

[Helion.pl](#) » [Hardware](#) » [Inne](#)

Codziennie inna książka 30% taniej

Internet rzeczy. Budowa sieci z wykorzystaniem technologii webowych i Raspberry Pi

Autorzy:

Dominique Guinard, Vlad Trifa

Okładka książki Internet rzeczy. Budowa sieci z wykorzystaniem technologii webowych i Raspberry Pi

Obróć

Zajrzyj do książki

Wydawnictwo:

Helion

Ocena:

Bądź pierwszym, który oceni tę książkę

Stron:

384

Druk:

oprawa miękka

Książka

67,00 zł

Dodaj do koszyka Wysyłamy w 24h

Ebook

67,00 zł

53,60 zł

Dodaj do koszyka lub Kup 1-kliknięciem

Przenieś na półkę

Do przechowalni

[Opis książki](#)

[Promocje i pakiety](#)

[Często kupowane razem](#)

[Szczegóły książki](#)

[Spis treści](#)

[Przykłady do książki](#)

[Opis książki](#)

[Czytaj fragment](#)

Internet rzeczy (IoT) przynosi kolejną technologiczną rewolucję: oto coraz więcej przedmiotów, także sprzętów codziennego użytku, dzięki wbudowanym czujnikom i dostępowi do sieci może

komunikować się z człowiekiem i z innymi przedmiotami. Możliwości takiej sieci są niewyobrażalne. Inżynierowie, którzy zajmują się tą koncepcją, rozwijają wiele różnych standardów mających służyć integracji IoT.

Okazuje się jednak, że w chaosie konkurujących rozwiązań brakuje spojrzenia na internet rzeczy z nieco szerszej perspektywy, która pozwoliłaby na opracowanie pragmatycznej i strukturalnej metodologii tworzenia urządzeń i usług IoT.

Niniejszą książkę napisano dla osób, które zaczynają swoją przygodę z internetem rzeczy. Zawarto tu informacje niezbędne do tworzenia prototypów urządzeń, produktów i aplikacji z wykorzystaniem infrastruktury WWW. Przedstawiono obszerne wprowadzenie w zagadnienia dotyczące internetu rzeczy. Znalazły się tu informacje o urządzeniach, czujnikach, standardach i potrzebnych narzędziach. Szczególnie dokładnie przedstawiono instrumentarium niezbędne do budowania WWW rzeczy — czyli warstwy aplikacji internetu rzeczy. Książka ta pozwoli na zrozumienie problemów dzisiejszego internetu rzeczy, poznanie dostępnych technik i ich wykorzystywanie.

W tej książce znajdziesz między innymi:

- omówienie koncepcji WWW rzeczy i internetu rzeczy
- sposoby wykorzystanie Node.js do implementacji WWW rzeczy
- kwestie związane ze stosowaniem protokołu HTTP oraz API typu RESTful
- charakterystykę poszczególnych warstw architektury WWW rzeczy
- metody integracji BeagleBone, Intel Edison oraz Arduino z internetem rzeczy

Internet rzeczy: fascynująca przyszłość, która zaczyna się dziś!

Dr Dominique Guinard — jest pionierem koncepcji architektury internetu rzeczy. Uczestniczył w wielu projektach związanych z tym zagadnieniem: badał duże sieci RFID, zajmował się telefonami komórkowymi jako bramami IoT, a także integrował sieci czujników z oprogramowaniem SAP.

Dr Vlad Trifa — jest uznanym ekspertem w dziedzinie rozproszonych rozwiązań pomiarowych, a także integracji urządzeń interaktywnych z aplikacjami korporacyjnymi. Zajmował się bioakustyką, przetwarzaniem sygnałów, interakcjami robotów humanoidalnych i sieciami neuronowymi. Najczęściej kupowane razem

Internet rzeczy. Budowa sieci z wykorzystaniem technologii webowych i Raspberry Pi

Podstawy projektowania interfejsów użytkownika

Tworzenie nowoczesnych systemów webowych

zestaw0 Internet rzeczy. Budowa sieci z wykorzystaniem technologii webowych i Raspberry Pi

zestaw1 Podstawy projektowania interfejsów użytkownika

zestaw2 Tworzenie nowoczesnych systemów webowych

119,85 zł

141,00 zł

Oszczędzasz: 21,15 zł (15%)

Podobne produkty

Oceny i opinie klientów (0) Dodaj opinię

Szczegóły książki

Tytuł oryginału:

Building the Web of Things: With examples in Node.js and Raspberry Pi

Tłumaczenie:

Piotr Rajca

ISBN Książki drukowanej:

978-83-283-2968-3, 9788328329683

Data wydania książki drukowanej:

2017-06-19

ISBN Ebooka:

978-83-283-2969-0, 9788328329690

Data wydania ebooka:

2017-06-16

Format:

170x230

Numer z katalogu:

59470

Rozmiar pliku Pdf:

10.5MB

Rozmiar pliku ePub:

9.1MB

Rozmiar pliku Mobi:

9.1MB

Pobierz przykładowy rozdział PDF

Przykłady na ftp

Zgłoś erratę

Poinformuj znajomego o książce

Kategorie:

Hardware » Inne

Technologie webowe » Inne

Spis treści

Przedmowa (9)

Podziękowania (11)

O książce (15)

O autorach (21)

CZĘŚĆ I. PODSTAWY IOT ORAZ WOT (23)

Rozdział 1. Od internetu rzeczy do WWW rzeczy (25)

1.1. Definicja internetu rzeczy (26)

1.2. Wejście do świata WWW rzeczy (28)

1.2.1. Scenariusz WWW rzeczy: podłączony hotel (29)

1.2.2. Porównanie IoT oraz WoT (30)

1.2.3. Internet rzeczy - krótka historia (35)

1.3. Przypadki zastosowania - dlaczego obiekty podłączone? (37)

1.3.1. Bezprzewodowe sieci czujników i pomiary rozproszone (37)

1.3.2. Urządzenia ubieralne i pomiary osobiste (39)

- 1.3.3. Inteligentne domy i budynki (40)
- 1.3.4. Inteligentne miasta i sieci energetyczne (41)
- 1.3.5. Inteligentna produkcja przemysłowa oraz Przemysł 4.0 (42)
- 1.3.6. Inteligentna logistyka i łańcuchy dostaw (43)
- 1.3.7. Marketing 2.0 (45)
- 1.4. WWW rzeczy - doładowany internet rzeczy (46)
 - 1.4.1. Większa łatwość programowania (48)
 - 1.4.2. Otwarte i rozszerzalne standardy (49)
 - 1.4.3. Szybkie i łatwe wdrażanie, utrzymanie i integracja (49)
 - 1.4.4. Luźne powiązania pomiędzy elementami (50)
 - 1.4.5. Powszechnie stosowane mechanizmy związane z bezpieczeństwem i prywatnością (51)
 - 1.4.6. WWW rzeczy - mankamenty (52)
- 1.5. Podsumowanie (53)

Rozdział 2. Witaj, świecie WWW rzeczy! (55)

- 2.1. Poznajemy urządzenie WWW rzeczy (56)
 - 2.1.1. Podejrżany: Raspberry Pi (57)
- 2.2. Ćwiczenie 1. Przeglądanie urządzenia na WWW rzeczy (58)
 - 2.2.1. Część 1. WWW jako interfejs użytkownika (58)
 - 2.2.2. Część 2. WWW jako API (62)
 - 2.2.3. Wnioski (69)
- 2.3. Ćwiczenie 2. Pobieranie danych z czujników w WWW rzeczy (69)
 - 2.3.1. Część 1. Odczyt bieżącej wartości czujnika (70)
 - 2.3.2. Część 2. Pobieranie danych z czujnika i rysowanie ich wykresu (71)
 - 2.3.3. Część 3. Aktualizacja danych na bieżąco (72)
 - 2.3.4. Wnioski (73)
- 2.4. Ćwiczenie 3. Działania w realnym świecie (74)
 - 2.4.1. Część 1. Zastosowanie formularza do zmiany tekstu na wyświetlaczu (74)
 - 2.4.2. Część 2. Utworzenie własnego formularza do kontroli urządzenia (77)
 - 2.4.3. Wnioski (79)
- 2.5. Ćwiczenie 4. Informujemy świat o naszym urządzeniu (79)
 - 2.5.1. Wnioski (83)
- 2.6. Ćwiczenie 5. Tworzenie pierwszej fizycznej aplikacji typu mashup (84)
 - 2.6.1. Wnioski (86)
- 2.7. Podsumowanie (87)

Rozdział 3. Node.js na potrzeby WWW rzeczy (89)

- 3.1. Era JavaScriptu - od klientów, przez serwery, do rzeczy! (90)
 - 3.1.1. Stosowanie JavaScriptu w rzeczach (92)
- 3.2. Wprowadzenie do Node.js (93)
 - 3.2.1. Instalowanie Node.js na komputerze (94)
 - 3.2.2. Pierwszy serwer webowy w Node.js (94)
 - 3.2.3. Zwracanie danych z czujnika w formacie JSON (96)
- 3.3. Modularność Node.js (98)
 - 3.3.1. npm - menedżer pakietów Node.js (98)
 - 3.3.2. Przejrzyste zależności dzięki zastosowaniu pliku package.json (99)
 - 3.3.3. Pierwszy własny moduł Node (101)
- 3.4. Przedstawienie pętli obsługi zdarzeń Node.js (102)
 - 3.4.1. Serwery wielowątkowe (103)
 - 3.4.2. Jednowątkowe serwery nieblokujące (103)
- 3.5. Pierwsze spotkanie z programowaniem asynchronicznym (106)
 - 3.5.1. Anonimowe funkcje zwrotne (106)

- 3.5.2. Nazwane funkcje zwrotne (110)
- 3.5.3. Biblioteki sterowania przepływem (112)
- 3.6. Podsumowanie i dalsze kroki (115)

Rozdział 4. Pierwsze spotkanie z systemami osadzonymi (117)

- 4.1. Świat urządzeń osadzonych (118)
 - 4.1.1. Urządzenia dla hobbystów a urządzenia do zastosowań przemysłowych (118)
 - 4.1.2. Systemy operacyjne czasu rzeczywistego a Linux (119)
 - 4.1.3. Podsumowanie. A co oprócz Pi? (121)
- 4.2. Przygotowanie pierwszego urządzenia WoT - Raspberry Pi (123)
 - 4.2.1. Prezentacja Raspberry Pi (124)
 - 4.2.2. Wybór urządzenia Pi (125)
 - 4.2.3. Lista zakupów (125)
 - 4.2.4. Przygotowywanie własnego Raspberry Pi (126)
 - 4.2.5. Nawiązywanie połączenia z Pi (131)
- 4.3. Instalowanie Node.js na Raspberry Pi (132)
 - 4.3.1. Stosowanie Git i serwisu GitHub na Pi (134)
 - 4.3.2. Czas na wnioski (135)
- 4.4. Podłączanie czujników i innych elementów do Pi (135)
 - 4.4.1. Prezentacja portów GPIO (135)
 - 4.4.2. Korzystanie z płytek stykowych i komponentów elektronicznych (136)
 - 4.4.3. Dostęp do portów GPIO z poziomu Node.js (138)
- 4.5. Podsumowanie (145)

Rozdział 5. Tworzenie sieci rzeczy (147)

- 5.1. Łączenie rzeczy (149)
 - 5.1.1. Topologie sieciowe (149)
 - 5.1.2. Modele klasyfikacji sieci (151)
- 5.2. Protokoły sieciowe dla rzeczy (153)
 - 5.2.1. Czynniki specjalne (154)
 - 5.2.2. Protokoły internetowe oraz IoT (154)
 - 5.2.3. Sieci osobiste internetu rzeczy (160)
 - 5.2.4. Sieci rozległe internetu rzeczy (165)
 - 5.2.5. A zatem które rozwiązanie wybrać? (168)
- 5.3. Protokoły warstwy aplikacji dla rzeczy (172)
 - 5.3.1. Stosy warstwy aplikacji ZigBee i Bluetooth (173)
 - 5.3.2. Apple HomeKit i Google Wave (174)
 - 5.3.3. Message Queuing Telemetry Transport (176)
 - 5.3.4. Constrained Application Protocol (178)
 - 5.3.5. A zatem co warto wybrać? (179)
- 5.4. Architektura WWW rzeczy (180)
 - 5.4.1. Warstwa 1. - dostęp (182)
 - 5.4.2. Warstwa 2. - odnajdywanie (182)
 - 5.4.3. Warstwa 3. - udostępnianie (182)
 - 5.4.4. Warstwa 4. - kompozycja (183)
 - 5.4.5. Dlaczego WWW rzeczy ma znaczenie? (183)
 - 5.4.6. Dalsze kroki (184)
- 5.5. Podsumowanie (185)

CZĘŚĆ II. TWORZENIE WWW RZECZY (187)

Rozdział 6. Dostęp: webowe API dla rzeczy (189)

- 6.1. Urządzenia, zasoby oraz WWW rzeczy (190)
 - 6.1.1. REST - Representational State Transfer (190)
 - 6.1.2. A dlaczego potrzebujemy jednolitego interfejsu? (192)
 - 6.1.3. Zasada 1. Możliwość adresowania zasobów (194)
 - 6.1.4. Zasada 2. Manipulacja zasobami poprzez reprezentację (198)
 - 6.1.5. Zasada 3. Zrozumiałe komunikaty (201)
 - 6.1.6. Zasada 4. Hipermedia jako mechanizm stanu aplikacji (208)
 - 6.1.7. Podsumowanie - proces projektowania rzeczy webowych (211)
- 6.2. Nie tylko REST: WWW rzeczy działająca w czasie rzeczywistym (212)
 - 6.2.1. WWW rzeczy potrzebuje zdarzeń! (212)
 - 6.2.2. Publikacja i subskrypcja (214)
 - 6.2.3. Webhook - webowe wywołania zwrotne (215)
 - 6.2.4. Comet - modyfikacje HTTP na potrzeby WWW czasu rzeczywistego (217)
 - 6.2.5. WebSocket (217)
 - 6.2.6. Przyszłość: od HTTP 1.1 do HTTP/2 (222)
- 6.3. Podsumowanie (223)

Rozdział 7. Implementacja rzeczy webowych (225)

- 7.1. Podłączanie urządzeń do WWW (226)
- 7.2. Wzorzec integracji bezpośredniej - REST na urządzeniu (227)
 - 7.2.1. Tworzenie serwera WoT (228)
 - 7.2.2. Projekt zasobów (230)
 - 7.2.3. Projekt reprezentacji (236)
 - 7.2.4. Projekt interfejsu (240)
 - 7.2.5. Implementacja interfejsu publikacji/subskrypcji przy użyciu WebSocket (243)
 - 7.2.6. Podsumowanie - wzorzec integracji bezpośredniej (246)
- 7.3. Wzorzec integracyjny bramy - przykład CoAP (246)
 - 7.3.1. Uruchamianie serwera CoAP (247)
 - 7.3.2. Użycie bramy do pośredniczenia w komunikacji CoAP (248)
 - 7.3.3. Podsumowanie - wzorzec integracyjny bramy (251)
- 7.4. Wzorzec integracyjny chmury - MQTT z użyciem platformy EVRYTHNG (251)
 - 7.4.1. Utworzenie konta EVRYTHNG (254)
 - 7.4.2. Tworzenie klienckiej aplikacji MQTT (258)
 - 7.4.3. Stosowanie akcji do kontrolowania wtyczki (260)
 - 7.4.4. Przygotowanie prostej aplikacji webowej do sterowania urządzeniem (262)
 - 7.4.5. Podsumowanie - wzorzec integracyjny chmury (266)
- 7.5. Podsumowanie (267)

Rozdział 8. Odnajdywanie: opisz i odkryj swoją webową rzecz (269)

- 8.1. Problem odnajdywania (270)
- 8.2. Odkrywanie rzeczy (272)
 - 8.2.1. Odkrywanie sieciowe (273)
 - 8.2.2. Wykrywanie zasobów w obrębie WWW (276)
- 8.3. Opisywanie rzeczy webowych (279)
 - 8.3.1. Prezentacja modelu Web Thing Model (281)
 - 8.3.2. Metadane (283)
 - 8.3.3. Właściwości (284)
 - 8.3.4. Akcje (286)
 - 8.3.5. Rzeczy (287)
 - 8.3.6. Implementacja modelu Web Thing Model na Pi (288)
 - 8.3.7. Podsumowanie - Web Thing Model (296)
- 8.4. Semantyczna WWW rzeczy (296)

- 8.4.1. Powiązane dane i RDFa (297)
- 8.4.2. Uzgodniona semantyka: Schema.org (301)
- 8.4.3. JSON-LD (302)
- 8.4.4. Dalsze kroki (305)
- 8.5. Podsumowanie (305)

Rozdział 9. Udostępnianie: zabezpieczanie i współdzielenie rzeczy webowych (307)

- 9.1. Zabezpieczanie rzeczy (309)
 - 9.1.1. ABC szyfrowania (311)
 - 9.1.2. Bezpieczeństwo w internecie dzięki TLS: to właśnie jest "S" z nazwy "HTTPS"! (313)
 - 9.1.3. Włączanie HTTPS i WSS korzystających z protokołu TLS na Raspberry Pi (315)
- 9.2. Uwierzytelnianie i kontrola dostępu (320)
 - 9.2.1. Kontrola dostępu z użyciem REST i żetonów API (321)
 - 9.2.2. OAuth: framework do uwierzytelniania (324)
- 9.3. Społecznościowa WWW rzeczy (327)
 - 9.3.1. Pośrednik uwierzytelniania społecznościowej WWW rzeczy (328)
 - 9.3.2. Implementacja pośrednika uwierzytelniania społecznościowej WWW rzeczy (330)
- 9.4. Dalsze kroki (339)
- 9.5. Podsumowanie (340)

Rozdział 10. Kompozycja: fizyczne aplikacje typu mashup (343)

- 10.1. Tworzenie prostej aplikacji - automatyczna generacja interfejsu użytkownika (345)
 - 10.1.1. Uniwersalny interfejs użytkownika dla rzeczy webowych (345)
- 10.2. Fizyczne aplikacje typu mashup (352)
 - 10.2.1. Node-RED: wizualne tworzenie aplikacji typu mashup (353)
- 10.3. IFTTT: tworzenie fizycznych aplikacji typu mashup przy użyciu kreatorów (360)
 - 10.3.1. Publikowanie informacji o włamaniach do arkusza Google (361)
 - 10.3.2. Wysyłanie żądań do rzeczy przy użyciu kanału Maker (363)
 - 10.3.3. Przesyłanie tweetów o włamaniach do arkusza w serwisie Google Drive (364)
- 10.4. Dalsze kroki (366)
 - 10.4.1. Aplikacje typu mashup - od prostych do złożonych aplikacji "big data" (366)
 - 10.4.2. Lepsze wrażenia użytkowników (367)
- 10.5. Podsumowanie (368)

Dodatek. Arduino, BeagleBone, Intel Edison i WWW rzeczy (369)

- A.1. Integracja BeagleBone z WoT (369)
 - A.1.1. Poznajemy BeagleBone Black (370)
 - A.1.2. Przygotowywanie BeagleBone Black na potrzeby tej książki (370)
- A.2. Integracja urządzenia Intel Edison z WoT (371)
 - A.2.1. Przygotowywanie urządzenia Edison na potrzeby tej książki (372)
- A.3. Integracja Arduino z WWW rzeczy (373)
 - A.3.1. Linux, SSH i Node.js (374)
- A.4. Integracja innych systemów osadzonych z WWW rzeczy (375)

Skorowidz (377)