

Helion



REDESIGN THE WEB

#3 | THE
SMASHING
BOOK EDYCJA POLSKA

Tytuł oryginału: Redesign The Web (The Smashing Book #3)

Tłumaczenie: Marek Pętlicki

Skład: Marcin Chłąd

ISBN: 978-83-246-5613-4

Published 2012 by Smashing Media GmbH, Freiburg, Germany.

All Rights Reserved. Authorized translation from the English language edition published by John Wiley & Sons Limited. Responsibility for the accuracy of the translation rests solely with Helion S.A. and is not the responsibility of John Wiley & Sons Limited. No part of this book may be reproduced in any form without the written permission of the original copyright holder, John Wiley & Sons Limited.

Translation copyright © 2014 by Helion S.A.

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 bierze jednak żadnej odpowiedzialności ani za ich wykorzystanie, ani za związane z tym ewentualne naruszenie praw patentowych lub autorskich. Wydawnictwo HELION nie ponosi również żadnej odpowiedzialności za ewentualne szkody wynikłe z wykorzystania informacji zawartych w książce.

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/redewe>

Możesz tam wpisać swoje uwagi, spostrzeżenia, recenzję.

Printed in Poland.

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

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

- 7** **Wstęp**
Elliot Jay Stocks
- 9** **Biznesowa strona przeprojektowywania**
Paul Boag
- 37** **Wybór platformy: techniczne kwestie przeprojektowywania**
Rachel Andrew
- 71** **Zanurz się w HTML5**
Ben Schwarz
- 93** **Przestyluj, przekoduj, przeobraź za pomocą CSS3**
David Storey i Lea Verou
- 135** **Odkryj JavaScript na nowo:
sztuczki pozwalające zastąpić skomplikowany kod jQuery**
Christian Heilmann
- 161** **Techniki budowania lepszego user experience**
Dmitry Fadeyev
- 195** **Projektowanie z myślą o przyszłości, za pomocą Photoshopa**
Marc Edwards
- 229** **Przeprojektowywanie z osobowością**
Aarron Walter
- 253** **Urządzenia mobilne w projektowaniu interfejsu użytkownika:
„Wersja WWW czy natywna?”**
Aral Balkan
- 283** **Nowy projekt przepływu: podejście z myślą o przyszłości**
Stephen Hay
- 309** **Bądź cudownie elastyczny: projektowanie atomów i pierwiastków**
Andy Clarke
- 333** **Skorowidz**



Bądź cudownie elastyczny: projektowanie atomów i pierwiastków

Andy Clarke

SĄ TAKIE TRZY SŁOWA, które moim zdaniem idealnie podsumowują pracę nad projektami WWW. Są to:

- Responsywny.
- Projekt.
- WWW.

To Ethan Marcotte wymyślił nazwę dla tej kombinacji elastycznych siatek, płynnych obrazów i definicji stylów dla różnych mediów w CSS3. Niedługo potem twórcy stron WWW na całym świecie zaczęli chwalić się przepięknymi szablonami stron, pakietami, skryptami i innymi rozwiązaniami wielu problemów zdefiniowanymi pojęciem wprowadzonym przez Ethana: *responsywnym projektem WWW* (ang. *responsive Web design*).

Responsywny projekt WWW nie dotyczy tylko tego, w jaki sposób projektanci i programiści wykorzystują takie technologie jak media queries w CSS3. Nie chodzi o to, jak obsługujemy obrazy o różnych rozmiarach czy tabele danych. To są tylko wybrane wyzwania technologiczne, ale responsywność nie dotyczy tylko pokonywania problemów. Nie polega na uczeniu się owych języków ani zdobywaniu doświadczenia w ich wykorzystaniu. Chciałbym, żeby to było takie proste.

Ale nie jest.

Projektowanie w sposób responsywny to coś o wiele, wiele bardziej skomplikowanego.

Czy to Ci się podoba, czy nie, responsywny projekt WWW stawia wyzwania całej wiedzy, jaką posiadamy na temat WWW, i dotyczy to wszystkich osób zaangażowanych w proces. Właśnie dlatego w tym rozdziale zademonstruję, dlaczego stare metody projektowania są już nieaktualne, i wprowadzę nowe mechanizmy, które reagują w sposób responsywny i które z powodzeniem zastosowałem w swoich projektach.

PODSTAWA PRACY

Jak sądzę, nie jestem jedyną osobą, która wierzy w to, że responsywne projektowanie stron WWW stanowi fundamentalną zmianę sposobu projektowania. Andy Hume napisał¹⁰⁴:

Dla mnie responsywne projektowanie stanowi kolejny przykład powrotu do ducha WWW. Właśnie dlatego nie jest to jeszcze jeden zbędny dodatek. To podstawa tego, co powinieneś robić.

Zgadzam się z Andym i choć niektóre osoby postrzegają responsywne projektowanie WWW jako jeszcze jeden trend, jakich wiele, ja wierzę, że to prawdopodobnie największa i najważniejsza zmiana w filozofii projektowania stron WWW od momentu powstania tej sieci. Napisałem kiedyś¹⁰⁵:

¹⁰⁴ Andy Hume, *Responsive by Default* (responsywny z założenia), smashed.by/respdef.

¹⁰⁵ Andy Clarke, *I Don't Care about Responsive Web Design* (nie martwię się o responsywne projektowanie WWW), smashed.by/respcare.

Wszystko, co jest sztywne i nieresponsywne, nie powinno być już nazywane projektem WWW, jest czymś innym. Jeśli nie potrafisz wykorzystać wrodzonej płynności sieci WWW, nie jesteś projektantem WWW, jesteś kimś innym. Projektowanie WWW jest responsywne. Responsywne projektowanie WWW jest projektowaniem WWW realizowanym w prawidłowy sposób.

Nie przesadzałem. Podtrzymuję to stwierdzenie, nawet mimo że niektórzy uważają, że mnie nieco poniosło. Wierzę, że analiza tego, w jaki sposób projekt będzie reagował na różnych rozmiarach wyświetlaczy na niezliczonych typach urządzeń, jest jednym z najistotniejszych aspektów pracy nowoczesnego projektanta WWW.

JAK, KIEDY I DLACZEGO?

Responsywne projektowanie WWW zmienia to, co robimy dla WWW, zatem zmienia też sposób, w jaki to robimy. Już nie tylko projektanci i programiści są zarażeni. Responsywne projektowanie WWW wywiera wpływ na wszystkich, którzy wymyślają, projektują, budują, płacą za WWW lub jej używają.

- Strateg treści? Ty.
- Projektant interakcji, doświadczenia lub grafiki. Zgadza się.
- Programista interfejsu lub strony serwerowej? To też Ty.
- Szef lub klient? Zgadłeś.
- Użytkownik? To też Ty, i to w dobrym znaczeniu tego słowa.

Responsywne projektowanie WWW stawia więcej pytań, niż udziela odpowiedzi. Wpływa na pracę i interakcje między wszystkimi osobami zaangażowanymi w proces, od specjalistów od treści i programistów wszelkiej maści, po szefów i klientów, którzy akceptują pracę, którą wykonujemy, i płacą za nią.

Projektowanie responsywne stawia przed nami wyzwania w zakresie tego, *jak, kiedy i dlaczego*. Tym wyzwaniom nie zawsze łatwo sprostać i zmiany, które wprowadzają, nie zawsze są popularne. Należy oczekiwać sporego oporu u ludzi, którzy nie potrafią lub nie chcą zrozumieć potrzeby adaptacji.

TO BYŁO NAPRAWDĘ PROSTE ŻYCIE

W 1998 roku, gdy tworzyłem swoje niewielkie studio projektowe, największym wyzwaniem technicznym były problemy ze zgodnością wyświetlania stron w przeglądarkach Internet Explorer 4 i Netscape 4. Trzeba przyznać, że choć pracowałem z niedojrzałymi technologiami i musiałem radzić sobie z koszmarnymi przeglądarkami, życie miałem wówczas znacznie prostsze. Wszyscy takie mieliśmy.

Naprawdę proste życie.

Oto schemat mojej pracy, założę się, że Twój był taki sam. Może nadal taki jest?

1. Tworzyłem projekt w Adobe Photoshop lub Macromedia Fireworks: pojedynczy projekt, który miał działać dla wszystkich, niezależnie od tego, z jakiej przeglądarki korzystali i na jak dużych ekranach.
2. Demonstrowałem projekt klientom w postaci makiet (które nazywałem statycznymi prezentacjami wizualnymi, ponieważ były nieinteraktywne).
3. Po konsultacji z klientami nanosiłem poprawki. Ich wynik eksportowałem do kolejnego zestawu statycznych prezentacji. Po dwóch, trzech lub czterech iteracjach projekt był akceptowany, wówczas kroilem go na fragmenty w HTML-u i CSS-ie, wysyłałem na serwer i wracałem do domu na herbatę i powtórki *Animal Magic*¹⁰⁶.

W tamtych czasach tworzenie pojedynczego projektu było jak najbardziej do przyjęcia, ponieważ dla nas, naszych szefów i klientów sieć WWW była medium obsługiwanym z platformy desktopowej. Dziś jednak nie możemy sobie pozwolić na komfort pojedynczego projektu. Życie już nie jest tak proste, ponieważ kiedyś mieliśmy dwie istotne przeglądarki, dziś mamy ich potencjalnie setki na różnych rodzajach urządzeń, na których ludzie korzystają z zasobów WWW.

WSZYSTKO PŁYNIE

Nawet już na przełomie tysiącleci powinniśmy zacząć się orientować, że sieć WWW znacząco różni się od innych mediów i że część jej unikalności polega na braku kontroli nad tym, w jaki sposób ludzie ją oglądają i wchodzą w interakcje z treściami i usługami przez nią świadczone. Prawda jest taka, że od początku powinniśmy tworzyć nasze projekty tak, aby uwzględnić elastyczną naturę sieci WWW.

W rzeczywistości było kilka osób, które zdawały sobie sprawę z tego faktu i próbowały nas uświadomić, że żyliśmy w błędzie. W 2000 roku John Allsopp napisał tekst, który jest przez wiele osób uznany za najważniejszy artykuł na temat projektowania WWW w historii: *A Dao of Web Design* (Tao projektowania WWW). W tym artykule John napisał¹⁰⁷:

Kontrola, jaką projektanci znają z mediów drukowanych i często pożądana w mediach WWW, jest po prostu efektem ograniczeń kartki papieru. Musimy pogodzić się z tym, korzystać z faktu, że projektowanie WWW nie ma takich ograniczeń, i projektować z myślą o tej elastyczności. Ale w pierwszej kolejności musimy się pogodzić z tym, że wszystko płynie.

¹⁰⁶ smashed.by/anmag.

¹⁰⁷ John Allsopp, *A Dao of Web Design* (Tao projektowania WWW), smashed.by/dao.

Dobra rada.

Czy z niej skorzystaliśmy?

Nie żartuj.

Co zrobiliśmy zamiast tego?

Oszukiwaliśmy się, że mamy kontrolę. Próbowaliśmy wymusić ten sam poziom kontroli, jaki mieliśmy w druku. W znacznym stopniu ignorowaliśmy fakt, że WWW jest elastyczna i próbowaliśmy zmusić ją do tego, żeby była nieruchoma. Oto kilka przykładów naszych zabiegów.

PROJEKTOWANIE OD KRAWĘDZI KONTENERA

Gdy dokonałem przejścia z druku do WWW, to medium było zupełnie nowe. Nie było żadnych reguł. Dlatego ja i inni projektanci podobni do mnie zaimportowaliśmy zasady, idee i narzędzia, które doskonale znaliśmy. Ludzie, którzy nas wynajmowali, również nie znali się na WWW, dlatego oni także zaadaptowali swoją wiedzę z redakcji i drukarni i wykorzystali to, co zawsze dla nich działało.

Do projektowania stron WWW używałem Photoshopa i każdy nowy projekt rozpoczynałem od pustego dokumentu: tworzyłem prostokąt w układzie pejzażu i wypełniałem go pomysłami.

Te pierwsze prostokąty miały 640 pikseli szerokości i 480 wysokości, ponieważ wówczas taka była rozdzielczość większości monitorów w komputerach PC. Projektanci byli przyzwyczajeni do kontroli, dlatego robiliśmy wszystko, co w naszej mocy, aby ją zachować, między innymi takie idiotyzmy jak pisanie skryptów, które nie pozwalały zmienić wymiarów okna przeglądarki, albo opakowywanie zawartości w pięć lub więcej zagnieżdżonych ramek¹⁰⁸. Ten prostokąt o szerokości 640 pikseli był dość ciasny, ponieważ (zapewne to pamiętasz) klienci nie chcieli zmuszać użytkowników do przewijania treści, więc wszystko musiało (po prostu musiało) mieścić się jednocześnie na ekranie. Ekran o szerokości 640 pikseli wkrótce ustąpiły rozdzielczości 800×600, więc projektanci po prostu rysowali większe prostokąty. Wizja większej przestrzeni była upajająca, ale w niedługim czasie i te wielkie prostokąty zostały wypełnione i zaczęliśmy patrzeć na następną okazję: rozdzielczość 1024×768. I znów, i jeszcze raz, na każdym etapie, studiowaliśmy logi serwerów i łamaliśmy głowy, czy to już czas, żeby przełączyć się na większą szerokość prostokąta.

To może wydać się dziwne, ale w 2005 roku dyskusja nad tym, czy lepsze są układy statyczne, czy płynne, była bardzo gorąca. Poważnie.

Kiedyś przeprowadziłem wywiad z Jasonem Santa Marią na temat jego nowego projektu strony A List Apart, serwisu dla osób tworzących strony WWW¹⁰⁹. Dziwiłem się, dlaczego Jason zdecydował się ustawić szerokość strony na 1240 pikseli, zamiast zastosować płynny układ oparty na procentach. Jason odpowiedział¹¹⁰:

¹⁰⁸ Dokument FAQ aplikacji Dreamweaver, *Using Frames To Align Page Content* (używanie ramek do wyrównania treści strony), smashed.by/tut.

¹⁰⁹ Jason Santa Maria, *A List Apart Redesign* (projekt serwisu A List Apart), smashed.by/mari.

¹¹⁰ Andy Clarke, *A List (taken) Apart* (A List Apart rozłożony na części), smashed.by/andy.

A List Apart zawsze starał się być serwisem wyprzedzającym inne koncepcyjnie. Nie obsługujemy już rozdzielczości 800×600 ani oczywiście 640×480. Czy wy tak robicie? Gdy strony przestały poprawnie się wyświetlać w rozdzielczości 640×480, ludzie wpadli w szat... w tej chwili nikogo to już nie obchodzi. Wszystko się zmienia. Zaufaj mi, niedługo przekonasz się, że coraz więcej stron zacznie się rozszerzać, jakby chciały wygodnie się rozciągnąć, a nie gnieździć z podkulonymi kolanami.

Ludzie oburzyli się wówczas na nowy, szerszy A List Apart. No może nie aż tak drastycznie. Jeremy Keith (o którym trudno powiedzieć, że nie jest konsekwentny) powtarzał wątpliwości Jona Hicksa¹¹¹ na temat rozbieżności między „rozsądkiem projektantów a preferencjami użytkowników”. Jeremy powiedział wówczas¹¹²:

Argumenty o szerokościach 640, 800 czy 1024 piksele są niczym dyskusja nad tym, czy pepsa smakuje lepiej niż coka, gdy tak naprawdę potrzebujesz szklanki wody. Zabawa w cyferki to ściema. Wielka ściema o ustalonej szerokości.

Przez lata oszukiwaliśmy się, że skoro 640, 800, 1024 piksele były najpopularniejszymi szerokościami ekranów, powinniśmy projektować strony tak, aby miały taką szerokość. Desperacko czepiliśmy się ustalonych wymiarów, ponieważ rzeczywistość, czyli fakt, że sieć WWW jest medium o płynnych granicach i niestalonych wymiarach płótna, bez krawędzi, była zbyt trudna do zaakceptowania.

SIEĆ WWW NIE MA KRAWĘDZI

Dziś nie jesteśmy w stanie przewidzieć rozmiaru ani formatu, w jakim będzie oglądana nasza strona, zatem sieć WWW efektywnie nie ma krawędzi. Co zatem robią projektanci?

Tworzymy krawędzie... Gdy szerokość 640 została zastąpiona przez 800, a ta następnie przez 1024, stosowaliśmy coraz szersze płótna jako punkt wyjścia dla naszych projektów. Rysowaliśmy coraz większe prostokąty, a następnie wypełnialiśmy je naszą treścią, zaczynając od krawędzi i kontynuując do środka.

Gdy Steve Jobs (pokój jego duszy) w 2007 roku wyciągnął z kieszeni pierwszego iPhone'a z przeglądarką Safari, zademonstrował dwie orientacje ekranu w jednym urządzeniu. Zaimplementowane w iPhone gesty przesuwania i rozciągania udowodniły ostatecznie, że krawędzie nie mają znaczenia. Sieć WWW zmieniła się tego dnia na zawsze, dzięki temu jednemu urządzeniu. Co zatem zrobiliśmy my, projektanci?

Narysowaliśmy maleńki prostokąt. 320 pikseli szerokości, 480 wysokości.

¹¹¹ Jon Hicks, *Is 1024 OK? (czy 1024 jest OK?)*, smashed.by/hicks.

¹¹² Jeremy Keith, *A List Too Far Apart? (A List Apart rozstawiony za szeroko?)*, smashed.by/runaway.

Gdy Steve usiadł na kanapie i pokazał pierwszego iPada, ponownie ustawiliśmy nasze płótna na wymiary 1024 na 768 pikseli. Tak bardzo zafiksowaliśmy się na wymiarach płótna programu graficznego, ponieważ wiedzieliśmy, o co w tym chodzi, i tego oczekiwali szefowie i klienci.

Czy słyszałeś taką definicję obłędu: robienie za każdym razem tego samego, oczekując innego wyniku? Właśnie tak pracowaliśmy dotychczas. Prostokąt to się rozszerzał, to kurczył, to znów rozszerzał. Liczba wymiarów prostokąta ciągle rosła, ale nasze myślenie pozostawało bez zmian.

Z NOŻEM NA STRZELANINĘ

Nie powinniśmy być dla siebie zbyt surowi. Narzędzia, którymi kiedyś się zachwycaliśmy, wykonały swoje zadanie tak dobrze, jak były w stanie, przez długi czas utrzymując przy życiu wizję WWW o ustalonych wymiarach. Pomyśl o tym. Jaka jest pierwsza operacja, gdy zaczynasz nowy projekt w Fireworks lub Photoshopie?

Plik/Nowy

⌘N.

A następnie ustalasz wymiary dokumentu.

Zadaj sobie pytanie, czy istnieje powód tego, że tak wiele stron ma ustawioną na sztywno szerokość i wycentrowany układ. Czy to nie jest efekt tego, że klient widział, a następnie zaakceptował projekt stworzony na obrazku o ustalonej szerokości?

Dostawcy oprogramowania graficznego, jak Adobe, uzupełnili swoje produkty o mechanizmy wspomagania projektowania stron WWW, ale nie znajdziesz w nich niczego, co pomogłoby projektować w sposób responsywny. Nie są w stanie nawet pomóc nam w demonstrowaniu klientowi stanów i przejść między nimi, nie mówiąc już o obrazowaniu sposobu, w jaki elastyczny układ wpływa na zmiany rozmieszczenia elementów. Ponownie zacytuję Jasona Santa Marię¹¹³:

Każdy element strony może wpłynąć na położenie innych elementów. Musimy być w stanie określić, co ma się stać z elementem (float, clear, wrap itp.), gdy coś takiego się wydarzy. Dodatkowo okno przeglądarki jest płótnem o zmiennych rozmiarach. Aplikacje desktopowe działają wyłącznie na ustalonych wymiarach płótna, projektowanie w sposób płynny i elastyczny wymaga czegoś więcej niż tylko wyobraźni.

Obecnie używane przez nas narzędzia, w szczególności Photoshop i Fireworks, po prostu nie są przystosowane do obsługi wymagań projektowania responsywnego. Przynoszą nóż na strzelaninę.

¹¹³ Jason Santa Maria, *A Real Web Design Application* (realne zastosowania projektowania WWW), smashed.by/rwdapp.

TEN KRĘPUJĄCY FAKT

Przez długi czas najtrudniejszym zagadnieniem projektowania WWW nie był fakt, że wielu użytkowników korzystało z naszych projektów na wyświetlaczach różnych wymiarów. Działo się tak głównie dlatego, że *nie* przejmowaliśmy się tym.

Po prostu ignorowaliśmy ten krępujący fakt i kontynuowaliśmy swoją robotę, wierząc, że większość ludzi posiada ekrany o wystarczających wymiarach, aby nasz projekt wyświetlił się na nich bez problemu.

Gdy iPhone uświadomił nam w końcu, że nasza praca nie zawsze będzie *najlepiej wyglądała na ekranie o szerokości 1024 pikseli*, w pierwszym odruchu zaczęliśmy tworzyć projekty dedykowane dla iPhone'a, które były niezależne od tych dla desktopu. To automatycznie spowodowało podwojenie czasu niezbędnego na tworzenie projektu, uwagi od klienta, poprawki i zatwierdzenie.

Gdy na rynku pojawiło się więcej smartfonów, eReaderów i tabletów, szefowie i klienci zaczęli prosić o dedykowane wersje projektu strony również dla urządzeń takich jak¹¹⁴:

- iPhone
- Android
- iPad
- Kindle Fire

Ale tworzenie niezależnych wersji dla każdego z nich powoduje trzy-, cztero-, a nawet pięciokrotne zwiększenie ilości czasu niezbędnego na projektowanie. Trzy-, cztero-, pięciokrotne wydłużenie cyklu akceptacji. Trzy, cztery, a nawet pięć razy więcej spotkań!

Z tak wielką liczbą charakterystyk, możliwości i wymiarów tworzenie niezależnych projektów dla każdego z wymienionych urządzeń nie ma sensu, ekonomicznego ani praktycznego, nie jest też najbardziej efektywnym sposobem wykorzystania czasu i talentu projektanta. Tworzenie kilku projektów o ustalonych wymiarach nie ma żadnych zalet również ze względów technicznych, gdy projekty zaczniemy przekładać na kod. Jak Stephanie Rieger napisała w *The 'trouble' with Android*¹¹⁵:

Projektowanie z zastosowaniem sztywnych wymiarów ekranu w rzeczywistości nigdy nie było dobrym pomysłem... po prostu istnieje za wiele różnic nawet pomiędzy „najpopularniejszymi” urządzeniami.

Zamiast myśleć o niezależnych projektach dla każdego z urządzeń, powinniśmy zacząć myśleć, jak sugeruje Ethan Marcotte¹¹⁶, o pojedynczym projekcie, który ma wiele powierzchni, ułożonych zgodnie z pewnym konceptem, czymś na wyobrażenie *płynnego kontinuum*:

¹¹⁴ Powszechnie stosowane wymiary i gęstości ekranów smartfonów znajdziesz w rozdziale 7.

¹¹⁵ Stephanie Rieger, *The 'trouble' with Android* („kłopot” z Androidem), smashed.by/trouble.

¹¹⁶ Ethan Marcotte, *Responsive Web Design* (responsywne projektowanie WWW), smashed.by/rwdala.

Zamiast dopracowywać niezależne projekty dla każdego z nieskończonej liczby urządzeń, możemy potraktować je jako różne punkty widzenia na to samo doświadczenie.

Podoba mi się ten sposób pojmowania projektowania responsywnego: różne punkty widzenia na to samo, ponieważ doskonale wpasowuje się w koncepcję „wszystko płynie” Johna Allsoppa opisaną w jego Dao. O projekcie lubię myśleć jak o rzece, która zaczyna się jako strumyk i staje się coraz szersza, aż wpada do morza. Nigdy nie wiemy, w którym miejscu jej nurtu użytkownik będzie miał okazję jej doświadczyć, dlatego musimy być elastyczni.

Ten fakt nasuwa pytanie: jeśli nie możemy używać niezależnych projektów o ustalonej szerokości, co powinniśmy zrobić? W jaki sposób tworzyć projekt? W jaki sposób projektować WWW bez krawędzi?

„Hej, zobacz, co zrobiłem”

Dziś zupełnie normalne jest tworzenie kilku statycznych prezentacji tworzonego projektu. Istnieje kilka powodów takiego zachowania.

DLA PROJEKTANTÓW

Photoshop i Fireworks są niesamowitymi narzędziami do eksperymentowania, które pomagają nam rozwiązywać problemy i definiować wizualny kierunek projektu. Czasem sam fakt używania Photoshopa czy Fireworksa jest już pomocny, niekiedy przypadkowo pozwalając nam osiągnąć wyniki, których nie bylibyśmy w stanie uzyskać za pomocą innych narzędzi. Wynik możemy pokazać szefom i klientom, mówiąc: „Hej, zobacz, co zrobiłem”.

Problem ze statycznymi prezentacjami, jak pisałem już wielokrotnie, polega na tym, że nie potrafia przekazać rzeczywistego zachowania interaktywnej strony WWW. Gdy są użyte w nieprawidłowy sposób, mogą wywołać błędne oczekiwania. Projektanci używają tego typu prezentacji w celu uzyskania akceptacji projektu, a szefowie i klienci oczekują, że końcowy wynik będzie wyglądać dokładnie tak jak na prezentacji. W końcu właśnie to im pokazaliśmy.

Ale nie wszystkie przeglądarki są w stanie wyświetlić dokumenty w idealnie taki sam sposób. Wszystkie mają inne cechy, które są ignorowane przez statyczny projekt. Z tego powodu to tylko nasza wina, że musimy później spędzać godziny nad HTML-em, CSS-em i JavaScriptem w daremnych próbach dostosowania do statycznego projektu wyglądu strony w każdej przeglądarce.

Z tego też powodu w przyszłości projektanci muszą zaakceptować fakt, że to, co zobrazują na statycznych prezentacjach, nie musi wyglądać identycznie na każdym urządzeniu. Muszą się też nauczyć wywoływania odpowiednich oczekiwań w odbiorcach tych statycznych prezentacji: to tylko przykład wyglądu strony na niektórych platformach, ale nie na wszystkich.

DLA PROGRAMISTÓW

Statyczne prezentacje wizualne mogą być użyteczne dla programistów, którzy otrzymali zadanie implementacji projektu. Prezentacje służą jako schemat dla programistów oraz jako wzorzec dla szefów i klientów, którzy będą chcieli porównać, w jakim stopniu ostateczny wynik będzie pasować do wizji projektowych przekazanych w prezentacji.

Jednak od tej pory programiści muszą zaakceptować fakt, że statyczna prezentacja stanowi jedynie wzorzec wyglądu strony. Programista musi się nauczyć samodzielnie podejmować decyzje projektowe, dotyczące między innymi tego, w jaki sposób projekt dostosowuje się do zmian rozmiaru ekranów i urządzeń.

DLA KLIENTÓW

Najbardziej wartościowym etapem procesu projektowania są często dyskusje na temat tych statycznych prezentacji z klientami i szefami. Jeśli zostaną zdefiniowane odpowiednie oczekiwania, a szefowie i klienci rozumieją znaczenie tego, co widzą, statyczne prezentacje mogą stanowić nie-samowitą pomoc w definiowaniu planu i wywoływaniu komentarzy i konstruktywnej krytyki.

Problem polega na tym, że odpowiednie oczekiwania rzadko bywają zdefiniowane, a im więcej oczekujemy od statycznych prezentacji wizualnych, tym stają się one mniej odpowiednio i mniej efektywne w charakterze narzędzi do przedstawienia projektu klientowi. W szczególności prezentacje o stałej szerokości nie reagują na zawartość, a dodatkowo ludziom łatwo jest pomylić projekt z układem strony.

Projekt nie (zawsze) jest układem

Czy klient kiedykolwiek powiedział Ci „nie podoba mi się projekt”?

Jeśli to Ci się przytrafiło, czy spróbowałeś podrażnić temat i dowiedzieć się, co konkretnie w projekcie mu się nie spodobało? Załóżę się, że nie były to dobrane czcionki ani sposób ich użycia. Nie zostały nawet zauważone. Nie był to też sposób, w jaki użyłeś koloru. Ani elegancko dobrane linie, ramki i cienie wokół kontenerów. Być może szczegóły brzmiałyby jak: „Pasek boczny powinien być po lewej, nie po prawej”.

Innymi słowy, klient miał na myśli układ, ale krytykował projekt.

Nierzadko zdarza się, że szefowie i klienci traktują układ w tych samych kryteriach co inne zagadnienia projektu: typografię, kolor czy teksturę. Projektanci zajmują się również tym zagadnieniem, ponieważ od lat tworzyli i demonstrowali statyczne prezentacje, w których były zawarte wszystkie te rzeczy.

Od dziś musimy zaakceptować, że projekt i układ są wzajemnie niezależnymi problemami. Projekt komponentów i ich wzajemne ułożenie w poziomie i w pionie stanowią niezwiązane zagadnienia. Aranżacja układu komponentów będzie bez wątpliwości zupełnie inna na różnych szerokościach ekranu, ale projekt tych komponentów z pewnością pozostanie niezależny od układu.

W PIERWSZEJ KOLEJNOŚCI ZAPROJEKTUJ KOMPONENTY

Zastanówmy się przez chwilę nad komponentami, które wykorzystujemy w praktycznie każdym projekcie. Prawie na pewno musisz zaprojektować następujące elementy:

- kontenery;
- tabele danych i inne typy paneli;
- obrazy (treści i ikonografię);
- elementy interfejsu (karuzele lub przewijacze itp.);
- nawigację (na wielu poziomach);
- czcionki.

Zachowaj czujność. To jest istotny moment. Tworzenie projektu w oderwaniu od układu sprowadza się do definiowania stylu tych komponentów. To, w jaki sposób wygląda każda z tych części, będzie definiowało wygląd całości. A dokładniej: wygląd i zachowanie każdego z tych komponentów powinny być zaprojektowane w pierwszej kolejności, na długo zanim zaczniesz zastanawiać się nad ich układem. O takim podejściu lubię myśleć jak o pracy od absolutnych podstaw: na poziomie atomów.

Zdaję sobie sprawę, że idea projektowania komponentów w oderwaniu od kontekstu może wydać się dziwaczna, szczególnie jeśli przyzwyczyłeś się (jak większość z nas) do projektowania stron WWW w „tradycyjny” sposób. Ale prawda jest taka, że od dłuższego czasu i tak wykorzystywaliśmy podobnego typu abstrakcyjne koncepcje.

TABLICE NASTROJÓW

Tablice nastrojów od dawna były doskonałym narzędziem do gromadzenia materiałów na inspiracje. Niezależnie od tego, czy kolekcje obrazów zapisywałeś na swoim komputerze, czy w aplikacjach typu Evernote, usługach online jak Pinterest, czy po prostu przypinałeś pinezkami do tablicy korkowej, kombinacja różnych elementów była pomocna do wykreowania odpowiedniego nastroju.

Nasze tablice nastrojów mogą mieć nowoczesny, zabawny albo tradycyjny wydzźwięk oraz, nieważne jak to nazwiesz, wygląd, nastrój czy osobowość. Najważniejsze jest to, że za ich pomocą definiujemy styl. Oczywiście projektowania nie musimy zaczynać od papieru, nożyc i kleju. Jeśli wolisz, możesz nie brudzić rąk.

Jeżeli projektujemy w sposób responsywny, możemy zacząć od projektowania komponentów jako indywidualnych, ale stylistycznie powiązanych części projektu.

Być może Adobe Photoshop i Fireworks nie są idealnymi narzędziami do projektowania stron WWW, ale są wartościowe do projektowania różnych typów komponentów. Aby na tym etapie uniknąć myślenia o układzie, spróbuj utworzyć dokument Photoshopa albo Fireworksa o wymiarach 10 000 na 10 000 pikseli albo większy. Na tak wielkim płótnie pogrupuj elementy na nagłówki, style elementów tekstowych, formularzy, przycisków, komunikatów o błędach itd. Gdy to będzie konieczne, przesuвай komponenty na płótnie, porównując, jak wyglądają w różnych zestawieniach.

Trent Walton napisał¹¹⁷:

Projektanci WWW muszą na projekt patrzeć ponad układem elementów strony i wyobrażać sobie, w jaki sposób elementy będą zmieniały wzajemne ułożenie przy różnych szerokościach ekranu, nie tracąc przy tym formy i hierarchii. Media queries mogą być użyte do czegoś więcej niż naprawianie popsutych układów. Z odpowiednim planowaniem możemy zacząć tworzyć choreografię elementów proporcjonalnie do rozmiaru ekranu, serwując najlepsze z możliwych doświadczenie przy jego dowolnej szerokości.

Odseparowanie projektu od układu osiągnane dzięki pracy na poziomie atomowym jest ważnym zjawiskiem, ponieważ pomaga każdemu (projektantowi i jego szefowi lub klientowi) skupić się na szczegółach projektu, a jednocześnie pozwala uniknąć definiowania oczekiwań co do tego, w jaki sposób komponenty będą ułożone na ekranach różnych rozmiarów.

Przez kilka ostatnich miesięcy do celów prezentacji projektów klientom używałem arkuszy projektowych (ang. *design sheets*). Odkryłem, że są niezwykle efektywne, choć ich użycie wymaga odrobiny czasu na początku, w celu wyjaśnienia klientom zasady ich działania. Dzięki nim możemy zadać klientom pytanie: „Czy to jest nastrój, który chcemy osiągnąć?”

Arkusze projektowe są doskonałym narzędziem pozwalającym na bardzo wczesne ustalenie aspiracji klienta. Zamiast zachęcać go do stwierdzeń typu „Podoba mi się ten projekt”, to podejście pomaga zachować precyzję oceny poszczególnych elementów projektu:

¹¹⁷ Trent Walton, *Content Choreography* (choreografia treści), smashed.by/trent.

- Jak czysty lub minimalistyczny powinien być projekt?
- W jaki sposób użyjemy koloru do zasugerowania akcji?
- Ile będzie przejść tonalnych i jaki będzie kontrast?
- W jaki sposób będą użyte czcionki do rozróżnienia typów treści?
- Jak wiele będzie poziomów hierarchii typograficznej?

Projektowanie komponentów w ten sposób może pomóc znaleźć odpowiedź na te pytania i daje każdej zaangażowanej osobie możliwość zmiany zdania, zanim włożymy dużo wysiłku w projekt lub implementację.

Używanie arkuszy projektowych pozwala nam również na stosowanie iteracji projektowych, nawet jeśli inne aspekty strony WWW nie są jeszcze gotowe. Dzięki temu mamy możliwość porzucenia archaicznego przepływu pracy przemyczonego z drukarni do sieci WWW. Znasz to doskonale:

Plan → Zatwierdzenie → Projekt → Zatwierdzenie → Implementacja

Dzięki temu, że pracujemy w sposób bardziej elastyczny, projekt może zająć miejsce różnych etapów, czasem nawet można go zacząć, zanim zakończy się etap planowania, a ukończyć długo po rozpoczęciu etapu implementacji. Dzięki temu projektowanie staje się głęboko zintegrowane i jest procesem ciągłym w całym cyklu tworzenia strony.

Kafelki stylów

Sprowadzenie projektowania do wyglądu i zachowania elementów to coś, o czym myślała również projektantka Samantha Warren. Swoją technikę nazwała kafelkami stylów (ang. *style tiles*). Samantha tłumaczy¹¹⁸:

Kafelek stylów jest wizualną „paletą”, na której naniesione są wzory, materiały i farby realizujące cele klienta. Stworzyłam szablon Photoshopa z wymaskowanymi obszarami, w których [...] wyświetlam przykłady przycisków, elementów nawigacyjnych i możliwości typograficznych.

Dla Samanthy kafelki stylów są efektywnym narzędziem do komunikacji oraz projektowania na poziomie komponentów¹¹⁹.

¹¹⁸ Samantha Warren, *Style Tiles as a Web Design Process Tool* (kafelki stylów jako narzędzie projektowania WWW), smashed.by/styletiles.

¹¹⁹ Bardziej szczegółowy przegląd procesu projektowania Samanthy znajdziesz na stronie www.styletil.es.

BOOTSTRAP

Dla niektórych projektantów komponenty strony mogą być wykorzystane również w przyszłych projektach. Należą do nich Mark Oto i Jacob Tornton, para projektant-programista pracująca dla Twittera. Są twórcami projektu Bootstrap¹²⁰. Być może sieć WWW nie ma krawędzi, ale zawiera treści. Treści te wyrażamy odpowiednimi elementami HTML, takimi jak:

- nagłówki na poziomach od 1 do 6;
- akapity, cytaty i elementy tekstowe;
- listy: definicje, numerowane i nienumerowane;
- media: obrazy i ich etykiety (ang. *caption*);
- tabele;
- formularze: przyciski, elementy i etykiety.

Oprócz systemu siatki stworzonego na cele Twittera Bootstrap definiuje domyślne projekty „typografii, formularzy, przycisków, tabel, siatek, nawigacji i inne”. Bootstrap jest interesujący z kilku powodów. Nie tylko stanowi solidną bazę szablonów dla projektów, ale dzięki temu, że definiuje okazały zestaw elementów HTML wraz z wybranymi stylami domyślnymi, jest idealnym punktem początkowym dla nowych projektów.

DRIBBBLE

Dribbble¹²¹ jest stroną, na której projektanci dzielą się niewielkimi zrzutami (maksymalnie 400 na 300 pikseli) projektów, nad którymi pracują.

Zrób sobie wycieczkę po podstronach Dribbble, a znajdziesz mnóstwo przykładów projektów na poziomie komponentów i innych detali, którymi chcą się pochwalić projektanci. Natkniesz się na projekty elementów nawigacji oraz kompozycje obrazów. Jeśli interesują Cię formularze, napotkasz style różnych elementów formularzy oraz przycisków w każdym stanie. Nie brakuje też typografii, znajdziesz tyle przykładów czcionek i efektów, ile tylko chcesz.

Projekty prezentowane w serwisie Dribbble są doskonałymi przykładami projektów bez układu. Pokazują, w jaki sposób można projektować i demonstrować naszym szefom czy klientom decyzje projektowe w formacie, który nie pozwala pomylić projektu z układem.

¹²⁰ Twitter Bootstrap, smashed.by/boots.

¹²¹ smashed.by/drbb.

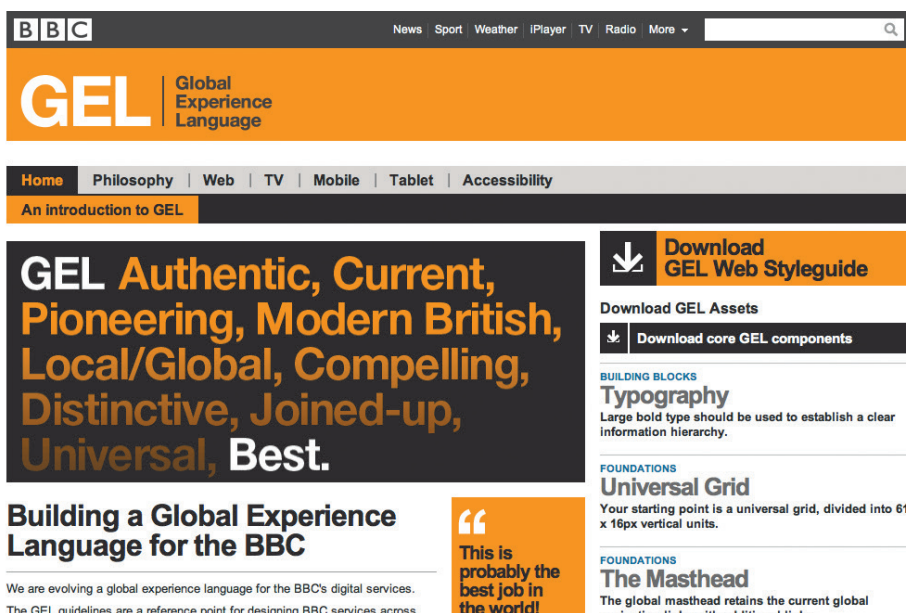
Rysunek 11.1. Na pierwszy rzut oka trudno zrozumieć, dlaczego takie narzędzie jak Bootstrap może być istotne dla twórcy projektów responsywnych. Możesz się zdziwić, ale Bootstrap jest doskonałym punktem wyjścia do tworzenia komponentów stron WWW

Rysunek 11.2. Dribbble prezentuje komponenty projektowe w neutralnym środowisku

ZALECENIA STYLOWANIA

Gdy pracujesz nad projektem, przy którym współpracują zespoły ludzi, precyzyjne i dobrze napisane zalecenia stylu mogą okazać się kluczowe dla zachowania spójności. Organizacje używają zaleceń stylu do różnych celów, wydawnictwa stosują je do definiowania reguł pisowni i interpunkcji, publicyści do zachowania jednolitego stylu w publikacjach. W sieci WWW najlepsze zalecenia stylu w jasny sposób określają zestaw stylów projektowych.

Jednym z najlepszych przykładów zaleceń stylu, jakie widziałem ostatnio, były zalecenia stworzone na potrzeby BBC. Dokumenty pod nazwą Global Experience Language (GEL)¹²² opisują kompletny język wizualny BBC na potrzeby WWW ze szczegółowymi informacjami na temat typografii, ikonografii i projektu komponentów interfejsu, jak warstwy, akordeony i karuzele.

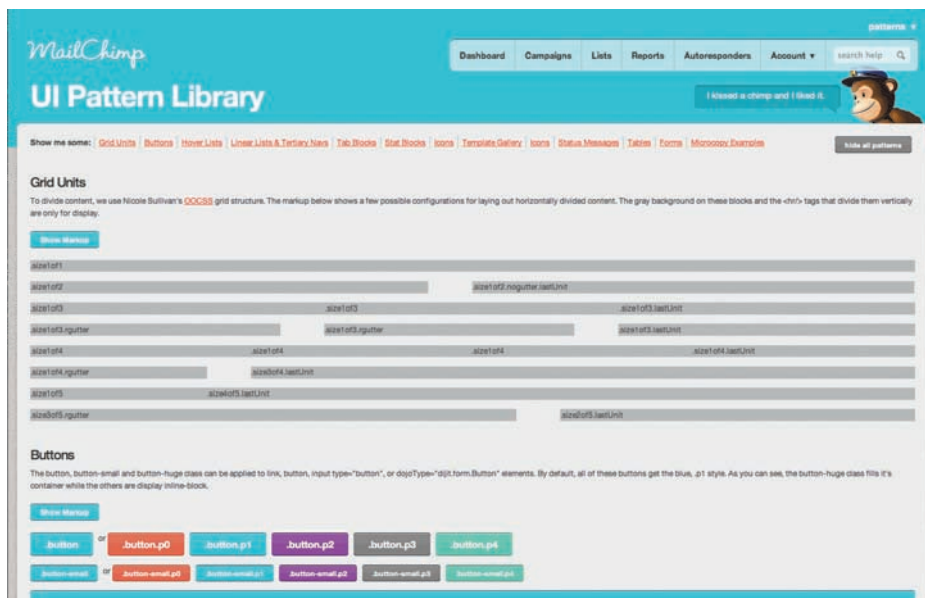


Rysunek 11.3. Zestaw dokumentów Global Experience Language (GEL) stworzony przez BBC opisuje kompletny język wizualny BBC na potrzeby WWW

Ekspert od poczty elektronicznej MailChimp¹²³ stworzył własną bibliotekę wzorców dokumentującą style takich elementów jak przyciski, formularze i zakładki kart. Może się zastanawiasz, dlaczego wspominam o zaleceniach stylu w rozdziale poświęconym projektowaniu responsywnemu, szczególnie biorąc pod uwagę to, że tego typu dokumenty są z reguły tworzone do udokumentowania zasad stylu po zakończeniu projektowania stron WWW.

¹²² smashed.by/gel.

¹²³ smashed.by/mc.



Rysunek 11.4. MailChimp stworzył własną bibliotekę wzorców dokumentującą style elementów interfejsu użytkownika

Podobnie jak Bootstrap Twittera, GEL autorstwa BBC i wzorce projektowe MailChimpa stanowią doskonałe przykłady projektów komponentów. Opisują ze szczegółami, jak ma wyglądać każdy z elementów, i realizują to zadanie, nie odwołując się do układu stron lub robiąc to w minimalnym stopniu.

Najbardziej szczegółowy dokument definiujący style nie opisuje po prostu projektu strony, on jest jej projektem. Dlaczego zatem czekać z jego utworzeniem do zakończenia projektu? Potraktuj zalecenia stylu jako jeszcze jeden przykład projektowania na poziomie komponentów. Używaj ich do udokumentowania, a następnie zaprojektowania każdego elementu, który pojawi się na stronie. W dalszej kolejności możesz wykorzystać te dokumenty do zaprezentowania kierunku, w jakim zmierza projekt, bez ujawniania szczegółów układu, a także uzyskać akceptację, nie wywołując oczekiwań związanych z układem.

UŻYWAJ STATYCZNYCH PREZENTACJI TAM, GDZIE TO MA SENS

Odkryłem, że projektowanie statycznych elementów w pierwszej kolejności naprawdę sprawdzi się w przypadku moich klientów. Projekty przebiegają sprawniej, szybciej i z mniejszą liczbą nieporozumień. Ale rozpoczęcie projektowania od komponentów wydaje się procesem trudnym do opanowania dla niektórych osób. Ponieważ część z nas jest zmuszona do pracy w procesach organizacyjnych, w których statyczne demonstracje wizualne są nadal głównym narzędziem komunikacji między projektantami, programistami i ich szefami, nie każdy ma komfort porzucenia tej formy przekazu.

Projektanci potrzebują widzieć elementy we wzajemnych relacjach. Dzieje się tak, ponieważ rozmiary i proporcje mają znaczenie i nic nie zastąpi wizji tego, jak wszystkie elementy pasują razem. W tej sytuacji statyczna prezentacja projektu z jednym wybranym układem może zawierać mnóstwo informacji. Sam robię coś takiego podczas każdego projektu.

Nie możemy też uciec przed faktem, że przez lata szefowie i klienci tak bardzo przywykli do oglądania pełnoekranowych kompozycji, że niektórzy z nich mogą poczuć się nieswojo, gdy przedstawisz im obrazy zawierające tylko projekty wybranych komponentów bez układu. Czegoś takiego doświadczyłem na wczesnym etapie przełączania się na strategię projektowania komponentów w pierwszej kolejności. Tego typu sytuacje mogą wymagać ostrożności i upewnienia odbiorcy, że dzięki temu podejściu wynik końcowy będzie znacznie lepszy. To jeden z powodów prezentowania projektu klientowi w postaci pełnoekranowej statycznej prezentacji.

Jeśli Twój przepływ pracy zmusza Cię do stosowania pełnoekranowych statycznych prezentacji, przynajmniej na razie, nie oznacza to, że musisz pożegnać się z zaletami rozpoczynania projektowania od komponentów. W każdym razie, jeśli postarasz się ustalić odpowiednie oczekiwania. Nasi szefowie i klienci muszą nauczyć się, że demonstrowane im statyczne prezentacje są po prostu odwzorowaniem atmosfery projektu i tylko jednym z możliwych odwzorowań.

Atmosfera projektu

Jeśli musisz kontynuować wykorzystywanie pełnoekranowych prezentacji układu, nadal możesz rozpocząć projektowanie od komponentów i używać ich na urządzeniach o różnych wymiarach ekranów. Rozłożmy proces projektowania strony WWW (dotyczy to każdego projektu) na składowe:

- **typografia:** czcionki, efekty i odstępy;
- **kolor:** wywoływanie nastroju i podkreślanie akcji;
- **tekstura:** aspekty dekoracyjne, w tym ramki, cienie i kształty;
- **układ:** elementy ułożone na siatce w poziomie i w pionie.

W pierwszej kolejności oddzielmy układ od innych elementów. Pozostaną: kolor, tekstura i typografia. Osobiście lubię nazywać je *atmosferą projektu*, ponieważ wyobrażam sobie koncert albo mecz gry w piłkę, których atmosfera została określona jako elektryzująca. Założę się, że każdy z nas był na przyjęciu, na którym atmosfera była chłodna. Być może nawet zdarzyło się komuś być świadkiem kłótni wywołanej samą atmosferą pomieszczenia.

W projekcie atmosfera opisuje uczucia, jakich doznajemy, które są wywoływane przez kolor, teksturę i typografię. O atmosferze możesz też myśleć w innych kategoriach. Możesz nazywać ją „uczuciem”, „nastrojem” albo „tożsamością wizualną”. Nieważne, jak to nazwiesz, atmosfera projektu nie zależy od układu. Jest niezależna od wzajemnego wizualnego położenia elementów. Można ją widzieć albo czuć na każdym rozmiarze ekranu, na każdym urządzeniu.

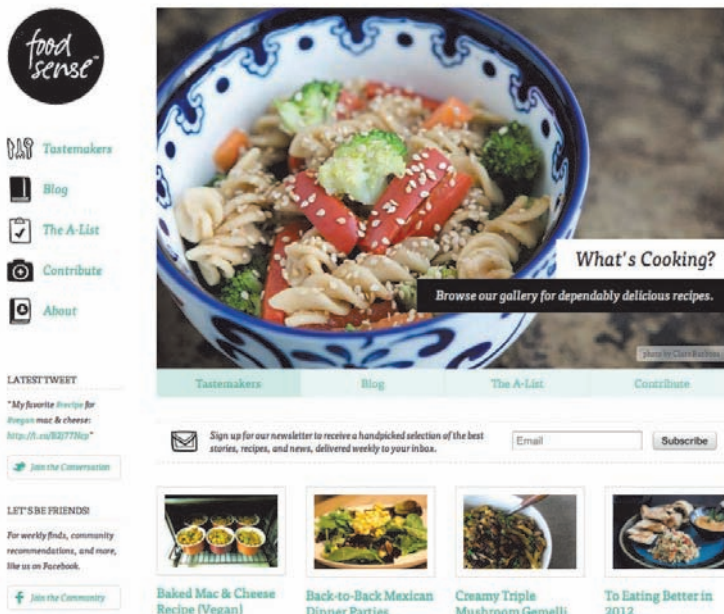
Gdy my albo nasza organizacja przechodzimy na nowe i lepsze sposoby tworzenia projektów¹²⁴, zrozumienie sposobów oddzielenia atmosfery od statycznych prezentacji wizualnych, a nawet istniejących stron WWW jest niezwykle istotne. To taka umiejętność, którą programiści oraz projektanci powinni zdobyć jak najszybciej, ponieważ wiedza na temat tego, w jaki sposób postrzegać, rozdzielać i zmieniać ułożenie elementów w formie układu przy każdej jego zmianie, jest kluczowa w dobrym responsywnym projekcie WWW.

WYDOBYĆ ATMOSFERĘ

Rzućmy okiem na niewielki wybór projektów WWW. Oddzielimy ich atmosferę i układ i przy każdym z nich zastanowimy się, co powoduje, że jego atmosfera jest unikalna.

Food Sense

Pierwszym przykładem jest Food Sense¹²⁵, przepięknie zaprojektowana, responsywna strona, której tematyką jest „żywność roślinna” (rysunek 11.5). Jeśli zastanowimy się nad nią w kryteriach statycznych, możemy skomentować dwukolumnowy układ desktopowy. Ale my chcemy spojrzeć głębiej. Szukamy jej atmosfery. Rozbijemy ją na kolor, teksturę i typografię.



Rysunek 11.5. Strona serwisu Food Sense

¹²⁴ W poprzednim rozdziale Stephen Hay opisuje tego typu praktyki projektowania z myślą o przyszłości.

¹²⁵ smashed.by/food.

Kolor: w atmosferze każdego projektu kolor wywołuje nastroje i kreuje emocje, ale słownika kolorów możemy też użyć do prowokowania działań, przekazywania komunikatów i innych interakcji. Food Sense wykorzystuje subtelne odcienie zieleni, czerni i bieli łączącej w jedno wszystkie elementy projektu.

Tekstura: gdy mówię o teksturze, nie mam na myśli dosłownego znaczenia tego terminu, choć nikt nie broni Ci zastosować tekstury skóry albo podartego papieru, jeśli lubisz takie rzeczy. Zamiast tego przez teksturę rozumiem aspekty dekoracyjne projektu, w tym linie (ramki i separatory), cienie i kształty.

Możemy na przykład przeanalizować linie. Czy są pojedyncze, podwójne, przerywane? Czy mają jednolitą szerokość? Czy istnieje hierarchia szerokości linii separatorów między głównymi i pomocniczymi sekcjami treści? Jeśli tak, w jaki sposób jest ona ustrukturalizowana?

Jak są wykorzystane cienie? Czy kontenery mają wyraziste tło, a może zastosowany jest gradient lub wzorek? Czy narożniki kontenerów są zaokrąglone, czy ostre? Te wszystkie zagadnienia odnoszą się do tekstury.

Food Sense ma jednolite kolory tła kontenerów i proste, przerywane oraz ciągłe linie separatorów. Przyciski i odnośniki mają subtelne, szare ramki i cienie. Obrazy wstawione w treści mają białe dopełnienie, co widać dzięki szarej ramce. Ręcznie rysowane ikony wzmacniają przyjacielską atmosferę projektu serwisu Food Sense.

Typografia jest prawdopodobnie tym z elementów atmosfery, który najłatwiej jest wyodrębnić, ale typografia to coś więcej niż wybór czcionki. Atmosfera typograficzna uwzględnia wysokości wierszy, odstępy między znakami oraz między elementami. Elliot Jay Stocks napisał¹²⁶:

Uważam, że rozpoczęcie projektowania od typografii zamiast od treści posuwa nas krok dalej od zbędnych komplikacji związanych z „projektowaniem dla samego projektowania” i krok bliżej do tego, byśmy zostali mistrzami typografii.

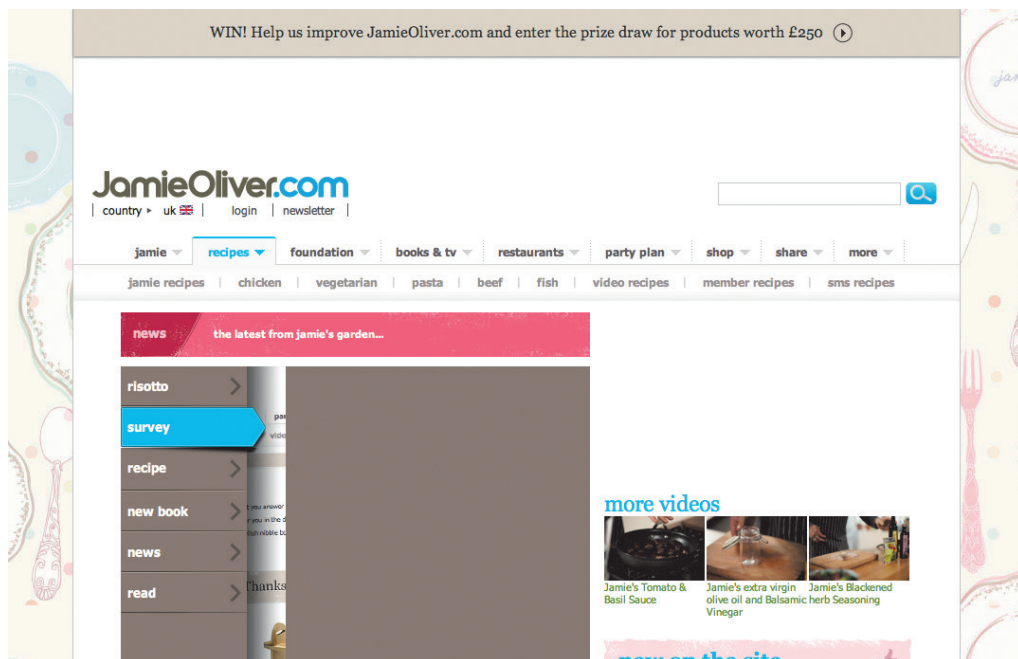
Projektanci Food Sense wybrali nagłówki, komunikaty i treści pisane małymi literami szeryfową czcionką FF Tisa Web Pro. Ich projekt dopuszcza dużą liczbę odstępów, co tworzy lekką, otwartą atmosferę pełną powietrza.

Jamie Oliver

Gdy dłuższy czas studiujemy atmosferę projektów, nasze oczy szybko przyzwyczajają się do ich różnic. Pozostając przy tematyce jedzenia, rzućmy okiem na stronę Jamiego Olivera¹²⁷. Ma ona zupełnie inny wygląd w porównaniu z Food Sense. Rozbijmy atmosferę strony Jamiego Olivera na elementy: kolor, teksturę i typografię. Zostawię Ci nieco pracy na samodzielnej analizie.

¹²⁶ Elliot Jay Stocks, *The Typography-out Approach in the World of Browser-based Web Design* (podejście do projektowania WWW oparte na typografii), smashed.by/typeout.

¹²⁷ smashed.by/jam.



Rysunek 11.6. Strona WWW Jamiego Olivera

Kolor: strona wykorzystuje połączenie brązów i błękitów, co jest widoczne również w logo. Czy uda Ci się znaleźć system regulujący zastosowanie kolorów w odnośnikach? W jaki sposób kolor jest użyty w nawigacji?

Tekstura: w jaki sposób strona wykorzystuje obrazy tła do tworzenia bloków treści? Czy zauważasz wzorce projektowe w tych panelach? Na przykład jak nagłówki są oddzielone od treści poniżej? W jaki sposób są zaprojektowane przyciski? Czy mają zaokrąglone, czy ostre narożniki? Czy zachowano tutaj spójność?

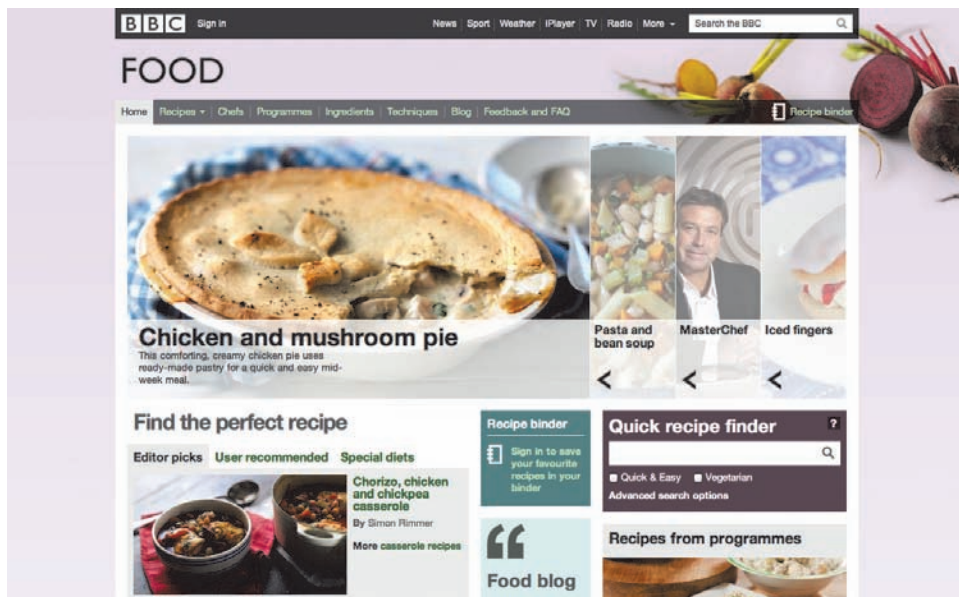
Typografia: strona wykorzystuje powszechnie dostępne czcionki. Czy zastosowano wyraźną hierarchię nagłówków? Czy istnieją różnice między typografią w treści? Jeśli tak, na czym polegają? Czy strona sprawia wrażenie otwartej, czy zatłoczonej? W jaki sposób dopełnienie i marginesy wpływają na poczucie przestrzeni w projekcie?

BBC Food

Do końca pozostaniemy przy tematyce żywieniowej. Przyjrzyjmy się stronie WWW poświęconej żywnieniu przygotowanej przez BBC¹²⁸, na której mamy okazję zaobserwować w działaniu zalecenia GEL dotyczące wyglądu. Ponownie nie będziemy zwracać uwagi na układ.

¹²⁸ smashed.by/bbc.

Zamiast tego będziemy szukać elementów atmosfery, które odróżniają tę stronę od innych stron o zbliżonej treści lub tematyce.



Rysunek 11.7. Strona BBC Food

Strona BBC Food wykorzystuje kolor do zasugerowania, które z elementów są odnośnikami. Zastosowana paleta jest przygaszona, być może w celu wyodrębnienia kolorów na fotografiach. W miejscach, w których są wykorzystane ikony z zestawu zdefiniowanego w dokumencie GEL, są one wyraziste i mają jednolite kolory, jak tła kontenerów.

Na stronie BBC krawędzie kontenerów są kwadratowe i ostre. W miejscach, w których treść nachodzi na obrazy, projektanci strony BBC Food zdecydowali się na półprzezroczyste tła w bieli lub czerni. Same obrazy są nieprzetworzone, bez krawędzi.

Strona BBC wykorzystuje pogrubioną typografię w celu „stworzenia silnie zarysowanych hierarchii i dramatyzmu na stronie”¹²⁹. Domyślną czcionką jest Arial, choć jeśli poszukasz głębiej, na platformie Mac znajdziesz i czcionkę Helvetica Neue. Te same czcionki są wykorzystane do nagłówków i treści.

Tego typu ćwiczenie można wykonać na dowolnej stronie WWW. Jest szczególnie przydatne w przypadku grupy projektantów lub programistów pracujących razem. Stało się kluczową częścią moich warsztatów poświęconych projektowaniu responsywnemu i uważam je za bardzo skuteczne.

¹²⁹ smashed.by/bbcfonts.

Stań się cudownie elastyczny

Niezależnie od tego, czy uczymy się projektować strony, zaczynając od komponentów, czy wydobywać atmosferę ze statycznych prezentacji wizualnych, nie musisz już tworzyć osobnych prezentacji statycznych dla każdej strony, nie mówiąc już o każdym urządzeniu. W rzeczywistości jestem skłonny zaryzykować stwierdzenie, że dni statycznych prezentacji projektu są policzone, ponieważ my, nasi szefowie i klienci, zaczynamy zdawać sobie sprawę z tego, jak są one mało precyzyjne i nieefektywne. A to oznacza, że musimy znaleźć nowe techniki projektowania responsywnego, czy nam się to podoba, czy nie.

Wiem z własnego doświadczenia z ostatnich dwunastu miesięcy, że im głębiej staram się sięgnąć w tematykę projektowania responsywnego, tym trudniej jest mi znaleźć odpowiednie narzędzia. Choć od lat promuję twierdzenie, że strony nie muszą wyglądać tak samo we wszystkich przeglądarkach, z jakiegoś powodu zrezygnowanie z kontroli nad układem nie przychodzi mi łatwo. Zorientowałem się, że moja rola jako projektanta uległa zasadniczej zmianie. Nie mogę już oczekiwać, że będę miał taką samą kontrolę nad układem na ekranach różnych rozmiarów, jaką cieszyłem się dawniej.

Responsywne projektowanie stron WWW niesie jednak istotne zalety dla projektantów, szczególnie w takiej organizacji pracy, gdzie ceni się elastyczność. Mogę teraz skupić się na szczegółach projektu na wszystkich etapach procesu implementacji, nie tylko na początku. Zamiast precyzyjnie definiować układy strony na każdym z możliwych rozmiarów ekranu, mogę wytłumaczyć swoje motywacje dla każdego elementu projektu i pozostawić innym członkom zespołu możliwość interpretacji projektu i adaptacji go w trakcie implementacji. Tego typu organizacja pracy działa doskonale dla mnie i moich klientów.

Jednak żeby to funkcjonowało prawidłowo, programiści, z którymi pracuję, muszą również nauczyć się podstaw projektowania oraz zdobyć umiejętności wydobywania atmosfery z projektu. Muszą wziąć odpowiedzialność za projekt i podejmować decyzje z nim związane. Dla niektórych to okazuje się przytłaczającym wyzwaniem, ale moi współpracownicy potraktowali to raczej jako możliwość rozwoju i nabycia nowych umiejętności.

Na początku tego rozdziału powiedziałem, że responsywne projektowanie stron WWW stawia więcej pytań, niż udziela odpowiedzi, że wystawia na próbę relacje w pracy i interakcje między wszystkimi osobami zaangażowanymi w proces. Wiem, że tym wyzwaniom nie zawsze łatwo sprostać. Jednak widziałem na własne oczy, jak zaangażowani projektanci i programiści stawiają im czoła, jak odpowiedzialni szefowie i klienci potrafią pozytywnie reagować, gdy orientują się, jak wiele korzyści może przynieść responsywny projekt strony. Dzięki wspólnej pracy możemy spowodować, że sieć stanie się bardziej responsywnym miejscem, jakim miała być od zawsze.



O autorze

Andy Clarke był nazywany wieloma określeniami, od kiedy rozpoczął projektowanie stron WWW dziesięć lat temu. Jego ego jest szczególnie połączane takimi jak: „ambasador CSS”, „branżowy prorok”, „inspirujący”, ale najbardziej dumny jest z tego, że Jeffrey Zeldman (ojciec chrzestny standardów WWW) raz nazwał go „potrójnie utalentowanym skubańcem”.

Andy zarządza Stuff and Nonsense, małym biurem projektowym, w którym współpracuje z takimi klientami jak ISO, STV oraz z rządem Wielkiej Brytanii. Prowadzi odczyty na konferencjach o projektowaniu WWW na całym świecie oraz jest autorem *Transcending CSS* i uznanego *Hardboiled Web Design*.

@font-face, 100
37signals, 182, 184
500 Startups, 241

A

ActionScript, 275
Adobe Flex, *Patrz:* Flex
Adobe Illustrator, *Patrz:* Illustrator
Adobe PhoneGap, *Patrz:* PhoneGap
Adobe Photoshop, *Patrz:* Photoshop
adres e-mail, 20, 62, 164
 no-reply, 187
AJAX, 137
Alfred Korzybski, 285
Allsopp John, 312, 317
amalgam of traits personality, *Patrz:* amalgamat osobowości śladowej
amalgamat osobowości śladowej, 231
Amazon Payments, 46
analiza
 heurystyczna, 22
 konkurencji, 22
 osobowości, 22
 przedprojektowa, 19, 20
 strategiczna, 22
Anderson Stephen, 241
Android, 199, 258
ankieta, 27, 28, 164
Anscas, 275
Anthony Wing Kosner, 241, 242
aplikacja
 aktualizacja, 265
 dedykowana, 272
 funkcjonalność, 287
 hybrydowa, 261, 280
 imersyjna, 275, 276
 międzyplatformowa, 272, 274, 277, 279, 285
 natywna, 33, 196, 199, 255, 256, 257, 263, 264, 268, 274, 275, 277, 278, 280
 nieimersyjna, 276, 277
 nienatywna, 260, 274
 projektowanie, 254, 268
 scenariusz, 299
 WWW, 33, 260, 261, 262, 264, 266, 268, 275, 278, 279
Apple, 192, 199, 258
Apple iCloud, 266, 267
Appleseed Jamie, 180
ASP, 13, 53
Ateş Faruk, 95
atrybut
 async, 77
 pubdate, 82
 role, 87
 typu, 76
Audible, 191

B

backend, *Patrz:* mechanizm strony serwerowej
Bagcheck, 179, 180, 181
Basecamp, 182
Baynard Institute, 180
baza danych, 61
 przeniesienie między hostami, 68
BB C Food, 329
Beanstalk, 67
Ben the Bodyguard, 176, 178
Berners-Lee Tim, 269
biblioteka
 HTML5 Shiv, 79
 jQuery, *Patrz:* jQuery
 Modernizr, *Patrz:* Modernizr

blog, 43, 80, 82
Bonlook, 183
Bootstrap, 322, 325
Budd Andy, 241
Burka Daniel, 18

C

cache, *Patrz:* dane podręczne
Carbonmade, 233
Carson Ryan, 187
CDN, 76
cel biznesowy, 11, 17, 19, 20
Chargify, 47
chmura, 64, 263, 266
Chrome, 72, 78, 96, 104, 107,
 ciasteczka, 88, 168
 cień, 100, 203, 211, 212
 client-side storage, *Patrz:* dane magazyn po stronie klienta
 CMS, 38, 40, 41, 42, 58
 Cocoa Touch, 259, 275, 277, 280
 Content Delivery Networks, *Patrz:* CDN
 continuous client, *Patrz:* klient ciągły
 conversion rate, *Patrz:* współczynnik powrotu użytkowników na stronę
 Cooper Alan, 231
 Corona, 275, 277
 cPanel, 64
 CSS, 13, 32, 72, 137, 223, 254
 klasa, 138
 przejście, *Patrz:* przejście
 selektor, *Patrz:* selektor skrypt resetujący, 79
 transformacje, 122, 124
 punkt odniesienia, 125
CSS3, 32, 94, 310
cutting-edge features, *Patrz:* przeglądarka nowe funkcje
 czcionka, 100, 101, 102, 219, *Patrz też:* typografia
 czynnik ekranu, 85

D

dane
 aktualizacja, 182
 magazyn po stronie klienta, 88, 297
 podręczne, 182
dark patterns, 190
deklaracja, DOCTYPE, 73, 75
delegacja zdarzeń, 140
Digg, 18
dokument
 projektowanie, *Patrz:* projektowanie dokumentów
 skupiony na treści, 268
DOM, 136
domena, 62, 68
Dribbble, 322
dziedziczenie, 86

E

e-book, 275, 277
e-commerce, 44, 45, 58, 59
efekt viralowy, 178
ekran
 kontrast, 198
 OLED, 221
 wielkość, 196, 197, 269, 285, 297, 313, 314, 315
electronic point-of-sale, *Patrz:* EPOS
element
 blokowy, 81, 84
 body, 98
 canvas, 153

em, 97
emocje, 238
EPOS, 52
etykieta, 173
ewangelista, 59

F

Facebook, 11, 16, 169, 181, 262
Feathers, 259
Firefox, 72, 96, 102, 122, 204107
Fireworks, 312, 315, 317
Flash, 262, 275
flash test, *Patrz:* test skojarzeń
Flex, 275
Flexbox, 107, 108, 111
Flexible Box Layout, *Patrz:* Flexbox
Food Sense, 327
forced continuity, *Patrz:* wymuszona kontynuacja
Ford Henry, 175
formularz, 142, 163, 173, 322, 324
akordeonowy, 170
 pytanie o pieć, 171
 walidacja, 143, 174
 wartości domyślne, 172

G

Gallagher Nicolas, 79
General Electric, 234
Git, 67
GitHub, 67
Gmail, 260, 264
Google Analytics, 75
Google Checkout, 46
Google Docs, 264
Google Web Fonts, 102
gra, 275, 277
 OpenGL, 223
gradient, 115, 201, 211, 214, 215, 216
graf punktów przelomowych, 300, 301
Groupon, 179
Grupa Robocza ds. Technologii Hipertekstowych
 Aplikacji Sieciowych, *Patrz:* WHATWG

H

HAML, 254
Headscape, 12
Heppo.se, 183
Hicks Jon, 314
Hipmunk, 236, 242
Holst Christian, 180
host
 konfiguracja, 67
 zmiana, 67
hosting, 38, 53, 60, 61, 62
 bezawaryjność, 63
 w chmurze, 64
 współdzielony, 64
Hsieh Tony, 184
HTML, 32, 137, 254
HTML5, 32, 72, 73, 90
HTML5 Element Flowchart, 85
Human Interface Guidelines, *Patrz:* interfejs użytkownika zalecenia
Hume Andy, 310

I

iframe, *Patrz:* ramka
Illustrator, 207, 208
 Zapisz dla Internetu, 208
 zarządzanie kolorem, 208

interakcja, 254
 interfejs użytkownika, 12, 14, 28, 184, 198
 mobilny, 181
 zalecenia, 258
 Internet Explorer, 79, 90
 iOS, 199
 iPhone, 197, 199, 202, 203, 219, 259, 260, 261, 275,
 278, 314, 316
 Irish Paul, 95
 Ive Jonathan, 192

J

Jade, 254
 Jamie Oliver, 328
 JavaScript, 32, 72, 76, 136, 137, 254, 277, 299
 jednostka
 em, *Patrz:* em
 rem, *Patrz:* rem
 Jobs Steve, 175, 192, 314
 Jos Buivenga, 101
 jQuery, 136, 140, 254
 JSON, 77

K

kafelek stylów, 321
 Kate Kiefer-Lee, 249
 Keith Jeremy, 263, 314
 klawiatura Swype, 259
 klient ciągły, 266, 267
 Klout, 179
 kompatybilność wsteczna, 73
 komunikator Trillian, 266
 konkurencja, 22

L

LABjs, 76
 LDAP, 52
 LESS, 254, 305
 lista, 141, 177, 180, 322
 ikona, 183
 LocalStorage, 88, 89, 90, 145, 146
 logo, 14

M

MailChimp, 186, 239, 245, 246, 325
 mapa bitowa, 201, 209
 Marcotte Ethan, 310
 marka, 14, 243
 McClure Dave, 241, 242
 mechanizm
 3-D Secure, 48
 antyspamowy, 62
 API JavaScriptu, 88
 blogowy, 43
 Drag and Drop, 153
 FileReader, 153
 Local Storage, *Patrz:* Local Storage
 logowania, 181
 MasterCard SecureCode, 48
 sessionStorage, 88, 90
 sklepu internetowego, 38, 44, 49, 50
 strony serwerowej, 38, 55
 śledzenia zamówień, 50
 uwierzytelniający, 181
 Verified by Visa, 48
 zarządzania kolorem, 204, 205, 208
 media query, 310
 Medina John, 238
 menu rozwijane, 83
 merchant account, *Patrz:* użytkownik konto
 dedykowane w systemie płatności

metainformacje, 87
 metoda, 138
 className, 138
 querySelector, 138
 querySelectorAll, 138
 Meyer Eric, 79
 miernik kompletności, 178
 miniatura, 153
 Modernizr, 79, 95
 Moll Cameron, 11, 12, 14
 Monotask, 164
 mood board, *Patrz:* szablon nastrojów
 MooTools, 254
 Mullenweg Mat, 163

N

nagłówek, 80, 81, 84, 85, 86, 99, 322
 natywność, *Patrz:* aplikacja natywna
 Neal Jonathan, 79
 Nerd Communications, 176
 Nielsen Jakob, 12
 non-core, *Patrz:* technologia poboczna

O

open source, 55
 Opera, 72, 73, 74, 96, 108, 154
 opowieść, 176
 organizacja
 logo, *Patrz:* logo
 marka, *Patrz:* marka
 rynek, 14
 wizerunek, 14
 zarząd, 20
 osobowość, 231, 232, 235, 236, 241, 243
 tworzenie, 245
 Oto Mark, 322

P

pamięć długotrwała, 238
 parser
 JSON, 77
 XML, 76
 Payment Card Industry Data Security Standard,
 Patrz: PCI DSS
 payment gateway, *Patrz:* płatności bramka
 payment service provider, *Patrz:* płatności
 bramka
 PayPal, 44, 46
 PCI DSS, 47, 49
 persona, 231, 242, 244
 Petrus Piotr, 85
 PhoneGap, 275, 277
 Photojojo, 233
 Photoshop, 196, 302, 317
 cienie, 203, 211, 212
 czcionka, 219
 gradient, 201, 211, 214, 215, 216
 maski wektorowe, 201, 217
 obiekt inteligentny, 217, 218
 plasterek, 223, 224
 oparty na warstwie, 225
 profil koloru, 204, 206, 207
 roztrząsanie, 216
 skalowanie, 203
 tekstura, 217
 warstwa
 efekty, 201
 kształtu, 209
 obracanie, 209
 wzorek, 217

Zapisz dla Internetu, 207
 zarządzanie kolorem, 204, 205
 PHP, 61
 PHPMyAdmin, 68
 piksel, 196
 gestość, 197, 286, 297
 Pinker Steven, 235
 Plesk, 64
 plik
 PDF, 201
 PNG, 200, 204, 220
 kompresja, 220
 SVG, 201, 297
 płatności, 38, 46
 bramka, 46, 47, 53
 kartą kredytową, 47, 173, 174
 okresowe, 49
 poczta e-mail, 62, 68
 odpowiedzi od automatów, 187
 podatek, 49
 poddomena, 64, 76
 podpiksel, 220, 221
 podręcznik stylu, 249, 324
 projektowanie
 aplikacji, *Patrz:* aplikacja projektowanie
 dokumentów, 254, 268
 interakcji, *Patrz:* interakcja
 komponenty, 319, 320, 322, 325, 331
 liniowe, 294, 296, 303
 od zewnątrz do wewnątrz, 255
 responsywne, 25, 33, 43, 310, 315, 331
 style, 324
 user experience, *Patrz:* UX
 w przeglądarce, 302, 303, 304, 305, 306
 zorientowane na użytkownika, 231, 255, 281
 przeglądarka, 25, 32, 72
 Firefox, *Patrz:* Firefox
 funkcja, 74, 75, 95, 137
 Google Chrome, *Patrz:* Chrome
 Internet Explorer, *Patrz:* Internet Explorer
 nowe funkcje, 72
 przedrostek wydawcy, 95, 96
 Safari, *Patrz:* Safari
 wersja beta, 72
 przejście, 129, 130, 138, 139
 przepływ, 285
 przycisk, 173, 176, 182, 324
 pseudoklasa, 127
 punkt przełomowy, 300, 301

Q

QuietWrite, 168, 169

R

ramka, 76, 100
 raport analityczny, 22
 Recurly, 47
 reguła, 94
 reguła, 102
 rem, 97
 RequireJS, 76
 Retina, 197, 198, 199, 202, 219
 Rieger Bryan, 295
 rozdzielczość, 13, 33, 197, 199, 286

S

SaaS, 46, 58, 59
 Safari, 72, 78, 96, 98, 104, 154
 Santa Maria Jason, 313
 screen readers, *Patrz:* czytnik ekranu

selektor, 94, 125, 142
kraj, 180

serwer, 38, 54
dedykowany, 64
DNS, 68
produkcyjny, 65
przedprodukcyjny, *Patrz:* serwer testowy
testowy, 65
VPS, *Patrz:* serwer wirtualny
wirtualny, 64

sessionStorage, 88, 90

sklep
internetowy, 44, 49
dostawa, 50
koszyk, 45, 58
produkty cyfrowe, 51
promocje, 45
z aplikacjami, 33, 264

skrypt, 76
Normalize.css, 79, 80
Reset CSS, 79
resetujący CSS, 79
wielkość, 78

Softpedia, 190

software as a service, *Patrz:* SaaS

Souders Steve, 78

Spent, 177, 178

sprite, 223

Square, 174, 175

staging, *Patrz:* serwer testowy
standardy
WAI-ARIA, *Patrz:* WAI-ARIA
WWW, 32, 94

storytelling, *Patrz:* opowieść

Stripe.com, 46, 47, 169

strona WWW, 268, 269, 270
aktualizacja w czasie rzeczywistym, 179, 180
analiza, *Patrz:* analiza
animacja, 129, 131, 147, 198
budżet, 53
dostępność, 85, 86
funkcjonalność, 287
integracja z innymi systemami, 51, 54
język programowania, 53, 56
kod, 13
moda, 16
modyfikowanie, 11, 12, 14, 55, 57
modyfikowanie pułapki, 15, 53
na urządzenia mobilne, *Patrz:* urządzenie
mobilne
odwiedzalność, 14
oglądalność, 44
osobowość, *Patrz:* osobowość
pozycjonowanie elementów, 106
prototyp, 303, 304
przejście, *Patrz:* przejście
przenoszenie między hostami, 67, 68
przeprojektowanie, 10, 12, 31, 38, 55, 65,
230, 286
czynniki, 13, 14
oparte na współpracy, 23
pułapki, 15, 53
ranking, 60
refaktoring, *Patrz:* strona WWW
przeprojektowanie
rytm, 98
sklep internetowy, *Patrz:* sklep internetowy
statyczna, 43
struktura, 84
struktura wewnętrzna, 13
szkielet, 24, 289, 291, 294
niezależny od treści, 290
szybkość ładowania, 78

testowanie, 26, 27

tło, 112, 117, 119, 128, 326
wielokrotne, 113

typografia, *Patrz:* typografia
układ, 319, 320
elastyczny blokowy, *Patrz:* Flexbox
wielokolumnowy, 105
użytkownik, *Patrz:* użytkownik
wersja demonstracyjna, 64
wydajność, 13, 78
zwrot z inwestycji, 17, 27, 34

Studholme Ola, 85

style tile, *Patrz:* kafelek stylów

Stylus, 254

Subversion, 67

symbol Unicode, 259

system
EPOS, *Patrz:* EPOS
iOS, 258
kontroli wersji, 38, 66, 67
zarządzania treścią, *Patrz:* CMS
szablon nastrojów, 23, 24, 319, 320

T

tablica nastrojów, *Patrz:* szablon nastrojów
technologia poboczna, 72

test
A/B, 44
porównawczy, 44
reakcji emocjonalnych, 28
skojarzeń, 28
użyteczności, 28

The Web Hypertext Application Technology
Working Group, *Patrz:* WHATWG

Think Vitamin, 187

Titanium Mobile, 277, 279, 280

Topolsky Joshua, 266

Tomnton Jacob, 322

Trillian, 266

Twitter, 62, 89, 167, 240, 322

TypeKit, 102

typografia, 97, 104, 296, 326, 328, 329, 330

U

UglifyJS, 78

Union Square Ventures, 241

Unity, 275, 277

Urban Ministries of Durham, 177

urządzenie
klasa, 298, 301
mobilne, 32, 196, 220, 270, 271, 294, 296, 297,
298

usability test, *Patrz:* test użyteczności

user experience, *Patrz:* UX

user experience design, *Patrz:* UX

usługa SaaS, *Patrz:* SaaS

UX, 46, 162, 167, 184, 186, 241, 242, 254, 255, 259,
266, 270, 296
antywzorce, 190, 191
dokumentacja, 186
udoskonalenia, 299, 301
użytkownik, 22, 26, 39, 164, 311
doświadczenie, *Patrz:* UX
hasło, 46, 181
konto, 46
dedykowane w systemie płatności, 46, 47
kupieckie, *Patrz:* użytkownik konto
dedykowane w systemie płatności
logowanie, 168, 170
progresywne, 181

polecenia, 188
poufność, 181
rekompensata, 187
testy, 28, 29, 30
zaangażowanie, 188, 231, 242, 244

V

Validator.nu, 85
vendor prefixed, *Patrz:* przeglądarka
przedrostek wydawcy
Verify App, 28
virtual private server, *Patrz:* serwer wirtualny

W

W3C, 73
WAI-ARIA, 85, 86, 90
Walton Trent, 320
Warren Samantha, 321
Web Accessibility Initiative: Accessible Rich
Internet Applications, *Patrz:* WAI-ARIA
Web Host Manager, 64
WebSockets, 179
WebSort, 28
WHATWG, 73
Wilson Fred, 241, 242
Wired, 191
wireframe, *Patrz:* strona WWW szkielet
WordPress, 43, 60, 61, 163
World Wide Web Consortium, *Patrz:* W3C
WorldPay, 48
Wroblewski Luke, 294
współczynnik powrotu użytkowników
na stronę, 14
Wuffo, 190, 240, 242
wymuszona kontynuacja, 191
wyszukiwarka, 60

X

XML, 76

Y

Yepnope, 76, 77

Z

Zappos, 184
zasada uniwersalności, 269
zmiany przyrostowe, 13, 17
znacznik
article, 80, 82, 84, 87
aside, 82
div, 83
figure, 83
footer, 82, 87
header, 81, 87
META, 75
nav, 83
odwołań
do arkuszy stylów, 76
do skryptów, 76
otwarty, 75
samozamykający, 75
script, 76, 77
section, 80, 82, 84
strong, 79
time, 82
tworzenie, 79
ul, 81

PROGRAM PARTNERSKI

GRUPY WYDAWNICZEJ HELION



- 1. ZAREJESTRUJ SIĘ**
- 2. PREZENTUJ KSIĄZKI**
- 3. ZBIERAJ PROWIZJĘ**

Zmień swoją stronę WWW
w działający bankomat!

Dowiedz się więcej i dołącz już dzisiaj!

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

GRUPA WYDAWNICZA

 **Helion SA**

THE SMASHING BOOK #3 EDYCJA POLSKA

TA KSIĄŻKA ZMIENI TWOJE PODEJŚCIE DO PROJEKTOWANIA STRON WWW!

Sieć WWW podlega ciągłym przemianom. Webdeveloperzy dysponują lepszymi narzędziami, a przeglądarki są wydajniejsze i potrafią znacznie więcej niż jeszcze kilka lat temu. Te zmiany mają ogromny wpływ na to, jak obecnie tworzy się atrakcyjne strony internetowe. A czy Twoje podejście do projektowania stron idzie z duchem czasu? Czy Twoje rozwiązania prawidłowo funkcjonują na urządzeniach mobilnych, które generują coraz więcej ruchu sieciowego? Jeśli masz wątpliwości, koniecznie sięgnij po tę książkę — kolejny podręcznik z kultowej serii **Smashing Magazine!**

Znajdziesz tu odpowiedzi na wiele kluczowych pytań. W trakcie lektury dowiesz się, jak do projektowania współczesnych stron internetowych podchodzą światowej sławy profesjonaliści. Poznasz najlepsze techniki wykorzystania możliwości HTML5, CSS3 i JavaScriptu. Nauczysz się również tworzyć strony zgodne z podejściem responsive web design. Te trzy słowa to klucz do Twojego sukcesu! Przekonaj się sam dzięki tej niesamowitej książce!

Sięgnij po najnowszą książkę z serii **Smashing Magazine** i:

- odkryj możliwości nowoczesnej strony WWW
- zbuduj atrakcyjny układ wizualny witryny
- zadbaj o dostępność projektowanej strony na urządzeniach mobilnych
- poznaj najnowsze osiągnięcia profesjonalnych webdeveloperów

helion.pl
księgarnia
internetowa

Nr katalogowy: 17374



Księgarnia internetowa:
<http://helion.pl>



Zamówienia telefoniczne:
0 801 339900



0 601 339900



Helion

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

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

sięgnij po WIĘCEJ



KOD KORZYŚCI

ISBN 978-83-246-5613-4



9 788324 656134

cena: 59,00 zł

Informatyka w najlepszym wydaniu