

Spis treści (skrócony)

Wprowadzenie	xxi
1. Wprowadzenie do mobilnych technologii webowych. <i>Wrażliwe projekty, czyli Responsive Web Design</i>	1
2. RWD na poważnie. <i>Koncepcja Mobile First w podejściu Responsive Web Design</i>	43
3. Oddzielna witryna mobilna. <i>Stawiamy czoła niezupełnie sprzyjającym okolicznościom</i>	91
4. Komu wsparcie, komu? <i>Które urządzenia powinny być obsługiwane?</i>	137
5. Bazy i klasy urządzeń. <i>Zapoznaj się z grupą</i>	151
6. Framework dla mobilnych aplikacji internetowych. <i>Tartanator</i>	217
7. Mobilne aplikacje w prawdziwym świecie. <i>Wyjątkowe mobilne aplikacje internetowe</i>	267
8. Tworzenie hybrydowych aplikacji mobilnych z PhoneGap. <i>Ustrzel tartan! — w stronę natywności</i>	313
9. Podejście „future friendly”. <i>Odnajdywanie (jakiegoś) sensu w chaosie</i>	357
A Ścinki. <i>Sześć najważniejszych spraw (o których nie mówiliśmy)</i>	373
B Postaw swój serwer. <i>Gdzieś trzeba zacząć</i>	387
C Instalowanie WURFL. <i>Jak wywyższyć urządzenia?</i>	397
D Instalowanie SDK i narzędzi dla Androida. <i>Zadbaj o środowisko</i>	403
Skorowidz	417

Spis treści (z prawdziwego zdarzenia)

W

Wprowadzenie

Twój mózg kontra technologie mobilne. Starasz się czegoś nauczyć, a mózg robi Ci „przysługę”, za wszelką cenę odciągając Twoją uwagę od nauki. Myśli: „Lepiej wyjdź i zajmij się czymś ciekawszym — wypatruj krwiożerczych bestii albo sprawdź, czy kiedy podpalisz swoje BlackBerry Bold, włączy się alarm pożarowy”. Jak w takim razie oszukać mózg, by uznał, że Twoje życie zależy od znajomości technologii mobilnych?

Dla kogo jest ta książka?	xxii
Wiemy, co sobie myślisz	xxiii
Wiemy też, co sobie myśli Twój mózg	xxiii
Metapoznanie — myślenie o myśleniu	xxv
Zespół korektorów merytorycznych	xxx
Podziękowania	xxxi

Wprowadzenie do mobilnych technologii webowych

1 Wrażliwe projekty, czyli Responsive Web Design

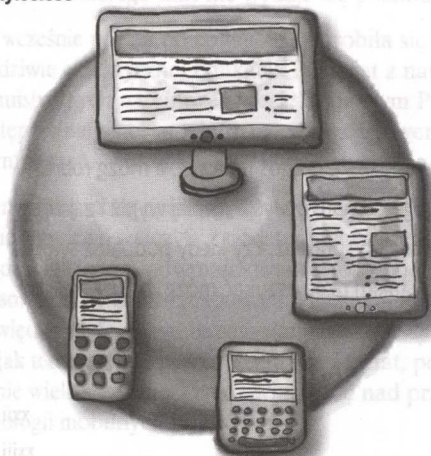
Witajcie! Jesteście gotowi na mobilne technologie webowe? Tworzenie witryn na urządzenia mobilne jest naprawdę ekscytujące. Wiele w tym uroku, emocji i momentów, w których chciałoby się wykrzyknąć: **Eureka!** Ale z drugiej strony pełno tu tajemnic i trudności. Technologie mobilne rozwijają się w tak niewiarygodnym tempie, że cały czas jesteśmy trochę w tyle. Trzymaj się więc mocno! Naszą przygodę rozpoczynamy od ciekawego podejścia do tworzenia witryn internetowych, znanego jako **Responsive Web Design (RWD)**. Dzięki niemu będziesz mógł sprawić, by strony wyglądały równie dobrze na wielu różnych urządzeniach mobilnych i, co ważne, przydadzą Ci się umiejętności, które już masz.

Wszyscy jedziemy na tym samym wózku. Wskakujesz?	2
Coś niedobrego stało się w drodze do pubu	4
Skoro przeglądarki w telefonach komórkowych są takie świetne...	5
...to czy nie powinno to po prostu działać?	5
Wrażliwe projekty — Responsive Web Design	10
Różne arkusze stylów w różnych sytuacjach	12
Zapytania o media w CSS	13
Dotychczasowa struktura witryny pubu Pod Paradnym Morsem	15
Analiza dotychczasowego arkusza CSS	16
Co trzeba zmienić?	17
Szukamy stylów wymagających zmiany	18
Droga do stylów dostosowanych do urządzeń mobilnych	19
Co jest nie tak z układami o stałej szerokości?	26
Dlaczego płynne jest lepsze?	27
Wzór płynności	28
Ciąg dalszy przekształceń	29
Przełączanie kontekstu	31
Co się stało z tymi obrazami?	32
Płynne obrazy	33
Pamiętaj, by być wrażliwym	36
Oto strona w stylu RWD!	40
Podejście Responsive Web Design to również stan umysłu	41

index.html



styles.css



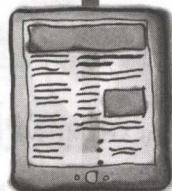
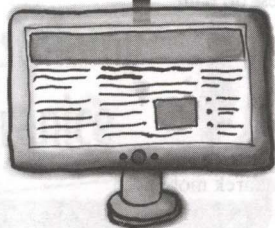
RWD na poważnie

Koncepcja Mobile First w podejściu Responsive Web Design

2

Oto śliczna mobilna witryna. Ale nie oceniaj jej tylko po pozorach. Pod tą piękną powłoką znajdziesz bowiem coś zupełnie innego. Być może wygląda jak mobilna witryna, ale to wciąż zwykła, desktopowa witryna, z tym że przebrana w mobilne ciuszki. Jeśli chcesz, żeby na urządzeniach mobilnych chodziła jak dobrze naoliwiona maszyna, musisz zastosować zasadę **Mobile First**. Jednak najpierw musimy przeprowadzić sekcję obecnej witryny, by odnaleźć ukrywający się w jej wnętrzu desktopowy szkielet. Następnie gruntownie posprzątamy i zaczniemy pracować na świeżo, zgodnie ze strategią **stopniowego ulepszania**, zaczynając od budowania podstawowych elementów, a kończąc na bogatej wersji desktopowej. Gdy skończymy, nasza strona będzie zoptymalizowana pod każdą możliwą rozdzielczość ekranu.

Stopniowe ulepszanie bazujące na rozmiarze ekranu oraz możliwościach klienta

Bardzo małe
ekrany (telefony)Małe ekrany
(smartfony)Średnie ekrany
(tablet)Większe ekrany
(komputery
i telewizory)

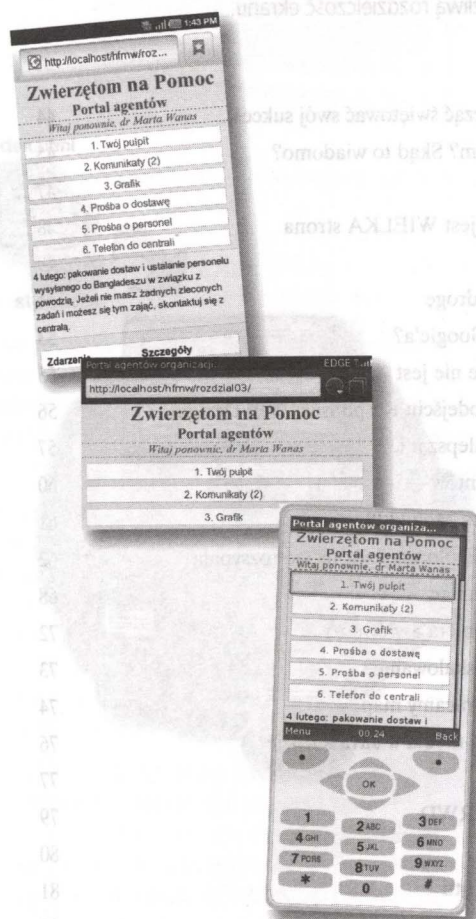
Gdy właśnie zamierzałeś zacząć świętować swój sukces...	44
Czy to naprawdę jest problem? Skąd to wiadomo?	45
Co zrobić, gdy nie śmiga?	47
Nie ma co się oszukiwać, to jest WIELKA strona	48
Dobrodziejstwa pliku HAR	49
Wyteż wzrok i znajdź zawalidrogę	51
Skąd pochodzi skrypt map Google'a?	53
Wygląda przyjaźnie, ale takie nie jest	55
Koncepcja Mobile First w podejściu Responsive Web Design	56
Na czym polega stopniowe ulepszanie?	57
Poprawiamy pływanie elementów	60
Zapytania o media w technice Mobile First	61
Niespodzianka! W Internet Explorerze strona się rozsyłała	62
Problemy z jednym atrybutem src	68
Powiększanie w znaczniku <meta> viewport	72
Czy powinno się umożliwiać skalowanie?	73
Z pomocą JavaScriptu przywracamy mapę	74
Budujemy pseudozapytanie o media w JavaScriptcie	76
Wstawiamy skrypt na stronę	77
Ten widżet nie jest zgodny z RWD	79
Przenosimy atrybuty do CSS	80
Usuujemy atrybuty z JavaScriptu	81
Mapa znów zasłania treść strony	83
Niech prowadzi Cię zawartość strony	84
Wartości graniczne przybywają na ratunek	87

Oddzielna witryna mobilna

Stawiamy czoła niezupełnie sprzyjającym okolicznościom

3

Wizja jednego, wrażliwego projektu witryny jest cudowna... Mamy tylko jeden układ strony opracowany w zgodzie z koncepcją Mobile First, który dopasowuje się do specyfiki różnych przeglądarek i urządzeń. Brzmi pięknie. Co się jednak stanie, gdy tę wizję przyprawimy choćby szczyptą realizmu? Niezaktualizowane systemy, stare urządzenia lub ograniczenia budżetu klienta to tylko kilka przykładów. A co, jeśli zamiast łączyć wersję desktopową z mobilną, chciałbyś je rozdzielić? W tym rozdziale przyjrzymy się szczegółowo **wykrywaniu użytkowników korzystających z urządzeń mobilnych, wspieraniu starszych telefonów i tworzeniu odrębnych witryn dla urządzeń mobilnych.**



Agenci organizacji Zwierzętom na Pomoc patrolują pola	92
Jak agenci mogą otrzymywać i przekazywać informacje?	93
Wysyłamy mobilnych użytkowników na zoptymalizowaną witrynę	96
Jak wywęszyc mobilnych użytkowników?	97
Rozpoznajemy agenta użytkownika	98
Łańcuch user-agent — dzieło szatana?	101
Mówiąc wprost — większość dużych witryn ma swoją wersję mobilną	104
Kiedy jedynym rozwiązaniem jest przekierowanie...	105
Rzuć okiem na skrypt	106
Jak działa skrypt?	107
Przygotowujemy makietę wersji mobilnej	108
Specjalna dostawa... spraw komplikujących życie	110
Nie wszystkie telefony to smartfony...	113
Prostota przede wszystkim — poznaj XHTML-MP	114
Dlaczego chcemy użyć tak starego rozwiązania?	115
XHTML-MP pomaga unikać problemów	116
Przy okazji — przewijanie jest do bani	119
Ostatni problem	119
Klawisze dostępu w akcji	123
Co poszło nie tak?	124
Naprawiamy błędy	125
CSS dostosowany do przeglądarek mobilnych	127
Hm... czegoś tu brakuje	132
Bardzo nam brakuje tych przycisków!	133
Wielki sukces!	134

Komu wsparcie, komu?

Które urzędnicy powinny być obsługiwane?

4

Przetestowanie witryny na wszystkich urządzeniach trwałoby wieczność.

Musisz ustalić granicę wyodrębniając urządzenia, które zamierzasz wspierać. **Ale jak podjąć tę decyzję?** Co z użytkownikami korzystającymi z telefonów, które już dawno powinny trafić na złomowisko? W jaki sposób stworzyć witrynę, by działała na urządzeniach, o których w ogóle nie słyszałeś? W tym rozdziale przygotujemy magiczną miksturę złożoną z **wymagań projektowych i informacji o odbiorcach**. Pomoże nam ona zdecydować, **które urządzenia mamy wspierać i co zrobić z tymi, których nie wspieramy**.

Skąd wiedzieć, gdzie ustalić granicę?	138
Odejdź na chwilę od komputera	139
Urządzenia, których <i>nie wspierasz</i> , kontra te, których <i>nie możesz wspierać</i>	140
Zadawaj dużo pytań o swój projekt	142
Składniki magicznej mobilnej mikstury	144
Zajrzyj do szafki z narzędziami i danymi	145
Skąd mam wiedzieć, czy moi klienci używają odpowiednich urządzeń?	150



Bazy i klasy urządzeń

Zapoznaj się z grupą

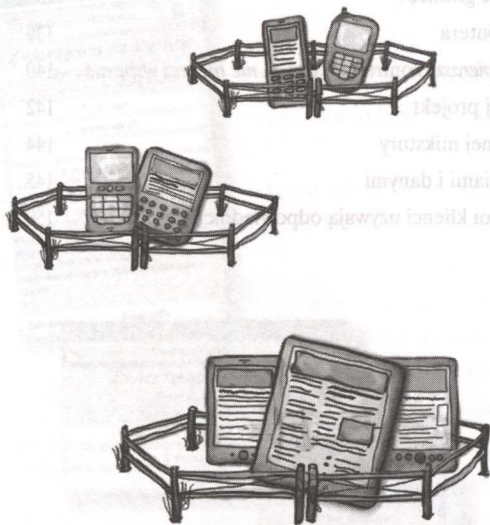
5

Podczas wybierania wspieranych urządzeń nie wzięliśmy pod uwagę kilku dokuczliwych problemów. Jak mamy się dowiedzieć wystarczająco dużo o mobilnych

przeglądarkach użytkowników, by przed dostarczeniem treści spełnić ich oczekiwania? Jak uniknąć tworzenia tylko podstawowej zawartości odpowiadającej najmnijemu wspólnemu

mianownikowi urządzeń? No i jak, pozostając przy zdrowych zmysłach, zapanować nad tym

wszystkim? W tym rozdziale wkroczymy w świat **możliwości urządzeń** i nauczymy się korzystać z **baz danych urządzeń**, by wreszcie odkryć, jak te wszystkie urządzenia grupować w **klasy**.



Przycisk awaryjny dla spanikowanych studentów	152
Źródła danych o urządzeniach mobilnych spieszą na ratunek	154
Poznaj WURFL	155
WURFL i jego możliwości	156
WURFL — sprytny interfejs API	159
My też możemy zbudować eksplorator	160
Eksplorator — przygotowanie środowiska	161
Szast-prast i eksplorator ulepszony	168
Czas zaprząć możliwości do pracy	170
Korzystamy z WURFL do różnicowania zawartości stron	170
Inicjalizacja obiektu urządzenia i przygotowanie danych	172
Czy to jest urządzenie mobilne?	172
Dzięki WURFL strona staje się sprytniejsza	176
Przycisk awaryjny — tylko w telefonach	177
Klasy urządzeń	181
Kolejny lukratywny kontrakt z firmą DaRadę!	182
Jak strona główna wygląda przez mobilne okulary?	183
Zdefiniowanie odmian witryny w zależności od wymagań	184
Przybliżamy klasy urządzeń	185
Zapoznajemy się z funkcją dopasowującą	191
O co chodzi w tej instrukcji switch?	192
Używamy funkcji dopasowującej do testowania możliwości	193
Wypełniamy luki w testach klas urządzeń	200
Musimy bardziej zadbać o bezpieczeństwo	211
Lepiej zapobiegać, niż leczyć	212

Framework dla mobilnych aplikacji internetowych

Tartanator

6

„My chcemy aplikację!”. Jeszcze rok czy dwa lata temu tego typu hasło wiązało się nieodłącznie z jednym — tworzeniem natywnych aplikacji dla każdego urządzenia, które zamierzaliśmy wesprzeć. Na szczęście teraz nie jest to jedyne możliwe rozwiązanie. Aplikacje internetowe dla mobilnych przeglądarek są coraz doskonalsze, zwłaszcza ostatnio, kiedy wkroczył **HTML5** wraz z nieodłącznymi kompanami — **CSS3** i **JavaScriptem**. W świat mobilnych aplikacji internetowych wejźmy wraz z **mobilnym frameworkiem**, czyli zbiorem gotowych rozwiązań programistycznych upraszczających i przyspieszających tworzenie aplikacji.

Hm... całkiem łatwe, ale czy moglibyście nad tym jeszcze popracować, by zachowywało się jak prawdziwa natywna aplikacja?



HTML5, aplikacja internetowa... Co te słowa znaczą?	219
Jak się zachowują klasyczne witryny internetowe?	220
Jak się zachowują witryny przypominające aplikacje?	221
Plan pierwszej fazy projektu Tartanator	224
Po co używać frameworków?	225
Dla projektu Tartanator wybraliśmy framework jQuery Mobile	226
Tworzenie prostej strony z jQuery Mobile	228
Kod pozostałych elementów strony	229
Atrybuty data-*	231
Odsyłacze do wielu stron w jQuery Mobile	234
Sedno Tartanatora — tartany jako takie	240
Wrzucamy pozostałe tartany	243
Filtrowanie i porządkowanie listy	244
Dodajemy pasek narzędzi w stopce	249
Upiększamy pasek narzędzi	250
Finalizowanie pracy nad strukturą	251
Czas na przygotowanie formularza do tworzenia tartanów	253
Tłumaczymy wzory tartanów na formularz	255
Tworzymy formularz w HTML5	256
Dodajemy podstawowe pola	257
Dodawanie kolorów umożliwiając użytkownikom listy w listach	258
Pary kolor – rozmiar: pole wyboru koloru	259
Pary kolor – rozmiar: pole rozmiaru	260
Odsyłacz do formularza	262



Mobilne aplikacje w prawdziwym świecie

7

Wyjątkowe mobilne aplikacje internetowe

Mobilne aplikacje są jak dzieci w klasie. Wiesz, chodzi o tę mieszankę fascynacji, przekonania o tym, że można zrobić niesamowite rzeczy, ale też tajemniczości i niepohamowanego rozrabiactwa. Staramy się trzymać pod kontrolą tych nadpobudliwych geniuszy, ustalając granice, a także dbając, by ich nie przekraczali. Przychodzi jednak czas zbierania korzyści z naturalnych talentów mobilnych aplikacji internetowych. Możemy zastosować metodę **stopniowego ulepszania**, by dopieścić interfejs na potrzeby lepiej rozwiniętych przeglądarek, a problemy z ciągłością dostępu do internetu rozwiązać za pomocą **trybu offline**. Możemy też wydobyć esencję mobilności, korzystając z **geolokalizacji**. Nie ma czasu do stracenia, zabierzmy się za tworzenie wyjątkowych mobilnych aplikacji internetowych!



Wygląda niezłe...	268
Aplikacje mobilne w prawdziwym świecie	270
Na miejsca, gotowi, ulepszać!	274
Ulepszamy formularz	275
Widżet zarządzający listą kolorów i rozmiarów	276
Zaglądamy pod przykrywkę	277
No tak, to było ulepszanie frontendu	278
...a teraz pora na backend	280
Dwie twarze skryptu generate.php	281
Jeszcze tylko jedno...	282
Tryb offline to ważna sprawa	284
Prosty przepis na plik manifestu	285
Jak zwykle diabeł tkwi w szczegółach	286
Udostępniaj pliki manifestu z prawidłowym nagłówkiem content-type	287
Zwyciężyliśmy (w końcu)	297
Jak działa geolokalizacja?	298
Jak poprosić przeglądarkę o dane geolokalizacyjne?	299
Początek pracy nad stroną Znajdź wydarzenie — podstawy	301
Dołączamy geolokalizację	303
Nic nie znalazł	309

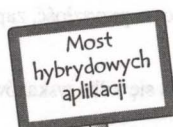
Tworzenie hybrydowych aplikacji mobilnych z PhoneGap

Ustrzel tartan! — w stronę natywności

8

Czasem musisz się zdecydować na aplikację natywną. Być może potrzebujesz dostępu do czegoś, co nie jest (jeszcze) osiągalne z poziomu przeglądarki. A może klient chce, by aplikacja *koniecznie* znalazła się w sklepie. Nie możemy się już doczekać chwili, gdy przeglądarka będzie udostępniała wszystko, co jest potrzebne, by tworzyć pełnoprawne aplikacje mobilne. Jednak zanim do tego dojdzie, możemy skorzystać z możliwości **hybrydowego podejścia** — nadal będziemy tworzyć aplikacje z wykorzystaniem **technologii internetowych**, ale użyjemy **biblioteki, która pełni rolę mostu** między naszym kodem a natywnymi możliwościami urządzeń.

Międzyplatformowe natywne aplikacje zbudowane w oparciu o technologie internetowe?
Brzmi całkiem niezłe!



Most
hybrydowych
aplikacji

Nowe możliwości	314
Jak działają aplikacje hybrydowe?	317
Budowanie mostu za pomocą PhoneGap	318
Dołącz do PhoneGap Build	321
Jak ma działać aplikacja?	322
Śledzenie ustrzelonych tartanów	323
Anatomia projektu Ustrzel tartan!	324
Pobieranie utworzonej aplikacji	328
Wybierz drogę	329
Kto co widział? Zapisujemy znalezione tartany	334
W czym nam może pomóc localStorage?	335
Sprawdzamy, co obsługuje przeglądarka	339
Używamy funkcji wyświetlającej znalezione tartany	340
Metody toggle i toggleClass	341
Znalazłeś tartan? Udowodnij to!	344
Zapręgamy PhoneGap do robienia zdjęć	345
Integracja z PhoneGap jest już prawie gotowa	347
Teraz jesteśmy gotowi na zgłębienie API mediaCapture	348
W jaki sposób obsłużymy akcję zakończoną powodzeniem?	349
W praktyce zawsze wygląda to trochę inaczej	350
Odrobina anonimowości	351
Już ostatnia sprawa!	353
Daliśmy radę!	354



VIRGIN 3G 4:20 PM

Ustrzel!
tartan!

ODSZUKAJ TARTANY I WYGRAJ
WYCIECZKĘ DO EDYNBURGA!

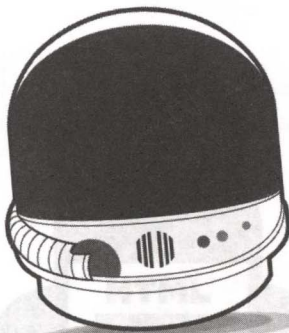
Podejście „future friendly” Odnajdywanie (jakiegoś) sensu w chaosie



Responsive Web Design. Wykrywanie urządzeń. Mobilne aplikacje internetowe. PhoneGap. Chwila... czego właściwie powinienem użyć?

Obecnie istnieje wiele metod tworzenia aplikacji za pomocą technologii mobilnych. Bardzo często w projektach łączy się wiele technik. Nie ma jednego, najlepszego rozwiązania, ale nie przejmuj się, ponieważ kluczem do sukcesu jest nadążanie za rozwojem technologii. **Nie bój się wyzwań.** Wystarczy przyswoić **podjęcie „future friendly”** i dać się ponieść fali, będąc przeświadczonym o swojej elastyczności i gotowości na wszystko, co przyniesie przyszłość.

I co teraz?	358
To nie takie proste	361
Manifest „future friendly”	362
Jeśli nie możesz się uodpornić na przyszłość, zaprzyjaźnij się z nią	364
Dziś aplikacja, jutro witryna	365
Czeka nas długa droga. Przyda się kilka wskazówek	366
Mieszamy mobilne składniki	369
Spójrz w przyszłość	371



• PRZECZYŃCIE OKRESKOWANIE •

Nie wieszono może się udać na każdym urządzeniu. Aby poradzić sobie w świecie wciąż rosnącego skomplikowania urządzeń, musimy się skupiać na najważniejszych – z punktu widzenia klientów – sprawach. I nie chodzi tu o rozwijanie oparte na największym wiodącym materiału, ale o wypracowanie treści i logik naszych interfejsów. Użytkownicy są w coraz większym stopniu zniechęceni rosnącym zbiorami informacji i konsekwentnie szukają sposobów na sprężenie treści. Muszą przycyżwić zagnieść zwięzłe usługi, zatem klienci i różnorodność urządzeń to za Ciebie.

• UNIWERSALNA ZAWARTOŚĆ •

Nadajmy im rekonesans jest tworzenie dobrze ustrukturyzowanej zawartości. Zastanów się, jak opracowana zawartość będzie się zachowywała w różnych kontekstach, uwzględniając ich ograniczenia i możliwości. Bądź odważny i korzystaj z nowych możliwości, ale pamiętaj, że przyszłość może podjąć w różnych kierunkach.

Nie ma nigdy idealnego odległy się zawartość zamierzonych urządzeń o ograniczonych możliwościach, jak i proste urządzenia z maksymalną funkcjonalnością. Tworząc strukturę zawartości, bierz to pod uwagę.

• DOWODZENIE PILOTA •

Korzystanie z wielu różnych urządzeń w codziennym życiu pozwala nam rozwijać zadania i informacje pomiędzy nimi. Gdy jakkolwiek zadanie jest zarządzane przez kolekcję urządzeń, także w nich można zastosować interakcje, i każdy z nich sobie najlepiej. Dzięki temu nie musimy dostarczać wszystkich aspektów danego zadania do każdego urządzenia, ponieważ zastanowimy się raczej na pracy w ekspozycjach możliwości urządzeń.

• ORBITOWANIE WOKÓŁ DANTCH •

Ekosystem urządzeń wymaga medycyny procesów międzyplatformowych i wydajnych mechanizmów wymiany danych. Dotarcie się do interakcyjnych, ale i dopiero co powstających możliwości, definiując dane w taki sposób, by:

- Był możliwy wielokrotny (i elastyczny) dostęp do danych.
- Były stosowane międzyplatformowe standardy.
- Dane były rekonesansowane na długoterminową integralność.
- Dane były zamawiane i szły odwołania do całej zawartości.
- Były wspierane operacje odczytu i zapisu.

• IDENTYFIKACJA OKRĘTÓW WIDNO •

Uwzględnienie w projekcie wszystkich możliwych konfiguracji urządzeń jest nie lada wyzwaniem. Proces adaptacji musi zapoczątkować, szczególnie wyszukaj odpowiednie i dobrze dobrane zbiory standardów dla różnych typów urządzeń. Standardy te muszą uwzględniać szczegółowym informacjami o profilu urządzenia.

Tego typu taksonomia może pomóc uszczególnić określone dane urządzenia w różnych produktach, pozwalając na dołączanie nowych typów urządzeń, które pojawiają się w przyszłości.

Ścinki

Sześć najważniejszych spraw (o których nie mówiliśmy)

A **Czujesz się, jakby coś Ci umknęło? Wiemy, co masz na myśli...** Zawsze jest tak samo — myślisz, że to już koniec, a okazuje się, że jest tego więcej. Nie mogliśmy się pohamować, by nie przekazać Ci kilku dodatkowych szczegółów, o których nie wspomnieliśmy w treści książki. Gdybyśmy chcieli napisać o wszystkich ciekawych sprawach, książkę musiałbyś transportować w panczernej walizce na kołach. Rzuć okiem, co dla Ciebie wybraлиśmy.

1. Testowanie na urządzeniach mobilnych	374
2. Zdalne debugowanie	376
3. Sprawdź, co obsługują przeglądarki	382
4. Interfejsy API urządzeń	384
5. Sklepy z aplikacjami oraz dystrybucja	385
6. RESS: REsponsive design + komponenty Server-Side	386

Postaw swój serwer

Gdzieś trzeba zacząć

B **„Mobilny internet” nie istnieje bez słowa „internet”.** Bez dwóch zdań. Jeśli chcesz się zająć tworzeniem witryn i aplikacji mobilnych, będziesz potrzebował serwera WWW. Prędzej czy później dojdzie do sytuacji, w której będziesz potrzebował części serwerowej swojej aplikacji. Możesz wtedy skorzystać z bezpłatnego lub komercyjnego hostingu albo uruchomić serwer na swoim komputerze. W tym dodatku opiszemy proces **stawiania lokalnego serwera WWW z obsługą PHP** z wykorzystaniem bezpłatnego oprogramowania.

Czego będziesz potrzebował?	388
Dostępny tylko lokalnie	389
Windows i Linux — zainstaluj i skonfiguruj XAMPP-a	390
Ciąg dalszy XAMPP-a	391
Na koniec Mac — MAMP	392
Sprawdź, czy zadokowałeś we właściwym porcie	393
Dostań się do swojego serwera	394
Informacje od phpinfo	396

Instalowanie WURFL

Jak wywęszyć urządzenia?



Pierwszy krok do rozwiązania tajemnicy wykrywania urządzeń wymaga trochę zachodu. Każdy przyzwoity gliniarz wie, że trzeba zbierać poszlaki i przesłuchiwać świadków. Musimy zacząć od mózgu całej operacji, którym jest **WURLF API dla PHP**. Później przyjdzie kolej na mięśniaka — jeden **plik XML** zawierający informacje o możliwościach tysięcy urządzeń. Nie będzie jednak łatwo zmusić ich do współpracy, więc będziemy musieli poświęcić im trochę czasu i pogrzebać w **konfiguracji**.

Skąd wziąć mózg?	398
A co z mięśniakiem?	399
Jak zmusić tę dwójkę do współpracy?	400
Czas na porządki w systemie plików	401
Zwróć na to uwagę!	402

Instalowanie SDK i narzędzi dla Androida

Zadbaj o środowisko



Aby gruntownie testować natywne aplikacje pod Androida, musisz przygotować sobie odpowiednie środowisko. Musisz dołączyć swój komputer do małego ekosystemu, do którego zagonisz wszystkie aplikacje Androida uruchamiane zarówno na wirtualnych (emulowanych), jak i rzeczywistych urządzeniach. Abyś mógł stać się pasterzem tego stada, pokażemy Ci, jak pobrać **SDK (ang. Software Development Kit) Androida**, jak zainstalować niektóre **narzędzia dla tej platformy**, jak **utworzyć wirtualne urządzenia** i jak **instalować oraz odinstalowywać aplikacje**.

Pobieramy SDK Androida	404
Wybierz odpowiednie narzędzia	405
Tworzenie nowego urządzenia wirtualnego	408
Znajdź właściwą ścieżkę	413



Skorowidz

417