



Technologia w e-commerce

TEORIA I PRAKTYKA. PORADNIK MENEDŻERA

Tomasz Ejtminowicz / Marcin Engelmann / Przemysław Fedorowicz / Grzegorz Godlewski
Michał Handl / Piotr Karwatka / Daniel Korzeniowski / Jan Lekszycki / Dorota Raczkowska
Joanna Sieniawska / Marek Stolarski / Kamil Zasada / Michał Żyliński

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.

Autorzy 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. Autorzy oraz Wydawnictwo HELION nie ponoszą również żadnej odpowiedzialności za ewentualne szkody wynikłe z wykorzystania informacji zawartych w książce.

Redaktorzy prowadzący: Barbara Gancarz-Wójcicka, Michał Mrowiec
Korekta: Katarzyna Jeznach, Izabela Strońska
Projekt okładki: Jarosław Brykowicz

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

Drogi Czytelniku!

Jeżeli chcesz ocenić tę książkę, zajrzyj pod adres

<http://onepress.pl/user/opinie?techec>

Możesz tam wpisać swoje uwagi, spostrzeżenia, recenzję.

ISBN: 978-83-246-7541-8

Copyright © Helion 2013

Printed in Poland.

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

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

Spis treści

Podziękowania	13
Wstęp	15
Informacje o autorach	19
Rozdział 1. Zarządzanie projektem i komunikacja	33
Jak przeprowadzić projekt z sukcesem?	34
Jak prowadzić projekt?	34
Co robić w sytuacjach kryzysowych?	37
Jak być dobrym project managerem?	39
Jak porządkować komunikację z firmą wdrożeniową?	41
Myślimy o komunikacji już na etapie podpisywania umowy	42
Etap analityczny i projektowy	43
Testy i odbiór	45
Utrzymanie	46
Jak budować i rozwijać zespół technologiczny?	50
Przede wszystkim — wspólny cel i szacunek dla pracy innych	50
Dobra atmosfera	51
Elitarność i autorytet	52
Odpowiedzialność	53
Nowe technologie	53
Wymiana wiedzy	54
Jak zapewnić sprawną współpracę project managera i programistów?	56
Jaką wybrać metodykę? Scrum i Kanban	61
Metodyki zwinne	62
Jak działa Scrum?	63
Czym jest Kanban i jak może pomóc?	65
Jakie są różnice pomiędzy Scrumem a Scrumbanem?	67
Jaką metodykę wybrać?	67

Rozdział 2. Projekt i dokumentacja	69
Jak skutecznie zebrać wymagania?	70
Jak radzić sobie ze zmianami wymagań?	74
Wpływ zmian na projekt	76
Jak wdrażać zmiany?	77
Czym jest i co powinien zawierać projekt funkcjonalny?	81
Czym jest projekt funkcjonalny?	81
Jak przygotować projekt?	82
Projektowanie a użyteczność — podstawy podstaw	87
Co powinna zawierać dokumentacja techniczna?	94
Dokumentacja techniczna projektu	94
Dokumentacja wdrożeniowa	95
Dokumentacja użytkownika	95
Zawartość dokumentacji	96
Testowanie dokumentacji	100
Rozdział 3. Wybór platformy	101
Dedykowana platforma, oprogramowanie pudełkowe czy open source?	102
Platforma dedykowana	104
Oprogramowanie pudełkowe	105
Oprogramowanie pudełkowe zamknięte, oprogramowanie w formie dzierżawy	105
Mit rozwiązań komercyjnych	106
Oprogramowanie pudełkowe otwarte — open source	107
Licencje open source	109
Porównanie dostępnych opcji	111
Dla praktyków — najpopularniejsze platformy	113
Jak ocenić oprogramowanie pod kątem dalszego rozwoju?	115
Przede wszystkim — licencja	116
Portfolio wdrożeń	117
Wsparcie producenta	117
Dokumentacja	117
Architektura	118
Testy automatyczne	119
Historia dotychczasowych zmian	120
Rozdział 4. Jakość i rozwój oprogramowania	121
Jak w odpowiedzialny sposób poprawiać jakość techniczną projektu?	122
Motywacje i cele	123
Jak refaktoryzować?	125
Jak planować rozwój oprogramowania oraz pracę?	127
Obawy użytkowników	128
Organizacja wiedzy	129

Jak wypracować nawyki związane z jakością?	135
Jak wydawać kolejne wersje oprogramowania?	139
Ciągła integracja	139
Przed wdrożeniem	144
Wdrożenie	145
Sposób wdrazania	146
Jak i kiedy aktualizować kod platformy oraz jak dbać o jego bezpieczeństwo?	148
Zmiany wykonywane w trakcie wdrożenia	149
Śledzenie historii zmian	150
Testy i zarządzanie wersjami	151
Jak przejąć niezrany projekt od firmy trzeciej i dalej go rozwijać?	152
Dokumentacja i szkolenia	152
Wywiad i poznanie systemu	154
Uruchomienie i podstawowe czynności	154
Analiza kodu źródłowego	154
Zespół do zadań specjalnych	156
Testy jednostkowe	156
System kontroli wersji i śledzenie zmian	157
Dziennik (log) projektu i dokumentacja	157
Rozdział 5. Hosting	159
Jak wybrać dostawcę hostingu i na co zwracać uwagę w umowach hostingowych?	160
Świadomy wybór firmy hostingowej	160
Po pierwsze: pytaj i zapisz w umowie lub regulaminie	161
Na co zwracać uwagę	161
Poziom jakości usług	163
Normy i standardy	164
Polska, Europa, świat	164
Umowa powierzenia przetwarzania danych	165
Co to jest SLA i jakie powinno mieć parametry?	169
Dostępność systemu	170
Tryb monitorowania, raportowania i eskalacji	171
Tryb zgłaszania problemów	172
Zakres asysty technicznej	172
Czas reakcji	172
Czas naprawy i obejścia błędu	173
Czas przywracania kopii zapasowych	173
Niezawodność sprzętu	174
Czas naprawy lub wymiany sprzętu	174
Przepustowość łącza i limity transferu	175
Dostępność sieci	175

Wirtualizacja. Jakie są plusy i minusy?	176
Czym jest wirtualizacja?	176
Wady i zalety	178
Najpopularniejsze technologie wirtualizacji	181
Czy to się opłaca?	182
Jak użyć chmury obliczeniowej do skalowania?	184
Wzorce pracy serwisów internetowych	184
Kłeska urodzaju	186
W czym mogą pomóc nam chmury?	188
Jak migrować?	190
Rozdział 6. Bezpieczeństwo danych	197
Jak bezpiecznie przetwarzać informacje?	198
Centrum przetwarzania	199
Serwery i osprzęt sieciowy	200
Systemy	202
Bazy danych	205
Jak bezpiecznie przetwarzać korespondencję masową?	211
Komponenty	211
Zgody i regulaminy	213
Zagrożenia i dodatkowa ochrona	214
Jakim testom bezpieczeństwa poddać aplikację?	217
Środowisko testowe	217
Głębokość testów	218
Wybór	223
Ustawiczne testowanie	224
System śledzenia błędów	225
Jak sprawić, aby pracownicy dbali o bezpieczeństwo danych?	226
Ufać czy kontrolować?	226
Zabezpieczenia	226
Kontrola administracyjna	227
Kontrola techniczna	227
Kontrola fizyczna	228
Kontrola prewencyjna, monitorująca i korygująca	228
Polityka czystego biurka i czystego ekranu	229
Rozdział 7. Aspekty prawne	231
Jakie gwarancje powinna mi dawać firma wdrożeniowa?	232
Wdrożenie	233
Utrzymanie	238

Jak zadbać o bezpieczeństwo danych osobowych?	244
Chronić należy nie tylko dane osobowe	245
Klient sklepu internetowego nie jest anonimowy	245
Dane osobowe to nie tylko imię i nazwisko	245
Zbiór danych osobowych a baza danych w systemie informatycznym	246
Wymagania techniczne wobec oprogramowania sklepu	247
Zabezpieczenie transmisji przez internet	248
Kopie zapasowe	249
Dokumentacja i procedury	249
Konsekwencje prawne niezabezpieczenia danych	251
Podsumowanie	252
Na co zwracać uwagę przy współpracy z firmą informatyczną?	253
Jak reagować na przypadki niewłaściwego realizowania umowy przez wykonawcę? ...	255
Jak ma się zachować zamawiający wobec wykonawcy, jeśli aplikacja zawiera wady?	256
Jak dokumentować przypadki wadliwego wykonywania umowy?	258
Jakie problemy występują w praktyce	
w trakcie współpracy zamawiającego z wykonawcą?	260
Jakie są skutki odstąpienia od umowy z firmą wdrożeniową?	272
Prawne znaczenie protokołów odbioru	273
Jak obsługiwać i rozwijać oprogramowanie bez udziału wykonawcy?	276
Co się dzieje, jeśli wykonawca „bankrutuje” w trakcie realizacji umowy?	281
Wybrane zagadnienia z zakresu open source	283
Jak chronić wartość intelektualną?	287
Prawne zabezpieczenia własności intelektualnej w branży e-commerce	287
Rozdział 8. Integracje	295
W jaki sposób zintegrować system księgowy ze sklepem?	296
Jakie systemy wymieniają dane?	296
Jakie dane są wymieniane?	298
Jak zapewnić sprawną komunikację?	301
Jak działa komunikacja ESB?	302
Platformy wymiany danych	303
Jak zaprojektować protokoły komunikacji?	306
Zależności pomiędzy protokołami	307
Zagadnienia projektowania	308
Bezpieczeństwo	311
Dokumentowanie protokołu	313
Jak monitorować komunikację oraz zapewnić jej ciągłość?	314
Wypadki chodzą po firmach	314
Monitorowanie komunikacji	316
Zapewnianie ciągłości komunikacji	320
Ciągłość i monitoring w ujęciu biznesowym	322

Rozdział 9. Skalowalność i wysoka dostępność	323
Co to jest wysoka dostępność?	324
Podstawy wysokiej dostępności	326
Wysoka dostępność jako proces	328
Czy Twój system jest wystarczająco dostępny?	328
Jak zapewnić dostępność bazy danych i aplikacji?	331
Redundancja sprzętowa	332
Nadmiarowość serwerów aplikacji	332
Replikacja bazy danych	333
Rozproszony system plików	335
Wirtualizacja	336
Chmury obliczeniowe	337
Czym jest skalowalność?	339
Po pierwsze — oprogramowanie	340
Po drugie — sprzęt	343
Skalowanie poziome warstwy aplikacji	344
Skalowanie warstwy bazy danych	346
Skalowanie warstwy przechowującej pliki	349
Gotowe rozwiązania	349
Dla praktyków	350
Co zrobić, jeśli sklep ma problemy wydajnościowe?	354
Monitoruj i wcześniej wykrywaj problemy	354
Stosuj pamięć podręczną — cache	355
Odciaż serwery aplikacji	355
Zainstaluj moduł mod_pagespeed	357
Stosuj osobny serwer bazy danych	357
Trzymaj sesje w bazie danych lub pamięci podręcznej	358
Dodawaj serwery aplikacji	358
Wybieraj dobre oprogramowanie	359
Ostatnie, ale nie mniej ważne — testuj	359
Co dalej?	359
Dla praktyków	359
Jak dbać o kopie zapasowe i szybkość ich przywracania?	361
Kopie zapasowe należy wykonywać	361
Archiwizacja danych a kopia zapasowa	361
RAID to nie kopia zapasowa	362
Na jakich nośnikach wykonywać i archiwizować kopie zapasowe	363
Systematyczność	364
Narzędzia	365
Odtwarzanie	366
Jak długo przechowywać	366

Jak uchronić się przed awarią centrum przetwarzania danych?	367
Procesy biznesowe w organizacji	367
Planowanie ciągłości działania	367
Disaster Recovery, czyli odtwarzanie po awarii	368
Ile jest czasu na przywrócenie i ile danych można utracić	369
Zasoby i procesy biznesowe	369
Zagrożenia i ich wpływ	370
Przeciwdziałanie i odtwarzanie po awarii	371
Bilans korzyści	373
Rozdział 10. Interfejs użytkownika, SEO	375
Jakie warunki powinna spełniać dostępna witryna internetowa?	376
Jak zapewnić dostępność do serwisu osobom niepełnosprawnym?	377
Testowanie serwisu	382
Korzyści	384
Jakie aspekty SEO wpływają na wysoką pozycję w wynikach wyszukiwania?	385
Bez błędny i semantyczny kod HTML	386
Każda strona sklepu powinna mieć tytuł i opis	386
Mapa sklepu — plik sitemap.xml	388
Plik robots.txt	388
Przyjazne adresy URL	388
Odnosiniki	389
Obrazki i ich opisy	390
Integracja z Google Analytics	390
Oznaczone dane strukturalne zgodnie ze schema.org	390
Integracja z serwisami społecznościowymi	391
Krótki czas ładowania stron = wyższa pozycja	392
Podsumowanie	392
Jak zwiększyć widoczność produktu w wyszukiwarce?	393
Google Rich Snippets — rozszerzony widok danych strukturalnych	393
Czym są dane strukturalne?	395
Od czego zacząć?	396
Jak optymalizować szybkość witryny?	399
Zmniejsz liczbę żądań HTTP	400
Minimalizuj rozmiar plików CSS i JavaScript	401
Pisz semantyczny kod HTML i unikaj skomplikowanych selektorów CSS	401
Włącz kompresję	403
Kompresuj pliki graficzne	403
Wykorzystaj pamięć podręczną przeglądarki	404
Używaj sieci dystrybucji treści (ang. CDN — Content Delivery Network)	
i rozprosz zasoby na kilka domen	405
Umieszczaj pliki arkuszy stylów CSS w sekcji <head>	406

Umieszczaj skrypty na końcu kodu HTML	407
Umieszczaj style CSS i skrypty JavaScript w zewnętrznych plikach	408
Optymalizacja szybkości ładowania witryn pod kątem urządzeń mobilnych	408
Jak wdrożyć wersję mobilną serwisu?	410
Czy ktoś pamięta jeszcze, co oznacza skrót WML?	411
Nadeszły czasy HTML5 i CSS3	412
Projektowanie pod urządzenia mobilne	412
Brak kursora myszy	412
Grube palce	413
Niskie rozdzielczości powracają!	414
Niska przepustowość łącza	414
Responsive Web Design	415
Elastyczny układ strony	416
Elastyczne zdjęcia	418
CSS3 Media Queries	419
Testowanie	420
Responsive Web Design to nie jedyne słuszne rozwiązanie	421
Rozdział 11. Porady praktyczne	423
Jakie testy powinny być przeprowadzane?	424
Testy użyteczności interfejsu użytkownika (ang. UI Tests)	425
Testy bezpieczeństwa	426
Testy obsługi sytuacji wyjątkowych (error handling tests)	427
Testy funkcjonalne, testy podstawowych scenariuszy i ścieżek	428
Testy regresyjne i testy dymne (smoke tests)	428
Testy przywracania systemu (recovery tests)	429
Testy obciążeniowe	430
Testy integracyjne	430
Jakie testy są obowiązkowe?	431
Jak napisać scenariusze testów funkcjonalnych?	432
Zakres testów	433
Założenia	434
Jak może wyglądać scenariusz?	434
Inne scenariusze testowe	437
Jak przetestować wydajność sklepu internetowego?	438
Co testować?	439
Jak testować?	439
Czym testować?	439
Jak testować integracje z systemami?	446
Czy warto testować?	447
Scenariusze testowe	447
Jak testować?	448

O czym należy pamiętać przy zmianie platformy?	450
Testy	450
Katalog produktów	452
Treści statyczne	452
SEO	453
Metody dostawy i płatności	454
Integracje	455
Poczta, domena, certyfikat SSL	455
Zgodność z prawem	456
W jaki sposób monitorować dostępność aplikacji?	457
Nie możesz zarządzać tym, czego nie mierzysz	457
Monitoring rozproszony geograficznie	458
Monitorowanie procesu zakupowego	458
Polecane narzędzia	458
Dostępność serwisu a jego wygląd w przeglądarce	459
Jakie są typy i zastosowania certyfikatów SSL w e-commerce?	460
Jakie dane zawierają certyfikaty?	460
Czy istnieje podział certyfikatów ze względu na moc szyfrowania?	461
Podział certyfikatów ze względu na sposób weryfikacji wraz z krótkim opisem procesu weryfikacji	461
Zastosowanie certyfikatów — gdzie stosować certyfikaty i dlaczego?	463
 Skorowidz	 467

Rozdział 3.

Wybór platformy

Wybór platformy sklepowej to jedna z tych decyzji, z których w przyszłości trudno będzie się wycofać. Wdrożenie technologii jest długotrwałe i drogie. Jeszcze droższa może być późniejsza zmiana.

Nie istnieją jedyne sprawdzone rozwiązania. To, co będzie świetnie działało przy małej skali biznesu, wraz z jego wzrostem może okazać się nieefektywne. Trzeba zwrócić uwagę na pułapki związane z dalszym rozwojem oprogramowania oraz jego kosztami.

NA PYTANIE

Dedykowana platforma, oprogramowanie pudełkowe czy open source?

odpowiada Michał Handl, główny specjalista ds. systemu ITS w ZDiUM we Wrocławiu

Na rynku dostępne jest obecnie całe spektrum platform do prowadzenia sprzedaży internetowej. Oferowane są różne modele licencyjne i scenariusze wdrożenia. Wybór oprogramowania ma znaczenie zasadnicze, ponieważ wpływa nie tylko na czas i koszty wdrożenia, ale też na możliwość późniejszego rozwoju.

Wybierając oprogramowanie, w pierwszej kolejności należy określić kluczowe wymagania. Bardzo ważne jest zebranie dokładnych i precyzyjnych wymagań funkcjonalnych i niefunkcjonalnych. Pomocne może się okazać zebranie odpowiedzi na szereg pytań, takich jak:

- ◆ W jaki sposób będziemy korzystać z panelu administracyjnego?
Czy posiadamy bardzo dużo produktów, które będziemy chcieli edytować jednocześnie?
- ◆ Jakie opcje edycji, rabatów i promocji muszą być dostępne, abyśmy mogli zaprezentować nasze produkty w sposób zgodny z modelem biznesowym?
- ◆ Czy produkty, które chcemy sprzedawać, są w jakiś sposób nietypowe (np. konfigurowalne; składają się z wielu elementów itp.)?
- ◆ Czy potrzebne są nam integracje (np. z wybranym systemem księgowym)?
- ◆ W jaki sposób i przy pomocy jakiego spedytora będziemy wysyłali przesyłki oraz w jaki sposób chcemy obsługiwać płatności?

Zbierając wymagania, będziemy mogli odpowiedzieć na pytanie: „Czy jest dostępne oprogramowanie, które spełnia je bez konieczności modyfikacji, czy też będziemy musieli wprowadzać dedykowane zmiany?”.

Jeśli oprogramowanie będzie wymagało zmian, kluczowe jest odpowiedzenie sobie na pytania: „Czy chcemy je samodzielnie wprowadzać, czy też zatrudnimy programistów, grafików, ewentualnie zespół techniczny? Czy zmiany będziemy zlecali, ponieważ nie mamy wewnętrznych kompetencji do ich wprowadzenia?”.

Kluczowe jest też pytanie: „Jak duży budżet jesteśmy w stanie poświęcić na uruchomienie platformy oraz jakie stoją przed nami perspektywy rozwoju (czy w perspektywie roku, dwóch widzimy konieczność dodawania nowych funkcjonalności, typów produktów)?”.

Wyposażeni w taką wiedzę możemy dokonać przeglądu dostępnych platform i podjąć stosowną decyzję.

Abstrahując od konkretnych marek oprogramowania i dostępnych w nich funkcjonalności, platformy e-commerce można w zasadzie podzielić na:

1. Platformy dedykowane.
2. Oprogramowanie pudełkowe:
 - ◆ Oprogramowanie pudełkowe zamknięte (płatne),
 - ◆ Oprogramowanie pudełkowe zamknięte (płatne w formie dzierżawy),
 - ◆ Oprogramowanie pudełkowe otwarte.

Taki podział wynika z kilku parametrów kluczowych:

- ◆ możliwości (lub jej braku) wprowadzania dedykowanych zmian oraz funkcjonalności dostępnych w standardzie;
- ◆ sposobu rozliczeń/płatności, a co za tym idzie również kosztów wdrożenia;
- ◆ licencji i otwartości na modyfikacje.

Wybór, który będzie optymalny dla startującego przedsiębiorstwa, może się okazać zupełnie nieefektywny przy działającej już od jakiegoś czasu firmie szukającej perspektyw rozwoju. Warto przyjrzeć się wszystkim dostępnym opcjom i nie podejmować decyzji pochopnie. Bardzo łatwo można dokonać wyboru, który będzie efektywny kosztowo, ale może bardzo ograniczyć możliwości naszego biznesu. Z drugiej strony nietrudno o wybór oprogramowania o ogromnych możliwościach, których nie będziemy w stanie wykorzystać.

Platforma dedykowana

Platforma dedykowana jest to oprogramowanie tworzone specjalnie na potrzeby zamawiającego, ewentualnie gotowe oprogramowanie bazowe, które jest w bardzo dużym stopniu modyfikowane pod konkretne potrzeby. Klient może mieć wpływ na wszystkie elementy i aspekty wytworzonego systemu, począwszy od zastosowanej technologii, a skończywszy na procesach przepływu danych, jakie zostaną zaimplementowane w tym rozwiązaniu.

Zaletą takiego podejścia jest możliwość stworzenia platformy idealnie dopasowanej do środowiska informatycznego, które już istnieje w przedsiębiorstwie (np. bardzo dobra integracja z systemem księgowym lub magazynowym). Czasami takie podejście jest jedynym możliwym. Jeśli chcemy zastosować niecodzienny model biznesowy (przykładem mogą być zakupy grupowe), mamy do czynienia z nietypowym produktem (na przykład produkty konfigurowalne lub cyfrowe) lub ze skomplikowaną ścieżką zakupową — wówczas daleko idące modyfikacje oprogramowania będą jedynym wyjściem.

Dedykowane rozwiązania mogą być też podyktowane wymaganiami niefunkcjonalnymi związanymi z rozwojem biznesu. Duża skalowalność, wysoka dostępność (więcej o tym w rozdziale „Skalowalność i wysoka dostępność”) mogą pociągać za sobą konieczność optymalizacji oprogramowania. W przypadku rozwiązań o naprawdę dużej skali może się okazać, że tylko utrzymywanie oprogramowania na własnych serwerach i własny rozwój będą uzasadnione ekonomicznie.

Rozwiązania dedykowane niosą ze sobą wiele zagrożeń, z których trzeba zdawać sobie sprawę, podejmując wybór.

Dedykowane oprogramowanie to ogromna odpowiedzialność. Pomijając aspekt kosztowy i czasowy (wytworzenie oprogramowania od zera jest zazwyczaj bardzo drogie i trwa długo), warto się zastanowić, czy będziemy w stanie zaprojektować i zrealizować system posiadający przewagi konkurencyjne. Czy ten proces nie będzie trwał zbyt długo?

Istniejące aplikacje są rozwijane od lat i ich twórcy w tym czasie popełnili, a następnie poprawili setki błędów. Musimy wykonać projekt całego systemu, dokumentację, przeprowadzić prace programistyczne, a następnie przeprowadzić testy (na które inni producenci przeznaczają miesięcznie dziesiątki godzin).

Nie ma co liczyć na aktualizacje producenta (bo sami nim jesteśmy!). Nie otrzymamy dodatkowych funkcjonalności, a błędy w oprogramowaniu nie zostaną rozwiązane w ramach bezpłatnych aktualizacji. Koszt modyfikacji oraz utrzymania oprogramowania po jego uruchomieniu może wielokrotnie przewyższać koszty wdrożenia pierwszej wersji.

Oprogramowanie pudełkowe

Oprogramowanie pudełkowe. To rodzaj oprogramowania, które można (mówiąc skrótowo) kupić bezpośrednio od dystrybutora, zainstalować i używać.

W większości przypadków charakteryzuje się ono dość zamkniętą strukturą, której użytkownik nie musi później modyfikować.

Istnieje bardzo dużo gotowych produktów tego typu. Różnią się między sobą licencją, ceną oraz możliwościami. W dużym skrócie możemy wyróżnić następujące formy oprogramowania pudełkowego:

- ◆ **Oprogramowanie pudełkowe w formie dzierżawy** (*Software As a Service* — SaaS, czyli oprogramowanie jako usługa) to rozwiązanie płatne w formie ryczałtu miesięcznego i jednocześnie najtańszy sposób na rozpoczęcie sprzedaży w internecie. Zamawiający nie martwi się hostingiem, instalacją ani konfiguracją sklepu. Korzysta ze sklepu jak z usługi bez konieczności wykonywania dodatkowych prac czy ponoszenia dodatkowych kosztów.
- ◆ **Oprogramowanie pudełkowe zamknięte** to różnego rodzaju autorskie rozwiązania, wspierane najczęściej tylko przez producenta. Oprogramowanie jest „zamknięte” przez licencję, która nie umożliwia wprowadzania zmian przez zamawiającego.
- ◆ **Oprogramowanie pudełkowe otwarte** (darmowe lub płatne) to oprogramowanie wydane na jednej z licencji nurtu open source. Producenci często świadczą dodatkowe, płatne wsparcie. Zamawiający może je jednak dowolnie modyfikować i rozwijać we własnym zakresie.

Oprogramowanie pudełkowe zamknięte, oprogramowanie w formie dzierżawy

Oprogramowanie dostarczone w formie dzierżawy można podsumować stwierdzeniem: „szybko i tanio, ale z ograniczonym wachlarzem możliwości”.

Takie rozwiązanie dobre jest dla wszystkich tych, którzy szybko i bez dużych kosztów chcą uruchomić sklep internetowy. Rozwiązanie to jest idealne dla osób, które dopiero startują — chcą wypróbować pomysł bez ponoszenia (być może nieuzasadnionych) kosztów. Do uruchomienia sklepu nie są potrzebne prawie żadne informatyczne zdolności. Zazwyczaj jedyną zmianą w oprogramowaniu, którą należy wykonać, jest adaptacja szablonu graficznego. Dopasowanie grafiki jest wykonywane przez dostawcę platformy lub firmę partnerską wykonującą profesjonalnie usługi tego typu. Po wdrożeniu szablonu resztę działań można zrealizować samodzielnie z poziomu panelu administracyjnego.

W przypadku oprogramowania w formie dzierżawy (SaaS) praktycznie nie ma możliwości wprowadzania do oprogramowania jakichkolwiek zmian. Dzieje się tak, ponieważ system używany jest w takiej samej wersji przez wszystkich klientów platformy, a utrzymywanie osobnych modyfikacji jest dla producenta zazwyczaj nieopłacalne.

Inaczej jest w przypadku oprogramowania zamkniętego, które możemy pobrać i zainstalować na własnym serwerze. W tym przypadku zmiany zazwyczaj są możliwe. Koniecznie trzeba zwrócić uwagę na postanowienia licencji oraz to, czy wprowadzenie zmian nie spowoduje wygaśnięcia gwarancji na oprogramowanie. Jeśli wprowadzanie zmian jest możliwe tylko dla producenta systemu, będziemy od niego uzależnieni. Wprowadzenie modyfikacji może być kosztowne lub długotrwałe.

Największą zaletą oprogramowania pudełkowego jest ilość funkcji, jakie posiada w standardzie. Platformy walczą ze sobą pod względem jakości produktu oraz dopasowania do rynku (np. integracje z lokalnymi firmami kurierskimi i płatnościami). Producenci mają z jednej strony motywację do rozwoju, która wynika wprost z opłat za oprogramowanie, a z drugiej strony posiadają dużą bazę klientów sugerujących zmiany i nowe funkcje, które najczęściej stają się dostępne po pewnym czasie dla wszystkich pozostałych.

Okazuje się, że największym zagrożeniem w takim przypadku jest stabilność nie oprogramowania, a... jego producenta. Decydując się na wybór platformy rozwijanej przez jedną firmę, czasami mniejszą niż biznes zamawiającego, ponosi on duże ryzyko. Firma taka może zaprzestać rozwoju produktu lub zaprzestać działalności w ogóle (co można wtedy zrobić, opisano w rozdziale „Aspekty prawne”).

Decydując się na oprogramowanie mające wesprzeć nasz biznes przez najbliższych kilka lat, warto pomyśleć o rozwiązaniu otwartym, które da większe bezpieczeństwo biznesowe.

Mit rozwiązań komercyjnych

Firmy często korzystają z płatnych rozwiązań, ponieważ w naszej podświadomości zakorzenione jest przeświadczenie, że oprogramowanie darmowe musi być kiepskiej jakości.

W praktyce to właśnie rozwiązania komercyjne często mają istotne wady, takie jak:

- ◆ **Zamknięty kod źródłowy** — w większości przypadków rozwiązań komercyjnych kod źródłowy aplikacji nie jest dostępny do wglądu, modyfikacji czy zmian, przez co wykrycie i poprawienie błędów

w oprogramowaniu jest utrudnione. Mniejsza ilość opublikowanych informacji na temat błędów nie oznacza, że oprogramowanie jest lepsze.

- ◆ **Uzależnienie się od dostawcy rozwiązania** — (ang. *vendor lock-in*) wdrożenie zamkniętej platformy zamyka drogę do wyboru innej firmy wdrażającej czy rozwijającej wdrożony produkt. Powoduje to bardzo często problem związany z dużym kosztem utrzymania powdrożeniowego takiej platformy.
- ◆ **Niepewność rozwoju** — oprogramowanie komercyjne zwykle jest rozwijane przez jednego producenta, który może zaprzestać dalszego wsparcia. Użytkownik nie ma na to żadnego wpływu (co można zrobić w takim przypadku — w rozdziale „Aspekty prawne”).
- ◆ **Brak dokumentacji** — wiele przypadków rozwiązań komercyjnych pokazuje, że dokumentacja oprogramowania jest niepełna, istnieje tylko w formie porad do najczęstszych problemów i procedur. Dokumentacja techniczna nie musi być aktualna, bo... korzystałby z niej tylko producent (mający dostęp do kodu źródłowego).
- ◆ **Brak wsparcia** — wsparcie często ograniczone jest do płatnego supportu świadczonego przez producenta aplikacji, brak lub ograniczona ilość porad na forach internetowych.

Oprogramowanie pudełkowe otwarte — open source

Na rynku można spotkać kilka dobrych otwartych rozwiązań, np. bardzo popularny PrestaShop, Magento czy Gekosale. Oprogramowanie takie dzięki zastosowaniu licencji otwartej może być dowolnie dostosowywane do potrzeb przez użytkownika. Może on skupić się na modyfikacji tylko tych mechanizmów, które powinny działać w sposób dedykowany, podczas gdy wszystkie inne działają „z pudełka” bez wprowadzania zmian. Trzeba pamiętać o tym, że jeśli dokona się znacznych modyfikacji systemu we własnym zakresie, prawdopodobnie utraci się możliwość instalowania aktualizacji dostarczanych przez producenta. Przy zmianie kodu bazowego platformy w celu wykonania aktualizacji mogą być konieczne ręczne porównania i scalania plików.

Open source to nie tylko sposób dystrybucji, ale także ruch wolnego oprogramowania, który od końca lat 80. cały czas się rozwija i skupia wokół siebie wielu sympatyków. Istnieje co najmniej kilka modeli biznesowych wspierających producentów oprogramowania otwartego.

Przykładem osiągnięcia zysków z oprogramowania open source może być świadczenie dodatkowych płatnych usług związanych ze wsparciem dla darmowego

oprogramowania lub dystrybucja darmowej wersji oprogramowania z ograniczonymi opcjami, natomiast w wersji komercyjnej w opcji z rozszerzonymi funkcjonalnościami.

Rozwiązania open source bardzo się rozpowszechniły i dzisiaj wiele serwisów internetowych opartych jest na nich w części lub całości (np. system operacyjny Linux, serwer WWW Apache, baza danych MySQL, język programowania PHP).

Oprogramowanie open source, jak sama nazwa wskazuje, ma otwarte źródła, dzięki czemu każdy użytkownik, który używa tego oprogramowania, może w razie potrzeby analizować i modyfikować jego kod źródłowy — poprawiając błędy lub wprowadzając nowe funkcje.

Ponieważ każdy może to zrobić, to zagrożeniem będzie pojawianie się w internecie informacji o gotowych sposobach na włamanie czy wykradnięcie cennych informacji. Jednocześnie dzięki temu, że kod może być analizowany przez dosłownie każdego, przyspieszone jest wykrywanie błędów, a co za tym idzie, także pojawianie się łatek i zabezpieczeń. Z tego wynika konieczność częstego aktualizowania i śledzenia zmian systemów open source, z których korzystamy.

Decydując się na oprogramowanie otwarte, koniecznie powinniśmy sprawdzić opinie o jego bezpieczeństwie na forach dyskusyjnych. Warto obserwować też przez pewien czas aktualizacje strony internetowej produktu. Jeśli nie pojawiają się żadne aktualizacje, to być może produkt, któremu się przyglądamy, jest już przestarzały i nierozwijany. Spadająca motywacja jest częstą bolączką twórców oprogramowania wolnego, którzy nie posiadają wsparcia finansowego.

Rozwój projektów open source daje korzyści producentom rozwiązań komercyjnych. Między innymi IBM, RedHat, Varien, Intel i inni rozwijają oraz wspierają projekty otwarte. Z jednej strony traktują taki model jako reklamę (dzięki rozwiązaniom otwartym jest szansa trafić do większej liczby użytkowników, którzy zapoznają się z rozwiązaniem i prawdopodobnie w przyszłości będą korzystali z płatnych wersji, płatnego wsparcia itp.). Z drugiej strony takie podejście umożliwia im wykorzystanie wiedzy i doświadczenia programistów, którzy zainteresują się oprogramowaniem i dostarczą nowe moduły czy też znajdą i poprawią błędy.

Zanim zastanowimy się nad rozwiązaniem open source, musimy także zwrócić szczególną uwagę na licencję dystrybucyjną danego oprogramowania. Sam fakt tego, że program posiada otwarte źródła, nie jest równoznaczny z tym, iż można go używać w dowolny sposób.

Licencje open source

Oprogramowanie open source dystrybuowane jest na podstawie różnorodnych licencji (tabela 3.1). Niektóre z nich są dość skomplikowane i różnią się od innych tylko niewielkimi niuansami. Czasami drobne różnice okazują się kluczowe. Lista dostępnych licencji open source znajduje się pod adresem: <http://www.gnu.org/licenses/license-list.pl.html>.

Licencje wolnego oprogramowania dzieli się na zgodne oraz niezgodne z GPL (*Gnu Public License*).

Elementy, które powtarzają się w większości licencji, to:

- ◆ **Uznanie autorstwa.** Niektóre licencje wymagają umieszczenia w widocznym dla użytkownika miejscu informacji *O autorze*, inne wymagają tylko, aby informacja o autorze nie była kasowana z komentarza w kodzie źródłowym. Są także licencje zawierające klauzule, z których jasno wynika, że nie należy używać nazwy oryginalnego oprogramowania, jeśli mówimy o oprogramowaniu zmodyfikowanym.
- ◆ **Copyleft.** Licencje zawierające taką klauzulę są czasami nazywane wirusowymi. *Jeśli użyjesz jakiegokolwiek fragmentu kodu, który używa tej licencji, wówczas cały program musi być wydany na licencji kompatybilnej.* Celem tej klauzuli jest zatrzymanie producentów oprogramowania od kopiowania darmowych rozwiązań i budowaniu na ich podstawie rozwiązań dostępnych tylko odpłatnie. Copyleft dzielimy na dwie kategorie:
 - ◆ **Silna**, która definiuje, że tylko programy, które używają tej samej licencji, mogą używać kodu dystrybuowanego pod tą licencją;
 - ◆ **Staba**, która definiuje, że tylko zmiany kodu muszą być publikowane na tej samej licencji, natomiast dodatkowe elementy oprogramowania mogą być zbudowane na zasadzie biblioteki i publikowane na podstawie innej licencji.
- ◆ **Tolerancja.** Czasami ludzie dopytują się, jak bardzo tolerancyjna jest licencja. Im bardziej licencja jest tolerancyjna, tym więcej można zrobić z oprogramowaniem. Zasięg ograniczeń licencyjnych waha się od bardzo liberalnych licencji (takich jak MIT czy X License) do bardziej ograniczonych pozwoleń (takich jak EULA), dotyczących pakietów komercyjnych.

Tabela 3.1. Zestawienie najpopularniejszych licencji wraz z uwagami

Licencja	Kod źródłowy	Typ	Ilość klauzul	Uwagi
Brak licencji	Otwarty	Żaden	0	Brak licencji tak naprawdę nie oznacza, że można użyć kodu za darmo. Jest on bowiem chroniony prawami autorskimi i aby go użyć, należy skontaktować się bezpośrednio z autorem i zapytać o pozwolenie.
Domena publiczna	Otwarty	Tolerancyjny	0	Jeśli oprogramowanie umieszczone jest jako domena publiczna, to każdy może go wykorzystać w dowolnym celu.
Licencja GPL	Otwarty	Copyleft	12	Dystrybucja oprogramowania nie może w żaden sposób być prawnie zastrzeżona, a nawet zmodyfikowana.
Licencja LGPL	Otwarty	Copyleft (w większości przypadków)	16	Licencja identyczna jak GPL ze sprytnie skonstruowanym mechanizmem, który pozwala nakładać ograniczenia dystrybucyjne.
Licencja MIT/X11	Otwarty	Tolerancyjny	2	Bardzo krótka i ogólna. Zawiera tylko ogólne zastrzeżenia co do odpowiedzialności prawnej.
Licencja BSD	Otwarty	Tolerancyjny	2	Bardzo krótka i ogólna. Zawiera tylko ogólne zastrzeżenia co do odpowiedzialności prawnej.
Licencja Apache	Otwarty	Tolerancyjny	9	Wymaga stworzenia dzieła pochodnego, aby zgłosić jakiegokolwiek zastrzeżenia prawne do kodu źródłowego.
Eclipse Public License	Otwarty	Tolerancyjny	7	Licencja przyjazna dla biznesu. Umożliwia dla prac pochodnych wybór własnej licencji na ich publikację.
Mozilla Public License	Otwarty	Słaby copyleft	13	Umożliwia wprowadzenie liberalnej mieszanki prawnie zastrzeżonego licencjonowania oprogramowania.
MS Permissive License	Otwarty	Tolerancyjny	3	Bardzo podobna do licencji MIT oraz BSD. Nie została zaakceptowana formalnie przez OSI.

Tabela 3.1. Zestawienie najpopularniejszych licencji wraz z uwagami — ciąg dalszy

Licencja	Kod źródłowy	Typ	Ilość klauzul	Uwagi
MS Community License	Otwarty	Copyleft	3	Bardzo podobna do licencji GPL. Wymaga, aby wytworzony kod został opublikowany dla społeczności. Ten typ licencji nie został formalnie zaakceptowany przez OSI.
MS Reference License	Własność	Tylko do odczytu	3	Można przeglądać kod źródłowy, zrobić kopię, ale nie można go w jakikolwiek sposób używać ani zmieniać.

Porównanie dostępnych opcji

Biorąc pod uwagę specyfikę, w jakiej się obracamy, czyli wdrożenia e-commerce, trzeba wziąć pod uwagę czas i budżet. Jak wcześniej już wspomniano, najtańszym oprogramowaniem (pozbawionym wszelkich aspektów związanych z technologią) jest platforma w formie najmu. Większość rozwiązań open source oczywiście samo w sobie jest rozpowszechniane jako darmowe, jednak wdrożenie jako proces wprowadzania zmian i dostosowań oczywiście darmowe już nie jest. W ramach kosztów wdrożenia trzeba ująć wszystkie elementy konieczne do dostosowania lub modyfikacji, a mianowicie:

- ◆ dostosowanie wyglądu strony internetowej,
- ◆ tłumaczenia językowe,
- ◆ instalację dodatkowych modułów (np. moduł płatności),
- ◆ implementację mechanizmów integracji (w niektórych przypadkach wytworzenie ich od podstaw).

Wybór platformy jest bardzo istotny. Wybór formy dystrybucji i licencji jest decyzją podstawową — na jej podstawie powinno się dokonać analizy funkcjonalności i kosztów modyfikacji. Dopiero takie dwutorowe zestawienie pozwoli podjąć odpowiedzialną decyzję (tabela 3.2).

Warto mieć na uwadze, jakie możliwości i ograniczenia są związane z modelami dystrybucji, które zostały omówione w tym rozdziale.

Skorowidz

A

administrator baz danych, 357
adresy URL, 388
akceptacje, 43
aktualizacje, 128, 148, 203
algorytm kompresji, 403
algorytmy, 96
analiza
działającego oprogramowania, 218, 224
integracji, 44
kodu źródłowego, 154, 224, 427
konkurencji, 44, 82
PERT, 62
powłamaniowa, 204
techniczna, 44
wymagań, 81
wymagań biznesowych, 71
zdarzeń systemowych, 203
zgodności z WCAG, 382
analyzer ruchu sieciowego, 223
Anchor Text, 378
aneks do umowy, 37
anonimizacja rekordów, 218
API, Application Programming Interface, 119, 149
aplikacje
biznesowe, 187
natywne, 421
architektura
hybrydowa, 191
informacji, 84, 88
archiwizacja danych, archiving, 361
asynchroniczne operacje I/O, 345

atak

CSRF, 426
DDoS, 370
SQL Injection, 207, 427
XSS, 426
atrybut
itemprop, 397
itemscope, 397
itemtype, 397
title, 378
atrybuty
schema.org, 396
WAI-ARIA, 379
audyt systemu, 427
automatyczne
skalowanie, 189, 417
testowanie, 220, 221
testy aplikacji, 428
testy integracji, 430
autoryzacja działań, 312
awaria
centrum przetwarzania danych, 367
sprzętu, 174
techniczna, 370
urządzeń, 200

B

B2B, business to business, 302
BABOK, 71
Enterprise Analysis, 72
ocena zebranych wymagań, 72
odkrywanie wymagań, 72
planowanie analizy, 72
Requirements Analysis, 72
zarządzanie zmianami, 72

backlog, 131
backup, 206
badanie usterek, 220
bankructwo producenta, 282
barier chroot, 218
baza danych, 119, 205
 MongoDB, 348
 MySQL, 348
 PostgreSQL, 348
baza wiedzy, 327
BCP, Business Continuity Planning, 367
bezkosztowość, 292
bezpieczeństwo, 148
 danych, 197–229
 kontrola, 226–228
 standardy, 226
 danych osobowych, 244
 fizyczne danych, 199
 informacji, 361
 komunikacji, 304
 protokołu, 312
 przesyłanych danych, 310
 przetwarzanych danych, 198, 463
 systemu, 426
 w chmurze
 centrum danych, 193
 dostawcy usług, 193
 oprogramowanie, 193
BI, Business Intelligence, 297
biblioteka ORM, 98, 358
binlog, 334
blokowanie dostępu robotom, 388
błąd, 137
 główny, 236
 kategorii drugiej, 240
 kategorii pierwszej, 239
 kategorii trzeciej, 240
 krytyczny, 236
 niepoprawny, 236
 normalny, 236
brak łączności z serwerem, 327
broker usług chmury, 167
burza mózgów, 72, 78
Business Model Canvas, 82

C

CAPTCHA, 213, 381
CCPM, Critical Chain Project Management, 62
CDN, Content Delivery Network, 349, 353, 405, 458
CDN, Content Distribution Network, 167
cel projektu, 44, 57
cele UX, 88
centrum danych, Data Center, 326
centrum przetwarzania, 161, 199
 instrukcje, 209
 procedury, 209
certyfikat SSL, 248, 456, 460, 463–466
certyfikaty typu
 DV, 460, 461
 EV, 461, 462
 OV, 461, 462
checklista, 37, 382
chmura, 167
 hybrydowa, 191
 obliczeniowa, 178, 184, 188, 337, 349
 migracja, 190
 prywatna, 337
 publiczna, 337
 skalowanie serwisu, 188
 usługi dodatkowe, 189
ciągłość
 działania, 367
 komunikacji, 320
 na poziomie aplikacji, 320
 na poziomie infrastruktury, 321
 na poziomie systemu, 321
Copyleft, 109
CRM, Customer Relationship Management, 297
CSRF, Cross Site Request Forgery, 426
CSS3 Media Queries, 417, 419
czas
 ładowania stron, 355
 ładowania witryny, 438
 naprawy, 173
 naprawy błędów, 234, 241
 naprawy sprzętu, 174
 niedostępności, 325
 przywracania kopii zapasowych, 173
 przywrócenia do działania, 370
 reakcji, 234, 236, 242
 reakcji na zgłoszenia, 172

- realizacji połączeń sieciowych, 354
- trwania projektu, 87
- wygasania elementów, 355

czytniki ekranowe

- JAWS, 382
- Window-Eyes, 382

D

dane

- kontaktowe, 453
- osobowe, 164, 245, 249
- strukturalne, 390, 394, 395

data ważności zasobów, 405

dedykowane środowiska skalowalne, 167

delegacja domeny, 455

desk research, 82

DFS, Distributed File System, 335

diagram

- ERD, 153
- komunikacji, 303
- przepływu danych, 300
- przypadków użycia, 73

diagramy

- UML, 96, 97
- zależności danych, 96

dobre praktyki, 99, 392, 454

dochodzenie naruszonych praw, 292

dodatek

- Web Developer, 382
- Window Resizer, 420

dodawanie serwerów aplikacji, 358

dokumentacja, 69, 278

- dla webdevelopera, 90
- funkcjonalna, 96
- kodu źródłowego, 119
- protokołu wymiany danych, 300
- systemu informatycznego, 157
- techniczna, 94, 96, 152, 327
- użytkownika, 95, 99
- wadliwego wykonywania umowy, 258
- wdrożeniowa, 95, 98

dokumentowanie

- protokołu, 313
- wymagań systemowych, 74

DOM, Document Object Model, 402

dostawca

- hostingu, 160
- infrastruktury jako usługi, 167

sieci CDN, 353

usług, 166

dostęp do

- bazy adresowej, 214
- bazy danych, 212, 358
- kodu źródłowego, 277, 278

dostępność, 323, 328, 330

aplikacji, 457

formularza, 379

łącza internetowego, 163

serwerów, 163

serwisu, 187, 377, 459

sieci, 175

sprzętu, 174

stron internetowych, 376

systemu, 170, 171

DR, Disaster Recovery, 368

DRY, Don't Repeat Yourself, 124

drzewo

DOM, 402

kategorii, 452

replikacji, 334

dyrektywa

@import, 407

ExpiresDefault, 405

dyski SSD, 343, 357

dział

biznesowy, 51

IT, 50, 52, 53

QA, 129, 140

działania project managera, 40

działanie systemów informatycznych, 43

dziedziczenie właściwości CSS, 402

dziennik, 133, 318

projektu, 126, 157

systemu, 154

E

EAV, Entity-Attribute-Value, 119

EDI, Electronic Data Interchange, 300, 307

efekt wykopu, 186

elastyczne zdjęcia, 418

elastyczny układ strony, 416

e-maile transakcyjne, 453

ERD, Entity Relationship Diagram, 96, 153

ERP, Enterprise Resource Planning, 297, 430, 446

ESB, Enterprise Service Bus, 302

eskalacja systemu, 171

etap

- analityczny, 43
- podpisywania umowy, 42
- projektowy, 43
- testowania, 45
- utrzymania systemu, 46
- wdrożeniowy, 37

etapy projektu, 35

etykieta, tag, 328

etykiety formularzy, 379

F

Facebook, 392

fałszywe dane, fake, 289

fazy projektu, 41

feedback, 37

filtr pakietów, 203

firewall, 203

firma

- hostingowa, 160
- wdrożeniowa, 44

format

- GIF, 404
- JPEG, 403
- PNG, 404
- PNG8, 404
- wymiany informacji, 299
- XML, 300, 302

formaty danych, 311

formularz GODO, 235

formularze, 380

framework, 318

- 1140 CSS Grid, 418
- ASP.NET MVC, 342
- Fluid 960 grid system, 418
- PhoneGap, 421
- Symfony2, 342
- Twitter Bootstrap, 418
- Zend Framework, 342

frameworki asynchroniczne, 346

front-end, 451

funkcje, 99

- nieprzewidziane w umowie, 37
- sklepu, 377

funkcjonalności aplikacji, 320

fuzzing, 219, 221

G

gałęzie projektu, 142

generowanie

- plików testowych, 442
- specyfikacji, 86

GODO, 252, 456

głębokość testów, 218

Google

- Analytics, 390
- Apps, 455
- Closure Tools, 401
- PageSpeed Insights, 408
- Rich Snippets, 390, 393, 395
- Webmaster Tools, 390

GPL, General Public License, 284

GPL, Gnu Public License, 109

granulacja scenariuszy, 436

grid, 418

gromadzenie wiedzy, 43, 58

gwarancje firmy wdrożeniowej, 232

gwarantowane czasy reakcji, 242

H

harmonogram

- dostarczania materiałów, 43
- projektu, 35
- realizacji, 45
- szczegółowy projektu, 47, 129
- wykonywania zadań, 36

hierarchiczność nagłówków, 386

historia zmian, 128

historie użytkownika, user stories, 77, 96

hoster, 166

hosting, 159–193

hosting współdzielony, 167

hotfix, 130

HTML5, 378

I

IA, information architecture, 88

IaaS, 167

IDE, Integrated Development Environment, 155

IMAP, Internet Message Access Protocol, 307

implementacja, 97

implementacja mikro danych, 396

informacje

- EXIF, 403

- o autorach, 19
- o szybkich poprawkach, 150
- o zmianach, 120
- infrastruktura, 341
- inspekcja kodu źródłowego, 420
- integracja
 - systemów, 295–322, 446, 449, 455
 - z serwisami społecznościowymi, 391
- integralność przesyłanych informacji, 311
- interakcja użytkownika z firmą, 88
- interfejs
 - programistyczny, 119
 - programu JMeter, 440
 - użytkownika, 375–421
- interfejsy wymiany danych, 298
- interpreter skryptów, 357
- IoC, Inversion Of Control, 341
- IP, Internet Protocol, 305
- iteracje MVP, 77
- iteracyjność, 55
- IxD, interaction design, 88

J

- jakość
 - grafiki, 404
 - oprogramowania, 121
 - projektu, 135, 138
 - techniczna projektu, 122
 - usług, 190
- język
 - CSS 3, 403
 - dokumentu, 380
 - XHTML 1.0 Strict, 386
 - XML, 302

K

- kalkulator zysków i kosztów, 183
- kanal
 - alfa, 404
 - komunikacyjny, 309
- Kanban, 61
 - limit pracy w toku, 65
 - przepływ, flow, 65
 - tablica, 65, 67
 - wizualizacja, 65
 - zarządzanie strumieniem, 65

- kary umowne, 237, 242, 265
- catalog usług, 169
- kategorie błędów, 236
- kick off projektu, 34
- kierownik
 - IT, 52
 - techniczny, 51
- klaster, 327
- klasy, 96
- klasy systemów, 297
- kod
 - karty produktu, 396
 - pośredni, 342
- kody EAN, 70
- kompresja gzip, 401, 403
- komunikacja, 33, 54, 314
 - z firmą wdrożeniową, 41
 - bezpośrednia, 44, 48
 - ESB, 302
 - między systemami, 297, 301, 449
 - w projektach, 59
 - w zespole, 36, 56, 59
 - wieloprotokołowa, 308
 - z klientem, 41, 49
- kontrola
 - administracyjna, 227
 - czasu pracy, 58
 - fizyczna, 228
 - korygująca, 228
 - monitorująca, 228
 - postępu prac, 36
 - prewencyjna, 228
 - techniczna, 227
- kontrole bezpieczeństwa, 226
- kontroler frontowy, 155
- koordynowanie
 - działań, 42
 - prac zespołu, 59
 - projektu, 35
- kopia
 - przyrostowa, 206
 - tekstowa, 206
 - w innej lokalizacji, 206
 - zapasowa, backup copy, 133, 249, 334, 361, 429
- kopie bezpieczeństwa, 204
- kopiowanie cudzych treści, 289

korespondencja masowa, 211

- adresowanie, 215
- komponenty, 211
- ujawnianie danych, 216
- zagrożenia, 214
- zgody, 213

koszty

- hostingu, 166, 167
- modyfikacji, 111
- prac dodatkowych, 37
- realizacji projektu, 95
- utrzymania aplikacji, 276
- wdrożenia, 111

L

lepkie sesje, sticky sessions, 345

licencja

- Apache, 110
- BSD, 110
- GPL, 110
- GPLv2, 114
- LGPL, 110
- Magento, 113
- MIT/X11, 110
- OSL, 285, 286
- OSL2, 114

licencje

- Copyleft, 109
- ograniczenia, 109, 115
- uznanie autorstwa, 109
- zestawienie, 110, 111

licencje open source, 105, 109, 283

liczba

- połączeń z domeną, 406
- zadań HTTP, 400, 404

limity transferu, 175

lista

- licencji open source, 109
- odnośników, 388
- zaufanych certyfikatów, 464

listy kontrolne, checklists, 136, 147

load balancer, 333, 345, 373

logika biznesowa, 96

Ł

łącza internetowe, 162

łączenie obrazów, 401

M

macierz

- cech i funkcjonalności, 84
- dyskowa, 335
- dysków, 200
- NetAPP FAS, 349
- RAID, 362
- RAID10, 349

makieta, 84, 86

- high fidelity, 85, 90, 92
- low fidelity, 85, 90
- interfejsu użytkownika, 80

makiety funkcjonalne, 44

Manifest Agile, 63

mapa

- myśli, 73
- serwisu, 388

mass market hoster, 166

MDM, Master Data Management, 297

mechanizm

- Application Cache, 409
- banerowy, 453
- CAPTCHA, 213, 381
- DKIM, 215
- Google Rich Snippets, 390, 393, 395
- Media Queries, 416
- SPF, 215
- szablonów, 149
- Time Machine, 204
- typu sandbox, 218
- unieważniania cache, 355
- wtyczek, 149
- zdarzeń, 149

metoda

- MoSCoW, 74
- top-down, 155
- Zarządzania Łańcuchem Krytycznym, 62

metodologia Agile, 77

metody

- dostawy, 454
- integracji, 449
- komunikacji, 311
- płatności, 454
- zarządzania zmianą, 75

metodyka, 41

- P3O, 61
- PMBOK, 61

- PMO, 61
- Prince2, 59–62
- metodyka zwinna, 41, 57, 79
 - Crystal Clear, 63
 - Kanban, 65
 - Scrum, 61–63
 - XP, 63
- mierzenie sukcesu, 44
- migawka, snapshot, 206, 337, 365
- migracja
 - danych, 134, 450
 - logiki biznesowej, 191
 - warstwy danych, 191
 - warstwy prezentacji, 191
- mikrodane, 390, 394–398
- mikroformat hCard, 395
- minimalizowanie
 - rozmiaru plików, 401
 - występowania SPoF, 326
- model
 - black box, 426
 - EAV, 119
 - opt-in, 213
 - opt-out, 213
 - zdarzeniowy, 118
- modularność plików CSS, 407
- moduł
 - CSS3 Gradients, 403
 - mod_pagespeed, 357
- modyfikacje projektu funkcjonalnego, 87
- monitoring
 - aplikacji, 430
 - rozproszony geograficznie, 458
- monitorowanie, 204
 - dostępności aplikacji, 457
 - komunikacji, 314, 316
 - na poziomie aplikacji, 317
 - na poziomie infrastruktury, 319
 - na poziomie systemowym, 319
 - procesu zakupowego, 458
 - serwerów, 354
- MVC, Model-View-Controller, 118, 341
- MVP, Minimum Viable Product, 77

N

- nadmiarowość, *Patrz* redundancja
- nagłówki
 - Accept-Encoding, 403
 - Content-Encoding, 403

- ETag, 405
- Expires, 405
- Last-Modified, 405
- nagłówki, 378, 386
- narzędzia
 - do automatycznego testowania, 221, 222
 - do ciągłego testowania, 224
 - do monitoringu, 459
 - do śledzenia błędów, 225
 - do wykonywania kopii, 365
 - Google Closure Tools, 401
 - Google Webmaster Tools, 390, 454
 - typu gettext, 119
 - zwiększające skalowalność, 351, 352
- narzędzie, *Patrz* program
- NAS, Network Attached Storage, 335, 341
- nazwy klas, 402
- niekompletność specyfikacji, 263
- nieuczciwa konkurencja, 290
- niezgodność oprogramowania, 263
- normy ISO, 164
- nośniki danych, 363
- notatka podsumowująca, 36, 49
- numer EAN, 312

O

- OAGIS, 299
- OAT, Operational Acceptance Testing, 144
- obciążenie systemu, 179
- obowiązki wierzyciela, 253
- obsługa zdarzeń, 412
- ocena testu, 437
- ochrona
 - bazy adresowej, 214
 - danych osobowych, 162, 165, 245, 249
 - infrastruktury technicznej, 371
 - kodu źródłowego, 289
 - przed spamem, 212
 - strategii biznesowej, 290
 - wielu domen, 464
- odbior jednostronny, 274
- odgałęzienie, branch, 328
- odnośniki, 378
- odpowiedzialność
 - wykonawcy, 255
 - wykroczeniowa, 293
- odstąpienie od umowy, 272

- odszkodowanie, 291
 - odtwarzanie
 - kopii, 366
 - po awarii, 368, 371
 - określanie
 - przedmiotu umowy, 42
 - zakresu prac, 35
 - opakowywanie, encapsulation, 208
 - opis
 - architektury, 97
 - architektury aplikacji, 153
 - do makiet, 90
 - działania algorytmów, 153
 - działania funkcji, 44
 - funkcjonalności, 45, 96
 - interakcji, 80
 - modułów produktu, 96
 - najważniejszych klas, 153
 - Persony, 83
 - procedury instalacji, 153
 - procesu wymiany danych, 300
 - procesów biznesowych, 44
 - przypadków użycia, 45, 74, 80
 - scenariuszy testowych, 448
 - schematu przekazywanych danych, 300
 - strony, 387
 - struktury bazy danych, 153
 - wymagań, 80
 - opóźnienia, 38, 43, 47, 260, 265
 - oprogramowania
 - aktualizacje, 128
 - architektura, 118
 - błędy, 137
 - dokumentacja, 117, 119
 - historia zmian, 120
 - podatność na wirusy, 112
 - podział na warstwy, 118
 - porównanie, 112, 113
 - poziom zależności, 124
 - rozwój aplikacji, 113, 115, 121, 277, 281
 - skalowalność, 112
 - testy automatyczne, 120
 - wdrożenie, 112
 - WebAPI, 119
 - wersje językowe, 119
 - wsparcie, 112
 - wsparcie producenta, 116, 117
 - zasada DRY, 124
 - oprogramowanie
 - dedykowane, 112
 - model czterofazowy, 143
 - model trójfazowy, 142
 - open source, 105, 108, 283
 - produkcyjne, 142
 - rozwojowe, 142
 - pudełkowe
 - otwarte, 103, 105, 107, 112
 - w formie dzierżawy, 105, 112
 - zamknięte, 103, 105, 112
 - testowe, 142
 - wirtualizacyjne, 178
 - High availability, 180
 - KVM, 181
 - Live Migrations, 179
 - Microsoft Hyper-V, 181
 - OpenVZ, 181
 - Private cloud, 180
 - Snapshots, 179
 - VirtualBox, 182
 - Virtualization ROI Calculator, 182
 - VMWare ESX, 181
 - Xen, 181
 - optymalizacja
 - dla wyszukiwarek, 375
 - kosztów, 188
 - SEO, 385
 - szybkości ładowania witryn, 408
 - organizacja wiedzy, 129
 - ORM, Object-Relational-Mapper, 98, 358
 - OSL, Open Source Licence, 285
 - osprzęt sieciowy, 200
 - otwartość na zmiany, 55
 - oznaczenia, tags, 155
- P**
- P3O, Portfolio, Programme and Project Offices, 61
 - PaaS, Platform as a Service, 338, 349
 - pamięć
 - lokalna urządzenia, 409
 - podręczna, cache, 342, 345, 355
 - podręczna przeglądarki, 404
 - parametry
 - biznesowe, 369
 - certyfikatów SSL, 464, 466
 - dostępności, 325

- partycjonowanie, 347
- PDA, Personal Data Assistant, 410
- peer code review, 55
- Persona, 82
- Pivotal Tracker, 49
- plan
 - ciągłości działania, 367, 373
 - komunikacji z klientem, 47
 - odtworzenia po awarii, 373
- planowanie, 185
- planowanie rozwoju oprogramowania, 127
- platforma
 - dedykowana, 104
 - jako usługa, 349
 - Windows Azure, 193
 - wirtualizacyjna, 178
 - zintegrowana, 296
- platformy e-commerce
 - Gekosale, 107
 - IAI-Shop, 114
 - Magento, 107, 113
 - osCommerce, 114
 - Prestashop, 113, 107
 - Shoper, 114
 - WellCommerce, 114
- platformy wymiany danych
 - IBM WebSphere ESB, 304
 - Microsoft Biztalk Server, 304
 - Mule ESB, 303
 - Oracle Enterprise Service Bus, 304
- plik
 - robots.txt, 388
 - sitemap.xml, 388, 454
 - sitemap.xml.gz, 388
- pliki
 - *.css, 401, 406, 408
 - *.js, 401, 407–409
 - Cookies, 441
 - CSV, 314, 317, 318
 - GIF, 404
 - graficzne, 403
 - JPEG, 403
 - PNG, 404
 - statyczne, 356
 - XML, 312
 - zewnętrzne, 408, 419
- plugin, 150, 284, 286
- plynny układ strony, 417
- PMBOK, 62
- PMO, Project Management Office, 61
- podpisywanie nagłówków, 215
- podpowiedź, 379
- podsieci, 202
- podsluchiwanie, 209
- podwykonawca, 39
- podział aplikacji na warstwy, 341
- pojedynczy punkt awarii, 200, 326, 332
- polecenie curl, 449
- polityka czystego
 - biurka, 229
 - ekranu, 229
- połączenie szyfrowane, 248
- poprawki
 - bezpieczeństwa, 150
 - błędów krytycznych, 150
- porównanie oprogramowania, 112, 113
- portfolio wdrożeń, 117
- potrzeby klienta, 88
- potwierdzanie operacji, 311
- powiązania między produktami, 452
- poziom jakości usług, 162
- pozycjonowanie, 454
- prawa zależne, 279
- prawo autorskie, 268–280
- prawo odstąpienia od umowy, 266
- preprocesor CSS, 407
- prędkość ładowania
 - podstron, 392
 - stron, 405
- Prince2, 61
- problemy komunikacyjne, 38
- procedura Charette, 78
- procedury składowane, 199
- proces
 - krytyczny, 367, 369
 - zarządzania zmianą, 75
- procesy
 - biznesowe, 369
 - pomocnicze, 367
 - zarządzania
 - w PMBOK, 62
 - w Prince2, 61

program

ActiveCollab, 43
 Adobe Edge Inspect, 420
 Adobe Edge Reflow, 420
 Adobe Photoshop, 404
 Apache Benchmark, 430, 444
 Apache JMeter, 440
 APC, 342
 Axure, 85
 Basecamp, 43, 58
 Collectd, 343, 360
 Cucumber, 428
 Ettercap, 223
 FastCGI, 357
 Google Docs, 136
 Google PageSpeed Insights, 408
 HAProxy, 333, 359
 Httpperf, 442, 444
 HttpPerf, 443
 Incinga, 343
 jpegtran, 403
 JMeter, 430, 440
 JSMIn, 401
 Magento Commerce, 348
 Memcached, 360
 Microsoft Project, 47
 mod_pagespeed, 357
 mod_php, 357
 PageSpeed, 360, 392
 Parallels, 337
 Paros Proxy, 223
 Perlball, 333
 PHP-FPM, 357
 pngcrush, 404
 PNGGauntlet, 404
 Redmine, 43, 58
 Rich Snippet Tool, 398
 rsync, 204
 Selenium, 428
 Siege, 441–444
 Skipfish, 426
 SOACleaner, 449
 SProxy, 441
 Squid, 356
 Toggl, 58
 Varnish, 333, 356
 VMWare, 337

Wiki, 43

Wireshark, 223

Yahoo! YUI Compressor, 401

Zend Optimizer, 342

programowanie w parach, 138

programy pochodne, 285

project manager, 34, 39, 56–59

projekt, 69

funkcjonalny, 81, 87

interfejsu użytkownika, 44

projektowanie

doświadczeń użytkownika, 89

interakcji, 88

pod urządzenia mobilne, 412

protokołu komunikacji, 307

responsywne, 416, 420

skupione na potrzebach, 88

UCD, 88

protokoły

autorskie, 209

binarne, 301

kommunikacyjne, 207, 306

tunelujące, 209

międzysieciowe, 198

odbioru, 273

TCP/IP, 208

transmisji danych, 198

protokół

BGP, 373

EDI, 307

ESMTP, 212

HTTPS, 209, 248

IMAP, 307

IP, 208, 305

KNX, 306

L2TP, 305

OAGIS, 307

odbioru prac, 37, 46

Open Graph, 391

PPTP, 305

REST, 301

RTP, 308

SIP, 308

SMTP, 212

SOAP, 301

XMPP, 307

prototypy strony WWW, 84
provider, 166
przedmiot umowy, 42
przeglądanie kodu, 126
przeglądarki, 383, 451
przeglądarki mobilne, 383, 409
przełączanie ruchu, 372
przełącznik, 201
przepływ danych, 448
przepustowość łącza, 163, 175, 414
przesyłanie danych, 315
przetwarzanie danych, 198–205
przetwarzanie danych osobowych, 249
przezroczystość, 404
przypadki użycia, use cases, 73, 96
przywracanie systemu, 429
pseudoklasa
 :focus, 380
 :hover, 381
pseudoklasy języka CSS, 380
publiczna beta, 188
punkt przywrócenia do działania, 370

Q

QA, Quality Assurance, 49, 129

R

RAID, 200, 335, 362
ranking wyszukiwania, 385, 392
raportowanie, 171, 221
redukcja ryzyka, 371
redundancja, 326
 serwerów aplikacji, 332
 sprzętowa, 332
refaktoryzacja, 120, 124–126
reguły komunikacji, 59
rejestrwanie zdarzeń, 318
relacje, 308
Release Manager, 140
replikacja bazy danych, 206, 333, 341, 348
repozytorium GitHub, 49
Responsive Web Design, 421
 CSS3 Media Queries, 416
 elastyczne zdjęcia, 416
 elastyczny układ strony, 416
REST, Representational State Transfer, 301
retest, 46

rękojmia, 232, 234
rodzaje
 testów, 219
 wad, 256
ROI, Revenue On Investment, 75
rola, 379
roszczenia regresowe, 271
rozdzielanie ruchu, 345
rozdzielczość ekranu, 414
rozgałęzienia, branches, 155
rozliczanie time-material, 46
rozmowa z klientem, 35
rozproszona architektura bazodanowa, 353
rozproszone systemy plików, 335
rozszerzone wyniki wyszukiwania, 390
rozwój
 aplikacji, 54, 113, 121, 127, 277, 281
 nieznanego systemu, 152, 156
 systemu, 156
 zespołu technologicznego, 50, 53
równoważenie obciążenia serwerów, 189, 333
RPO, Recovery Point Objective, 173, 369
RTO, Recovery Time Objective, 173, 369
RTP, Real-time Transport Protocol, 308
ruch organiczny, 385
RUP, 41
ryzyko, 95, 371

S

SaaS, Software as a Service, 106, 114, 166, 187, 349, 363, 458
scalenie zmian, 150
scenariusz
 integracji, 448
 logowania i rejestracji, 439
 oglądania produktów, 439
scenariusze
 testowe, 45, 428, 432, 437
 użycia, 83
schemat
 Business Model Canvas, 83
 przepływu danych, 447
schematy bazy danych, 96
SCM, Supply Chain Management, 297
Scrum, 41, 61, 63
 analiza sprintu, 64
 codzienne spotkania, 64

- Scrum
 - planowanie, 64
 - przeгляд sprintu, 64
 - scrum master, 63
 - wdrożenie metodyki, 64
 - właściciel produktu, 63
 - zespół deweloperski, 63
- Scrumban, 67
- sekcja Errors Section, 443
- selektory CSS, 401, 402
- semantyczny kod HTML, 401
- SEO, Search Engine optimization, 375, 385, 389, 393, 453
- SERP, Search Engine Results Page, 385
- serwer, 200
 - aplikacji, 341, 358
 - bazy danych, 341, 348, 357
 - cache, 341
 - CI, 141
 - ciągłej integracji, 139
 - DNS, 372
 - FTP, 301
 - hostingowy, 161
 - kolejkowania, 345
 - Lighthtpd, 349
 - Memcached, 355
 - middleware, 341
 - NGinx, 349
 - plików statycznych, 356, 358
 - poczty elektronicznej, 212
 - podstawowy, 372
 - pomostowy, 144
 - produkcyjny, 429
 - Proxy, 356, 441
 - raportów zdarzeń, 204
 - stagingowy, 49
 - VPS, 177
 - wirtualny, 165
 - zapasowy, 372
- serwis
- Adobe BrowserLab, 459
- serwis
 - CloudFlare, 371
 - CrossBrowserTesting.com, 459
 - gemiusRanking, 451
 - Incapsula, 371
 - internetowy, 184
 - Litmus, 459
 - mobilny, 421
- sesje, 358
- sezonowe wahania, 185, 186
- sieci
 - CDN, 349, 353
 - typu cloud, 353
 - VPN, 304
 - wirtualne, 179
- sieciowy systemu plików, 349
- sieć dystrybucji
 - plików, 349
 - treści, 405
- silnik bazodanowy, 334
- SIP, Session Initiation Protocol, 308
- skalowalnie, 184, 323, 339, 354
 - aplikacji, 344
 - bazy danych, 348
 - kontenera, 418
 - na warstwie sprzętowej, 343
 - pionowe, 343, 346
 - poziome, 344, 346
 - strony, 416
 - warstwy bazy danych, 346
 - warstwy przechowującej pliki, 349
 - zdjęcia, 418
- skanowanie antywirusowe, 224
- składowanie plików, 341
- skrypty JavaScript, 407
- skrzynka pułapka, 214
- SLA, Service Level Agreement, 16, 162, 169, 190, 325, 371, 457
- słownik schema.org, 390
- snapshot, 206
- SOA, Service Oriented Architecture, 304
- SOAP, Simple Object Access Protocol, 119, 301
- spam, 212, 215
- specyfikacja, 87, 263, 267
- specyfikacja do makiet, 86
- spis dobrych praktyk, 376
- SPoF, Single Point of Failure, 326, 332, 334
- spory pomiędzy stronami, 263
- sposób wdrażania, 146
- sprawdzanie skuteczności kampanii, 216
- SSD, Solid State Disk, 343
- staging, 49

- standard
 - HTML5, 379
 - POSIX, 335
 - schema.org, 390, 394
 - storage, 341
 - stosowanie refaktoryzacji, 124
 - Strategia Zarządzania Komunikacją, 59
 - strefa DNS, 215
 - strony
 - responsywne, 415
 - statyczne, 452
 - syndrom vendor lock-in, 190
 - system
 - do zarządzania projektem, 49
 - dystrybucji zasobów, 405
 - ERP, 446
 - FK, 455
 - gromadzenia wiedzy, 42
 - informatyczny, 43, 232
 - dokumentacja, 249
 - instrukcja zarządzania, 250
 - warunki gwarancji, 235
 - wdrożenie, 233
 - kontroli wersji, 49, 120, 152, 157
 - Git, 141, 328
 - Mercurial, 141
 - Subversion, 328
 - kontroli zadań
 - Basecamp, 80
 - Redmine, 80
 - kontroli zgłoszeń, 46
 - księgowy, 296, 446
 - magazynowy, 302, 446
 - plików
 - Ceph, 336
 - GlusterFS, 336
 - HadoopFS, 336
 - Lustre, 336
 - MogileFS, 336
 - ticketowy, 43, 48, 58, 129
 - Redmine, 120, 152
 - zgłaszania uwag, 155
 - systemy
 - ciągnione, 65
 - IDS/IPS, 203
 - operacyjne, 202, 204
 - pchane, 65
 - rozproszone, 335
 - śledzenia błędów, 225
 - sytuacje wyjątkowe, 427
 - szablony e-maili, 453
 - szkody rzeczywiste, 271
 - szybkość ładowania
 - stron, 399, 406
 - witryn, 408
 - szyfrowanie
 - danych, 460
 - wolumenów dyskowych, 212
 - szyna wymiany danych, 302
- ## Ś
- ścieżka
 - klienta, 439
 - zakupowa, 451
 - ścieżki
 - funkcjonalne, 432
 - okruszków, breadcrumbs, 452
 - śledzenie
 - błędów, 225
 - czasu pracy, 58
 - historii zmian, 150
 - postępów prac, 65
 - zgłoszeń błędów, *Patrz* system ticketowy
 - zmian, 157
 - środowisko testowe, 217
- ## T
- tabela kar umownych, 242
 - tabele pośrednie, 449
 - technika CSS Sprites, 400
 - techniki
 - zbierania wymagań, 72
 - zwinne, Agile, 79
 - technologia VoIP, 308
 - technologie wirtualizacji, 181
 - termin realizacji, 38
 - tester, 46
 - testy, 138
 - akceptacyjne, 188
 - automatyczne, 119, 125, 140, 220, 428
 - behawioralne, 440
 - bezpieczeństwa, 217, 426, 451
 - czarnoskrzynkowe, 432
 - dymne, 428

testy

- danych strukturalnych, 396
- dokumentacji, 100
- funkcjonalne, 127, 428, 449
- integracyjne, 430, 446–449
- jednostkowe, 125, 133, 156, 449
- metod HTTP, 441
- obciążeniowe, 430
- obsługi sytuacji wyjątkowych, 427
- oprogramowania, 45, 55, 425
 - scenariusz, 432–437
 - zakres testów, 433
 - założenia, 434
- podstawowych scenariuszy, 428
- przywracania systemu, 429
- regresyjne, 428
- ręczne, 222
- serwisu, 382, 459
- systemu, 151
- użyteczności interfejsu, 425
- wewnętrzne, 219
- wydajnościowe, 188, 224, 438
- witryny, 383
- z użytkownikami, 85
- zewnętrzne, 219

ticket, 43

transakcyjność, 328

transfer ryzyka, 371

transmisja danych, 198, 209

treści alternatywne, 377

tryb

- monitorowania systemu, 171
- renderowania, 383
- serwisowy, 146
- zgłaszania problemów, 172

tunel

- IPsec, 305
- TLS/SSL VPN, 305

tworzenie

- katalogu usług, 169
- kopii, 154
- kopii w trybie ciągłym, 173
- kopii zapasowych, 361
- makiety interaktywnej, 85
- migawek, 206, 337
- projektu funkcjonalnego, 82
- wycieczek, 118, 286

zrzutów, 206

typy

- certyfikatów SSL, 460
- danych, 310
- danych strukturalnych, 390
- mediów, 419
- relacji, 308
- sieci VPN, 305

tytuł

- odnośnika, 389
- strony, 387

U

UAT, User Acceptance Testing, 144

UCD, User Centered Design, 88

UML, 74, 96

umowa, 34, 37, 42

depozytowa, 278

powierzenia przetwarzania danych osobowych, 251, 456

SLA, 169

utrzymaniowa, 235, 238

unikanie sporu, 264

urządzenia

- aktywne, 200
- mobilne, 410–412
- NAS, 335

urządzenie typu load balancer, 373

usługa

- BrowserShots, 383
- CDN, 167
- Google Analytics, 390
- Memcached, 345
- serwisowa, 278
- utrzymaniowa, maintenance, 276
- WCF, 301
- Web Service, 301, 449

ustawienie nagłówków, 355

usterki, 220

utracone korzyści, 271

utrzymanie systemu, 46, 98, 238

uwierzytelnianie, 209, 248, 312

UX, user experience, 88

użyteczność, Usability, 87, 411

interfejsów użytkownika, 426

stron internetowych, 87

V

VLAN, 201
VPN, Virtual Private Network, 179, 304
VPS, Virtual Private Server, 165, 177

W

wady

- fizyczne, 256
- prawne, 256, 267
- projektu, 262, 265
- rozwiązań komercyjnych, 106

walidator, 382, 386

warstwa

- aplikacji, 331, 358
- backend, 399
- interakcji, 408
- Kontrolera, 341
- Modelu, 341
- prezentacji, 408
- sieciowa, 331
- sprzętowa, 331
- treści, 408
- widoku, 341

wąskie gardła, 66, 342

WCF, Windows Communication Foundation, 301

wdrożenie, 145, 149

- aplikacji, 269
- metodyki Scrum, 64
- systemu informatycznego, 41

WebAPI, 119, 149

wersja mobilna serwisu, 410

wersje projektu, 129–132, 139–141

weryfikacja

- cech niefunkcjonalnych, 433
- certyfikatu, 464
- dostępu do serwera, 462
- projektu, 36

widoczność produktu, 393

widok rozszerzony, 393

wina

- nieumyślna, 269
- umyślna, 269

WIP, Work in Progress, 65

wirtualizacja, 176, 336

- serwerów, 176
- stacji roboczych, 177

własność intelektualna, 287

właściciel produktu, Product Owner, 63, 71

WML, Wireless Markup Language, 411

WMS, Warehouse Management System, 297

wprowadzanie poprawek, 45

wsparcie techniczne, 190

współpraca

- międzydziałowa, 51
- z podwykonawcami, 39
- z zespołem, 56, 59

wybór

- celów monitorowania, 316
- certyfikatu, 463
- dostawcy hostingu, 161
- platformy sklepowej, 101–120
- testów, 223

wycena bazy danych

- liczenie konwersji, 292
- metoda odtworzeniowa, 291
- ustalanie wartości rynkowej, 292

wyciek danych, data leak, 212

wydajność, 354

- serwerów, 163
- sklepu internetowego, 438
- skryptów, 409

wykrywanie

- uszkodzonych serwerów, 333
- włamań, 203
- wycieku danych adresowych, 215

wyliczanie odszkodowań, 291

wymagania

- biznesowe, 70, 77
- funkcjonalne, 81, 99, 234
- klienta, 41
- niefunkcjonalne, 234
- systemowe, 70, 74
- techniczne, 98

wymiana

- danych, 298
- wiedzy, 54

wymuszanie prac dodatkowych, 39

wyniki testów, 443–445

wysoka dostępność, High Availability, 324, 328

wysyłanie

- korrespondencji, 211
- ofert i reklam, 213

wyszukiwanie pełnotekstowe, 348

wyszukiwarki, 390

wyświetlanie komunikatów, 381
 wytwarzanie oprogramowania, 66
 wyznaczniki jakościowe, 155
 wzorce pracy serwisów, 184
 wzorzec
 IoC, 341
 MVC, 341
 projektowy, 97

X

XMPP, Extensible Messaging and Presence Protocol, 307
 XP, Extreme programming, 63
 xSS, Cross Site Scripting, 426

Z

zaangażowanie klienta, 47
 zabezpieczenia, 226
 fizyczne, 161
 przed roszczeniami, 38
 techniczne, 161
 transmisji, 248
 własności intelektualnej, 287
 zadania
 klienta, 35, 47
 project managera, 57, 59
 zagrożenia dla infrastruktury, 370
 zakres
 prac, 35
 przesyłanych danych, 299
 testów, 433
 zapewnianie
 bezpieczeństwa danych, 252
 jakości, quality assurance, 135–138
 zapisy o pracach dodatkowych, 37
 zapisywanie ustaleń, 44
 zapytania mediów, Media Queries, 419
 zarządca wirtualizacji, Hypervisor, 337
 zarządzane przełączniki sieciowe, 203
 zarządzanie
 ciągłością działania, 368
 ludźmi, 59
 portfelami projektów, 61
 projektami, Projekt Management, 33, 61
 ryzykiem, 95
 wersjami, 151
 zgłoszeniami, 129
 zmianą, Change Management, 75, 79

zasada DRY, 124
 zasobniki usług, 302
 zasoby, 400
 zasoby rozproszone, 405
 zastosowanie certyfikatów, 463
 zbieranie wymagań, 70, 96
 analiza dokumentacji, 73
 analiza procesów biznesowych, 73
 burze mózgów, 72
 obserwacja, 73
 prototypowanie, 73
 warsztaty, 73
 wywiad, 73
 zbiór danych osobowych, 246
 zespół
 deweloperski, 63
 technologiczny, 50, 54
 wdrożeniowy, 56
 zestawienie licencji, 110, 111
 zewnętrzne składniki aplikacji, 267
 zgłoszenia błędów, 45
 zmiana platformy sklepowej, 450
 bezpieczeństwo, 451
 certyfikat SSL, 456
 Giodo, 456
 integracje, 455
 katalog produktów, 452
 metody dostawy, 454
 metody płatności, 454
 SEO, 453
 testy, 450
 treści statyczne, 452
 ustawienia poczty, 455
 wydajność, 451
 zmiany
 funkcjonalne, 126
 w oprogramowaniu, 235
 zmienianie
 parametrów środowiska, 167
 wersji, 150
 wymagań, 74–76
 zmniejszanie transferu danych, 400
 znacznik
 </body>, 407
 <a>, 389
 <fieldset>, 380
 , 390, 418

- <label>, 379
- <legend>, 380
- <link>, 408
- <meta name="description">, 387
- <script>, 408
- <title>, 387
- znaczniki
 - HTML, 378
 - kontrolne, 318
 - semantyczne, 378, 386
- zrzut, dump, 206
- zwinne wytwarzanie oprogramowania, 63
- zwrot z inwestycji, 182

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>

e-handel

w pytaniach i odpowiedziach

Nadzór nad procesem tworzenia i utrzymaniem sklepu internetowego lub dowolnej innej formy obecności firmy na rynku e-commerce jest naprawdę trudnym zadaniem. Wymaga ogromnej wiedzy i znacznego doświadczenia w kwestiach biznesowych i prawnych, a przede wszystkim technicznych. Do pracy nad tą książką Piotr Karwatka zaprosił grono uznanych ekspertów: dyrektorów IT oraz niezależnych konsultantów, którzy zajmują się m.in. wdrożeniami, audytami oraz utrzymaniem produkcyjnym zaawansowanych sklepów internetowych. Znajdziesz tutaj odpowiedzi na kluczowe pytania dotyczące całego spektrum technicznych zagadnień związanych z handlem w internecie.

Technologia w e-commerce... wypełnia ważną niszę na rynku wydawniczym, ponieważ proponuje kompleksowe podejście do zagadnień, które do tej pory były omawiane osobno w ramach literatury informatycznej i biznesowej. Większość dostępnych dziś książek skupia się albo na technicznych aspektach e-handlu (sposobach korzystania z baz danych, języków programowania i frameworków), albo na ujęciu biznesowym. Autorzy niniejszej publikacji starali się połączyć te dwa światy i stworzyć swoiste FAQ e-biznesu. Jeśli wciąż masz wątpliwości, czy dobrze prowadzisz swój e-biznes, sięgnij po tę pozycję. Tak cennej wiedzy nie znajdziesz nigdzie indziej!

Z tej książki dowiesz się między innymi:

- / jak dokonać dobrego wyboru platformy sklepowej i na jakie aspekty zwrócić uwagę
- / w jaki sposób zintegrować się z systemami ERP, stworzyć szynę wymiany danych oraz umiejscowić system w obecnym środowisku informatycznym
- / jak poradzić sobie z dużym obciążeniem i wykorzystać skalowalność oraz jakich narzędzi użyć
- / jak zapewnić bezpieczeństwo danych i nie narazić się GODO
- / jak stworzyć procedury w zespole wdrożeniowym i przestrzegać tych procedur
- / jak dbać o jakość projektów od etapu analitycznego po wdrożenie i utrzymanie
- / jak zapewnić wysoką dostępność aplikacji, SLA oraz wydajny hosting i wirtualizację

Patroni medialni:



MAM
STARTUP

NowyMarketing

eKomercyjnie.pl

książkiklasybusiness

Nr katalogowy: 13865

Księgarnia Internetowa:
<http://onepress.pl>

Zamówienia telefoniczne:
0 801 339900

0 601 339900

one
p r e s s

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

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

cena: 79,00 zł

ISBN 978-83-246-7541-8



9 788324 675418