block__element--modifier {}
```

Jeżeli mamy zamiar pisać kod stylów w taki sposób, musimy mieć na uwadze to, że będziemy musieli stylizować każdy element przy użyciu zwykłego CSS, np.:

```
.aboutus_header--title {}  
.contact_header--title {}  
.gallery_header--description {}
```

Zatem będziemy musieli wystylizować taką klasę. Jak by to wyglądało bez BEM?

```
.aboutus > .header > .title {}  
.contact > .header > .title {}  
.gallery > .header > .description {}
```

Czy w takim razie nie byłoby wygodniej zastosować preprocesora, który pozwala na tworzenie nowych klas w tych już istniejących? Tak, byłoby. Teraz też wiemy, dlaczego Bootstrapa nie oparto w 100% na BEM — kod stylów byłby naprawdę o wiele dłuższy! Lepszym rozwiązaniem jest stosowanie wyłącznie zwykłych nazw klas.

Jeśli teraz przyjrzyysz się liście zagadnień, którą sporządziliśmy jeszcze przed omówieniem użycia preprocesorów oraz BEM, dojdiesz do wniosku, że prawdopodobnie najwięcej pracy czeka nas przy przygotowywaniu prezentacji szczególnych osiągnięć restauracji. Już niedługo się przekonasz, jak bardzo Bootstrap potrafi wspomóc programistę. Bez takiego frameworka trzeba byłoby wykonać o wiele więcej pracy.

Ważnym atutem strony jest szybkość działania. Aby strona bogata w treści pracowała wydajnie, trzeba postarać się o możliwie najszybsze ładowanie się witryny. Właśnie tej kwestii poświęcony będzie kolejny rozdział. Dowiesz się w nim, które elementy można poprawić, aby zwiększyć wydajność działania strony.

ZADANIA

1. Zbadaj, jak działają schematy wykorzystane do tej pory w projekcie.
2. Jeżeli nie znałeś pojęcia poprawności semantycznej kodu, koniecznie się z nim zapoznaj.
3. Spróbuj napisać część strony od nowa, nie używając wtyczki *emmet*, i wyciągnij z tego wnioski.

Skorowidz

A

Atom, 37
 emmet, 40
 minimapa, 44
 polskie znaki, 45
 snippety, 42

B

badanie strony, 115
 optymalizacja, 108
 źródło, 27
BEM, Block Element Modifier, 18
Bootstrap Components, 90
Bootswatch, 47
 arkusze stylów, 49
 przeglądanie wyglądu stylów, 48
 style standardowe, 49

C

CDN, Content Delivery Network, 123
czcionki, 173
część powitalna, 184

D

Datanyze Insider, 26
dodatek
 Datanyze Insider, 26
 YSlow, 112
dokumentacja, 51
DOM, Document Object Model, 20
dostępność stron WWW, 21
dostosowywanie stylów, 169

E

edytor kodu, 37
 Atom, 37
ekran powitalny, 136
elementy charakterystyczne strony, 29
emmet, 40

F

framework CSS
 Foundation, 222
 Material Design Lite, 223
 Semantic UI, 222
front-end, 219

G

galeria zdjęć, 79, 145
generowanie szablonów, 131
GIMP, 67
 grupowanie warstw, 92
 kanał alfa, 69, 70
 okno warstw, 68
 optymalizacja obrazów, 104
 prowadnice, 68
 rozmycie Gaussa, 73
grupowanie warstw, 92

H

HTML, 129

I

język HTML, 129
 JPEG Optimizer, 106

K

kanał alfa, 69, 70
 karuzela, 150, 196
 kody

- CSS, 55, 109
- HTML, 55, 109

 kolejność ładowania plików, 127
 kolorystyka, 50
 kompatybilność z przeglądarkami, 106
 konsola JS, 110

Ł

ładowanie plików, 117, 127

M

MDN, Mozilla Developer Network, 55
 Media Object, 205
 minifikacja plików CSS, 122
 minimapa kodu, 44
 modyfikacja

- gotowych stylów, 118
- pliku HTML, 134

N

narzędzia

- deweloperskie, 15, 108
- do generowania szablonów, 131
- do minifikacji plików, 122
- zewnętrzne, 112

 narzędzie, *Patrz także* program

- Boilerplate, 131
- Initializr, 131
- PageSpeed Insights, 113
- W3Techs, 27
- YSlow, 112

 nawigacja, 71, 179

O

obsługa programu graficznego, 67
 optymalizacja grafiki

- GIMP, 104
- narzędzia online, 105

 outline kodu HTML, 17

P

PageSpeed Insights, 113
 pierwszy projekt, 63
 pixel perfection, 85
 placeholderzy, 57
 plan witryny, 63
 plik index.html, 135
 pliki

- bez konfiguracji, 121
- cookies, 111
- CSS, 122
- HTML, 134
- po konfiguracji, 121

 polskie znaki, 45
 portal

- Bootswatch, 47
- html5doctor, 57
- MDN, 55

 prezentacja zespołu, 150, 196
 program

- Atom, 38
- GIMP, 67

 Progressive Enhancement, 20, 21
 projekt

- galeria zdjęć, 79, 83
- plan witryny, 63
- strona domowa, 82
- strona wejściowa, 71

 przezroczyistość, 69, 70

R

responsywne strony, 217
 rozmiar zdjęcia, 105
 rozmieszczanie elementów, 143

rozmycie Gaussa, 73
 rozpoznawanie użycia Bootstrapa, 26, 31
 rozszerzenie .json, 120
 RWD, Responsive Web Design, 116, 218

S

sekcje, 76
 semantyka kodu, 16
 SEO, Search Engine Optimization, 117
 siatka, grid, 14
 slider, 29
 snippety, 42
 stałe fragmenty kodu, 50
 stopka, 97, 164, 207
 strona

- Bootstrap Expo, 34
- Build with Bootstrap, 34
- builtwith.com, 27
- csscompressor, 123
- domowa, 82
- fontawesome.io, 33
- gfycat.com, 32
- Musicam Sacram, 28
- nba.com, 31
- rozwal.to, 30
- wejściowa, 71

 strony

- dla urządzeń mobilnych, 115
- interaktywne, 88
- mobilne, 117
- responsywne, 217

 style

- normalizujące, 170
- resetujące, 170

 stylizowanie części strony

- cechy restauracji, 187
- część powitalna, 184
- galeria restauracji, 191
- karuzela, 196
- nawigacja, 179
- opinie gości, 205
- osiągnięcia i nagrody, 198

prezentacja pracowników, 196
 stopka strony, 207
 stylizowanie

- klas ogólnych, 173
- strony WWW, 171

 szablon strony, 71

- „Co mówią”, 160
- „Galeria”, 79
- „Jacy jesteśmy?”, 75, 139
- „Osiągnięcia i nagrody”, 88, 155
- „Poznaj opinie”, 93
- „Zespół”, 84

 praca ogólna, 99
 stopka, 97
 szablony HTML, 130
 szybkość pobierania elementów, 108

T

technologia swipe.js, 81
 tekst powitalny, 76
 testy optymalizujące stronę, 111
 treści na stronie, 110

U

urządzenia mobilne, 115, 209

W

W3Techs, 27
 WCAG2.0, 22
 własne style CSS, 169
 wtyczka emmet, 40
 wydajność strony, 103
 wymogi WCAG2.0, 22

Z

zagnieżdżanie kolumn, 14
 zakładka, 198

- Audits, 111
- Console, 110
- Network, 110
- Tabs, 90

zasada pixel perfection, 85
zastosowanie Bootstrapa, 25
zdjęcia, 79
 stockowe, 57
znacznik
 img, 86
 li, 42
 main, 17

Ź

źródło strony, 27

PROGRAM PARTNERSKI

GRUPY WYDAWNICZEJ HELION



1. ZAREJESTRUJ SIĘ
2. PREZENTUJ KSIĄŻKI
3. ZBIERAJ PROWIZJĘ

Zmień swoją stronę WWW
w działający bankomat!

Dowiedz się więcej i dołącz już dzisiaj!

<http://program-partnerski.helion.pl>

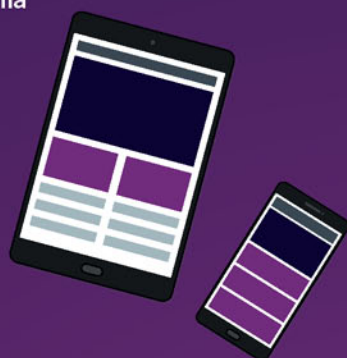
Podczas przeglądania stron WWW bardzo szybko się przekonujemy, że liczy się nie tylko prezentowana treść. Każdy internauta doceni stronę o eleganckim i uporządkowanym układzie, której zawartość szybko się ładuje i która wygląda tak samo dobrze w różnych przeglądarkach i na różnych urządzeniach, w tym mobilnych. Niestety, wciąż spotyka się mnóstwo stron, które nie spełniają tych standardów!

Niniejsza książka jest przeznaczona dla projektantów witryn WWW, którzy chcą pisać efektywne kod, ale też zadbać o warstwę wizualną tworzonych stron. Dzięki lekturze poznasz Bootstrapa — popularny framework CSS — a także ogólne zasady pisania kodu z jego użyciem. Znajdziesz tu kluczowe informacje o korzystaniu ze stylów w Bootstrapie i modyfikacji stylów wbudowanych. Dowiesz się więcej o zwiększaniu wydajności tworzonych stron i poznasz inne przydatne narzędzia deweloperskie, które znakomicie wzbogacą warsztat profesjonalnego projektanta stron WWW.

W tej książce znajdziesz między innymi:

- ogólne zasady pisania kodu HTML i CSS
- pomoc w przygotowaniu środowiska do tworzenia stron
- zasady projektowania aplikacji — od rozplanowania elementów strony po optymalizację gotowego kodu
- zaawansowane metody pracy z szablonami Bootstrapa
- opis tworzenia własnych stylów CSS i ich dostosowywania do szczególnych potrzeb

Radostaw Gryczan pasjonuje się tworzeniem stron WWW. Ma na koncie wiele ciekawych projektów stron. Bardzo chętnie dzieli się wiedzą, ma spory talent dydaktyczny. Aktywnie pomaga na kilku forach poświęconych informatyce. Ostatnio zaczął prowadzić cykl wykładów poświęconych podstawom tworzenia stron WWW. Jest miłośnikiem małych, urokliwych miejscowości — kiedy tylko ma okazję, promuje ich walory w mediach społecznościowych.



Bootstrap: pokaż swój najlepszy styl projektowania!

sięgnij po **WIĘCEJ**



KOD KORZYŚCI

Helion

księgarnia internetowa



<http://helion.pl>

zamówienia telefoniczne



0 801 339900



0 601 339900

Helion SA
ul. Kościuszki 1c, 44-100 Gliwice
tel.: 32 230 98 63
e-mail: helion@helion.pl
<http://helion.pl>

Sprawdź najnowsze promocje:
● <http://helion.pl/promocje>
Książki najchętniej czytane:
● <http://helion.pl/bestsellery>
Zamów informacje o nowościach:
● <http://helion.pl/nowosci>

ISBN 978-83-246-9255-2



9 788324 692552

Informatyka w najlepszym wydaniu

cena: 49,00 zł