

O autorze (9)

O redaktorach merytorycznych (11)

Wstęp (13)

Zawartość książki (14)

Narzędzia potrzebne do realizacji przykładów (15)

Do kogo kierowana jest książka? (15)

Konwencje typograficzne (16)

Przykładowy kod (16)

Rozdział 1. Docenić Node (17)

Unikatowo zaprojektowany Node (18)

Współbieżność (20)

Równoległość i wątki (21)

Współbieżność i procesy (23)

Zdarzenia (24)

Pętla zdarzeń (26)

Wpływ sposobu zaprojektowania Node na architektów systemów (30)

Budowanie większych systemów z mniejszych (30)

Strumienie (32)

Wykorzystanie JavaScript na każdym etapie projektu maksymalizuje dobre efekty (33)

Zamiana kodu (34)

Browserify (34)

Podsumowanie (35)

Rozdział 2. Instalacja i wirtualizacja serwerów Node (37)

Uruchamianie podstawowego serwera Node (38)

Witaj, świecie (38)

Tworzenie żądań HTTP (39)

Pośredniczenie i tunelowanie (40)

HTTPS, TLS (SSL) i zabezpieczanie serwera (42)

Tworzenie podpisanego samodzielnie certyfikatu na potrzeby programistyczne (42)

Instalacja prawdziwego certyfikatu SSL (43)

Instalacja aplikacji w chmurze Heroku (44)

Dodatki (46)

Git (46)

Zarządzanie zmiennymi konfiguracyjnymi (49)

Zarządzanie wdrożeniem (50)

Instalacja aplikacji w chmurze OpenShift (51)

Instalacja aplikacji Node i MongoDB (52)

Wdrożenie aplikacji (52)

Docker, czyli lekkie i wirtualne kontenery (54)

Najpierw jednak łuk systemu Unix (56)

Korzystanie z systemu Docker (57)

Tworzenie pliku Dockerfile (58)

Zbudowanie i uruchomienie obrazu Dockera (59)

Podsumowanie (61)

Rozdział 3. Skalowanie Node (63)

Skalowanie pionowe wykorzystujące wiele rdzeni procesora (64)

Funkcja spawn() (65)

Funkcja fork() (68)

Funkcja exec() (70)

Funkcja execFile() (71)

Komunikacja z procesem potomnym (71)

Moduł cluster (73)

Skalowanie w poziomie z wykorzystaniem wielu maszyn (77)

Użycie serwera Nginx (77)

Równoważenie obciążenia za pomocą Node (85)

Użycie kolejek komunikatów (87)

Użycie modułu komunikacji UDP serwera Node (91)

Podsumowanie (96)

Rozdział 4. Zarządzanie pamięcią i przestrzenią (97)

Jak poradzić sobie z dużym tłumem? (98)

Mikroserwisy (98)

Mechanizm pub-sub serwera Redis (99)

Mikroserwisy w narzędziu Seneca (102)

Zmniejszenie zużycia pamięci (105)

Używaj strumieni zamiast buforów (106)

Prototypy (107)

Wydajne pamięciowo struktury danych z wykorzystaniem serwera Redis (109)

V8 i optymalizacja wydajności (116)

Optymalizacja kodu JavaScript (116)

Strategie dotyczące pamięci podręcznych (121)

Wykorzystanie Redis jako pamięci podręcznej (121)

Wdrażanie CloudFlare jako CDN (125)

Zarządzanie sesją użytkownika (127)

Uwierzytelnianie i sesje JWT (127)

Podsumowanie (131)

Rozdział 5. Monitorowanie aplikacji (133)

Jak poradzić sobie z błędami? (134)

Moduł domain (135)

Wyłapywanie błędów innych procesów (137)

Tworzenie dzienników zdarzeń (140)

Dostosowywanie działania w zmieniającym się środowisku (144)

REPL narzędzia Node (144)

Zdalny monitoring i zarządzanie procesem Node (146)

Profilowanie procesów (149)

Wykorzystywanie niezależnych narzędzi do monitorowania (152)

PM2 (152)

Wykorzystanie New Relic do monitorowania aplikacji (156)

Podsumowanie (158)

Rozdział 6. Budowanie i testowanie (159)

Budowanie aplikacji za pomocą narzędzi Gulp, Browserify i Handlebars (160)

Narzędzie Gulp (160)

Wykonanie szkieletu systemu budowania (163)

Uruchamianie i testowanie zbudowanej wersji (169)

Wykorzystanie narzędzi do testowania wbudowanych w system Node (171)

Debugger wbudowany w system Node	(172)
Moduł assert	(175)
Testowanie za pomocą Mocha, Chai, Sinon i npm	(177)
Mocha	(178)
Chai	(180)
Sinon	(181)
Automatyzacja testów w przeglądarce internetowej dzięki PhantomJS i CasperJS	(187)
Testowanie z wykorzystaniem PhantomJS	(187)
Scenariusze nawigacyjne w CasperJS	(189)
Podsumowanie	(192)
 Rozdział 7. Wdrażanie i konserwacja	(193)
GitHub i mechanizm webhook	(194)
Włączenie mechanizmu webhook	(195)
Implementacja systemu budowania i wdrażania wykorzystującego mechanizm webhook	(198)
Synchronizacja wersji lokalnej i zdalnej	(201)
Tworzenie wersji lokalnej za pomocą narzędzia Vagrant	(201)
Przygotowywanie systemu przy użyciu Ansible	(205)
Integracja, dostarczanie i wdrażanie	(208)
Ciągła integracja	(208)
Ciągłe dostarczanie	(208)
Ciągłe wdrażanie	(209)
Budowanie i wdrażanie z użyciem narzędzia Jenkins	(209)
Wdrażanie w chmurze Heroku	(213)
Zarządzanie pakietami	(216)
Wersjonowanie semantyczne	(216)
Zarządzanie pakietami z użyciem npm	(218)
 Skorowidz	(225)