

# PRZYSPIESZENIE

Lean i DevOps w rozwoju  
firm technologicznych



Nicole Forsgren,  
Jez Humble, Gene Kim

Helion 

Tytuł oryginału: Accelerate: The Science of Lean Software and DevOps: Building and Scaling High Performing Technology Organizations

Tłumaczenie: Krzysztof Sawka

ISBN: 978-83-283-5806-5

Copyright © 2018 by Nicole Forsgren, Jez Humble, and Gene Kim  
All rights reserved.

Chapter 16 by Steve Bell and Karen Whitley Bell is licensed under the Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License.

Polish edition copyright © 2019 by HELION SA.  
All rights reserved.

All rights reserved. No part of this book may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording or by any information storage retrieval system, without permission from the Publisher.

Wszelkie prawa zastrzeżone. Nieautoryzowane rozpowszechnianie całości lub fragmentu niniejszej publikacji w jakiegokolwiek postaci jest zabronione. Wykonywanie kopii metodą kserograficzną, fotograficzną, a także kopiowanie książki na nośniku filmowym, magnetycznym lub innym powoduje naruszenie praw autorskich niniejszej publikacji.

Wszystkie znaki występujące w tekście są zastrzeżonymi znakami firmowymi bądź towarowymi ich właścicieli.

Autor oraz Helion SA 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. Autor oraz Helion SA nie ponoszą również żadnej odpowiedzialności za ewentualne szkody wynikłe z wykorzystania informacji zawartych w książce.

Helion SA

ul. Kościuszki 1c, 44-100 Gliwice

tel. 32 231 22 19, 32 230 98 63

e-mail: [helion@helion.pl](mailto:helion@helion.pl)

WWW: <http://helion.pl> (księgarnia internetowa, katalog książek)

Drogi Czytelniku!

Jeżeli chcesz ocenić tę książkę, zajrzyj pod adres

<http://helion.pl/user/opinie/przyld>

Możesz tam wpisać swoje uwagi, spostrzeżenia, recenzję.

Printed in Poland.

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

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

# SPIS TREŚCI

Przedmowa <i>autorstwa Martina Fowlera</i>	9
Przedmowa <i>autorstwa Courtney Kissler</i>	13
Wademekum: zdolności napędzające proces usprawniania	15
Wstęp	17

## Część I. Nasze odkrycia

1. Przyspieszenie	27
2. Mierzenie wydajności	35
3. Mierzenie i zmienianie kultury	49
4. Praktyki techniczne	59
5. Architektura	73
6. Integracja bezpieczeństwa informacji z cyklem życia dostarczania oprogramowania	81
7. Praktyki zarządcze	85
8. Wytwarzanie produktu	91
9. Podtrzymywanie zapału do pracy	97
10. Satysfakcja, tożsamość i zaangażowanie pracownika	107
11. Liderzy i menedżerowie	119

## Część II. Badania

12. Podstawy naukowe badań opisanych w książce	131
13. Wprowadzenie do psychometrii	141
14. Dlaczego ankieta?	151
15. Dane wykorzystane w projekcie	161

### Część III. Transformacja

16. Wysokowydajne przywództwo i zarządzanie	169
Podsumowanie	187
A Zdolności napędzające usprawnienia	189
B Statystyki	197
C Metody statystyczne użyte w naszych badaniach	207
Podziękowania	213
Bibliografia	217
Skorowidz	225
Informacje o autorach	231

# WSTĘP

---

**P**od koniec 2013 r. rozpoczęliśmy czteroletnie badania mające na celu określenie zdolności i praktyk mających istotny wkład w przyspieszenie procesów tworzenia i dostarczania oprogramowania, a zatem również generowanie wartości przez firmy. Wyniki te są analizowane w kontekście rentowności, produktywności i udziałów rynkowych. Podobnie silne zjawiska dostrzegamy w niekomercyjnych aspektach efektywności, wydajności i satysfakcji klienta.

Badania te realizują dotychczas niezaspokojoną potrzebę rynkową. Za pomocą rygorystycznych metod naukowych, tradycyjnie stosowanych wyłącznie na uczelniach, które zostają następnie udostępnione całej branży, dążymy do celu, jakim jest unowocześnienie stanu wytwarzania i dostarczania oprogramowania. Możemy rozwinąć nasz sektor, pomagając identyfikować i rozumieć zdolności rzeczywiście napędzające poprawę wydajności w sposób statystycznie znaczący i niebędące wyłącznie tematem anegdot, a także wykraczające poza doświadczenie tylko jednego lub kilku zespołów.

Do przeprowadzenia badań opisanych w książce (a także prowadzonych aktualnie) skorzystaliśmy z technik przekrojowych. Takie same metody są stosowane w badaniach medycznych (np. w określaniu związku pomiędzy piciem piwa a otyłością, Bobak i in., 2003), badaniach miejsca pracy (np. wykazujących związek pomiędzy środowiskiem pracy a chorobami układu krążenia, Johnson i Hall, 1988) czy w badaniach pamięci (np. analizujących różnice pomiędzy rozwojem a zanikiem pamięci, Alloway i Alloway, 2013). Chcemy naprawdę zbadać naszą branżę i zrozumieć, jakie czynniki stanowią motor napędzający usprawnianie wydajności oprogramowania i organizacyjnej w istotny sposób, dlatego korzystamy z rygorystycznych, akademickich metod projektowania badań i publikujemy znaczną część naszych dokonań w periodykach specjalistycznych. Więcej informacji na temat metod używanych w naszych badaniach znajdziesz w części II, „Badania”.

## BADANIA

W trakcie naszych badań zebraliśmy z całego świata 23 000 odpowiedzi na pytania ankietowe. Skontaktowaliśmy się z ponad 2000 różnych organizacji, począwszy od małych firm liczących mniej niż pięciu pracowników, skończywszy na olbrzymich przedsiębiorstwach zatrudniających ponad 10 000 osób. Zebraliśmy dane zarówno z młodych firm i innowacyjnych przedsiębiorstw internetowych, jak również z silnie regulowanych sektorów, takich jak finansowy, medyczny czy rządowy. W naszych danych i analizach uwzględniamy oprogramowanie tworzone na najnowocześniejszych platformach, a także programowanie i utrzymywanie starego kodu.

Wnioski wysnute w naszej książce znajdują zastosowanie zarówno w organizacjach wykorzystujących tradycyjną metodologię kaskadową (zwaną także metodologią bramkową, ustrukturyzowaną lub zorientowaną na plan), które dopiero rozpoczynają transformację technologiczną, jak również w firmach implementujących od lat praktyki zwinne i DevOps. Dzieje się tak dlatego, że dostarczanie oprogramowania mieści się w zakresie ciągłego usprawniania, a zgodnie z naszymi badaniami rokrocznie lepsze staje się jeszcze lepsze, a jeżeli ktoś nie jest w stanie wprowadzać usprawnień, zostaje coraz bardziej z tyłu.

### *Usprawnienia są dostępne dla każdego*

Nasza misja zrozumienia sposobu, w jaki należy mierzyć i poprawiać dostarczanie oprogramowania, była pełna spostrzeżeń i niespodzianek. Morał tej historii wynikający z danych jest następujący: usprawnienia dostarczania oprogramowania są dostępne dla każdego zespołu i w każdej firmie, dopóki liderzy zapewniają ciągłe wsparcie (w tym czas, działania i zasoby), ukazując w ten sposób zaangażowanie w chęć poprawy, a także dopóki członkowie zespołu angażują się w pracę.

Podczas pisania tej książki naszym celem było podzielenie się wnioskami pomagającymi w rozwoju organizacji, uszczęśliwianiu zespołów, które szybciej dostarczałyby lepsze oprogramowanie, a także pozwalającymi na rozwój osób i organizacji. W dalszej części wstępu omówimy w skrócie nasze badania, ich genezę oraz sposób, w jaki zostały zrealizowane. Szczegółowe informacje na ten temat znajdziesz w części II.

## PODRÓŻ I DANE

Często jesteśmy pytani o genezę naszych badań. Bazuje ona na nieodpartej ciekawości i próbie odpowiedzi na pytanie, co sprawia, że wydajne organizacje technologiczne są wybitne, a także w jaki sposób oprogramowanie sprawia, że organizacje stają się jeszcze lepsze. Każdy z autorów poświęcał niezależnie od pozostałych czas na zrozumienie nadrzędnej wydajności technologicznej przed połączeniem sił pod koniec 2013 r.:

- **Nicole Forsgren** uzyskała stopień doktora systemów zarządzania informacją. Przed znanym 2013 r. spędziła kilka lat na badaniu czynników decydujących o wpływie technologii na organizacje, zwłaszcza wśród profesjonalistów tworzących oprogramowanie i obsługujących infrastrukturę. Jest autorką dziesiątków artykułów na ten temat publikowanych w periodykach branżowych. Zanim uzyskała stopień doktora, była inżynierem sprzętu i oprogramowania, a także administratorem systemów.
- **Jez Humble** jest współautorem książek: *DevOps. Światowej klasy zwinność, niezawodność i bezpieczeństwo w Twojej organizacji*, *Lean Enterprise* i *Ciągłe dostarczanie oprogramowania*. Po ukończeniu studiów pracował w młodej firmie w Londynie (w 2000 r.), a następnie od 2005 r. spędził dekadę w firmie ThoughtWorks, dostarczając oprogramowanie i służąc radami jako specjalista ds. infrastruktury, programista i menedżer produktu.
- **Gene Kim** analizował wydajne organizacje technologiczne już od 1999 r. Był założycielem i przez 13 lat dyrektorem ds. technicznych w firmie Tripwire, a ponadto jest współautorem wielu książek, w tym *Projekt Feniks* czy *The Visible Ops Handbook*.

Pod koniec 2013 r. Nicole, Jez i Gene zaczęli wspólnie pracować w zespole będącym częścią firmy Puppet w celu przygotowania raportu *2014 State of DevOps Report*<sup>1</sup>. Poprzez połączenie praktycznej biegłości z rygorem

---

<sup>1</sup> Należy zauważyć, że nie był to pierwszy raport z tej serii. W 2012 r. zespół z firmy Puppet Inc. zaprosił Gene'a do udziału w drugiej edycji badań poświęconych zrozumieniu mało znanego zjawiska o nazwie DevOps, sposobu jego przystosowywania i oczekiwanych przez organizacje korzyści wydajnościowych. Firma Puppet była wielkim orędownikiem i motorem napędowym tej metodyki, gdyż koncepcja „DevOps” zaczęła nabierać kształtu już po pierwszej konferencji DevOpsDays, dyskusjach w serwisie Twitter i wystąpieniach Johna Allspawa oraz Paula Hammonda. Następnie Gene zaprosił Jeza do współpracy i razem zebrali oraz przeanalizowali 4000 ankiet z całego świata, dzięki czemu było to największe przeprowadzone badanie tego typu.

akademickim zespół był w stanie uzyskać coś niepowtarzalnego w skali całej branży: raport zawierający wnioski na temat usprawnienia dostarczania wartości pracownikom, organizacjom i klientom poprzez technologię w przewidywalny sposób. W czterech kolejnych raportach autorzy nadal współpracowali z zespołem Puppet, co pozwoliło im wykorzystywać schemat badań i ciągle zwiększać wiedzę branżową na temat czynników wpływających na znakomite dostarczanie produktu i aktywujących skuteczne zespoły techniczne, a także przekształcać firmy w wysokowydajne organizacje i zdobywać rynek za pomocą technologii. Ta książka opisuje wnioski uzyskane z czterech lat badań, počawszy od wspomnianego raportu (w latach 2014 – 2017).

Aby zdobyć dane, co roku wysyłaliśmy zaproszenia do osób z naszych list dyskusyjnych, ponadto korzystaliśmy z mediów społecznościowych, w tym z serwisów Twitter, LinkedIn i Facebook. Zaproszenia były docelowo kierowane do profesjonalistów pracujących w branży technologicznej, zwłaszcza zaznajomionych z paradygmatami wytwarzania i dostarczania oprogramowania oraz technikami DevOps. Zachęcaliśmy czytelników do zapraszania znajomych i współpracowników również mających styczność z tymi paradygmatami w celu poszerzenia naszych horyzontów. Jest to tzw. metoda kuli śnieżnej; w rozdziale 15., „Dane wykorzystane w projekcie”, wyjaśniamy, dlaczego ta metoda gromadzenia danych jest odpowiednia dla naszego projektu badawczego.

Dane do naszego projektu pochodzą z ankiet. Skorzystaliśmy z ankiet, ponieważ stanowią najlepszy sposób gromadzenia dużych ilości danych z tysięcy organizacji w krótkim czasie. Szczegółowe wyjaśnienia, dlaczego za pomocą ankiet można przeprowadzać dobre badanie, a także informacje na temat wykorzystanych przez nas mechanizmów gwarantujących rzetelność i trafność uzyskanych odpowiedzi są dostępne w części II.

Poniżej prezentujemy krótki zarys badań i ich ewolucję w czasie.

## ROK 2014: KŁADZENIE PODWALIN. SKUTECZNOŚĆ DOSTARCZANIA I WYDAJNOŚĆ ORGANIZACYJNA

Naszym celem w pierwszym roku działalności było wyznaczenie podstaw, jeśli chodzi o zrozumienie wytwarzania i dostarczania oprogramowania w organizacjach.

Niektórymi z kluczowych pytań były:

- Co to znaczy „dostarczać oprogramowanie” i czy proces ten jest mierzalny?



- Czy dostarczanie oprogramowania wpływa na organizację?
- Czy kultura ma znaczenie i jak możemy ją mierzyć?
- Jakie praktyki techniczne wydają się istotne?

Byliśmy pozytywnie zaskoczeni wieloma wynikami uzyskanymi w pierwszym roku. Odkryliśmy, że wytwarzanie i dostarczanie oprogramowania może być mierzone w sposób statystycznie znaczący, a najskuteczniejsze organizacje realizują to znacznie lepszymi sposobami od pozostałych organizacji. Zauważyliśmy również związek pomiędzy przepustowością a stabilnością i stwierdziliśmy, że umiejętność tworzenia oprogramowania wpływa pozytywnie na dochodowość, produktywność i udziały rynkowe firmy. Dostrzegliśmy znaczenie kultury i praktyk technicznych, a także wyznaczyliśmy sposoby ich mierzenia. Informacje te zostały szczegółowo omówione w części I.

Zespół skorygował również wcześniejszy sposób pomiaru większości danych i przeszedł z prostych pytań pozwalających wybrać „tak” lub „nie” do odpowiedzi w skali Likerta (w której osoby ankietowane wybierają odpowiedzi w szerokim zakresie, od „zdecydowanie się zgadzam” do „zdecydowanie się nie zgadzam”). Taka prosta zmiana pytań pozwoliła gromadzić subtelniejsze dane: odcienie szarości zamiast czerni i bieli. Umożliwiło to zespołowi bardziej szczegółową analizę danych. W rozdziale 14., „Dlaczego ankietę?”, wyjaśniliśmy, z jakich powodów wybraliśmy ankietę do tego projektu badawczego i dlaczego możesz ufać zdobytym w ten sposób danym.

## ROK 2015: POSZERZENIE ZAKRESU PRACY I POGŁĘBIENIE ANALIZY

Prowadzenie badań polega, podobnie jak w przypadku przekształceń technologicznych i rozwoju firmy, na powtarzalności, usprawnieniach przyrostowych i wielokrotnej analizie ważnych wyników. Uzbrojeni we wnioski z poprzedniego roku skoncentrowaliśmy się na ponownej analizie i potwierdzeniu niektórych kluczowych odkryć (np. czy dostarczanie oprogramowania może być definiowane i mierzone w sposób statystycznie znaczący lub czy dostarczanie oprogramowania wpływa na wydajność organizacyjną) przy jednoczesnym rozszerzeniu istniejącego modelu.

Oto niektóre z przygotowanych pytań:

- Czy możemy ponownie sprawdzić wpływ dostarczania oprogramowania na wydajność organizacyjną?

- Czy praktyki techniczne i automatyzacja wpływają na dostarczanie oprogramowania?
- Czy praktyki zarządzania szczupłego (ang. *Lean management practices*) wpływają na dostarczanie oprogramowania?
- Czy praktyki techniczne i praktyki zarządzania szczupłego wpływają na aspekty pracy związane z siłą roboczą, np. takie jak niepokój związany z wdrażaniem kodu lub wypalenie zawodowe?

Również w owym roku wiele naszych przypuszczeń znalazło potwierdzenie i natrafiliśmy na różne niespodzianki. Nasze hipotezy zostały podtrzymane, co stanowiło utwierdzenie i rozwinięcie pracy wykonanej w roku poprzednim. Wyniki te znajdziesz w części I.

## ROK 2016: SZERSZE SPOJRZENIE NA PRAKTYKI TECHNICZNE I ANALIZA ROZMYTEJ FAZY PRZEDPROJEKTOWEJ

W trzecim roku ponownie skorzystaliśmy ze zrębów naszego modelu i rozwinęliśmy je poprzez analizę znaczenia dodatkowych praktyk technicznych (takich jak bezpieczeństwo, programowanie bazujące na gałęzi głównej czy zarządzanie danymi testowymi). Inspirację stanowiły dla nas również rozmowy z pracownikami działów zarządzania produktem, dzięki którym poszerzyliśmy zakres analizy w kierunku umożliwiającym próbę zmierzenia skutków współczesnego przechodzenia od tradycyjnych praktyk zarządzania projektem ku wprowadzaniu technik szczupłych w zarządzaniu produktem. Rozwinęliśmy nasze badania tak, aby obejmowały metryki jakościowe, takie jak defekty, przerabianie wersji czy usprawnianie zabezpieczeń. Dołączyliśmy także dodatkowe pytania pomagające zrozumieć wpływ praktyk technicznych na kapitał ludzki: wskaźnik rekomendacji netto pracowników (ang. *employee Net Promoter Score* — eNPS) i tożsamość pracy, czyli czynnik, który prawdopodobnie zmniejsza wypalenie zawodowe.

Takie pytania zadawaliśmy w owym roku:

- Czy integracja zabezpieczeń z wytwarzaniem i dostarczaniem oprogramowania pomaga procesowi czy go spowalnia?
- Czy wytwarzanie bazujące na gałęzi głównej ma pozytywny wpływ na dostarczanie oprogramowania?

- Czy metody szczupłe w zarządzaniu produktem stanowią istotny aspekt wytwarzania i dostarczania oprogramowania?
- Czy dobre praktyki techniczne przyczyniają się do lojalności wobec pracodawcy?

## ROK 2017: UWZGLĘDNIENIE ARCHITEKTURY, ANALIZA ROLI LIDERÓW I MIERZENIE SUKCESÓW W ORGANIZACJACH NIENASTAWIONYCH NA ZYSKI

W czwartym roku badań zaczęliśmy się zastanawiać nad sposobami tworzenia systemów i wpływem architektury na zdolność zespołów oraz organizacji do wytwarzania/dostarczania oprogramowania, a także wartości. Zaczęliśmy też uwzględniać metryki wartości wykraczające poza zyskowność, produktywność i udziały rynkowe, dzięki czemu analiza mogła dotyczyć również grupy odbiorców niezorientowanych na zyski. Ponadto w owym roku zajęliśmy się badaniem roli liderów w celu określeniu wpływu przywództwa transformacyjnego w organizacjach.

Głównymi pytaniami w czwartym roku badań były:

- Jakie praktyki architektoniczne napędzają usprawnienia wydajności w dostarczaniu oprogramowania?
- W jaki sposób przywództwo transformacyjne wpływa na dostarczanie oprogramowania?
- Czy dostarczanie oprogramowania wpływa na wyniki niezorientowane na zyski?

## PODSUMOWANIE

Mamy nadzieję, że w trakcie lektury tej książki odkryjesz, jako technolog i lider technologiczny, podstawowe składniki usprawniające Twoją organizację, počawszy od dostarczania produktu. Właśnie poprzez proces udoskonalania zdolności dostarczania oprogramowania organizacje są w stanie szybciej dostarczać nowe cechy, modyfikować strategię w razie potrzeby, reagować na zmiany zgodności i zabezpieczeń, a także korzystać z zalet ścisłej współpracy z użytkownikami w celu przyciągania nowych klientów i satysfakcjonowania dotychczasowych.

W kolejnych rozdziałach zidentyfikujemy kluczowe zdolności decydujące o wydajności dostarczania oprogramowania (i definiujące wydajność dostarczania oprogramowania), a także skrótowo omówimy ich najważniejsze aspekty. W części I opisujemy nasze odkrycia, część II poświęciliśmy omówieniu zastosowanych metod naukowych, natomiast w części III prezentujemy analizę przypadku, w którym organizacje przystosowują i implementują opisane w tej książce zdolności w celu uzyskiwania lepszych wyników.

# SKOROWIDZ

---

## A

Almeida Thiago, 98  
analiza  
dedukcyjna, 136  
dedukcyjno-predykcjna, 133, 137  
deskryptywna, 133, 134  
raport, 133, 134  
eksploracyjna, 133, 134  
głównych składowych, 209  
klasyfikacyjna, *Patrz:* analiza skupień  
korelacji, 135  
mechanistyczna, 133, 137  
predykcjna, 133, 137  
przyczynowa, 133, 137  
przypadków, 131  
skupień, 40, 41, 138, 211  
hierarchiczna, 138  
wariancji, 212  
złożoność, 133  
ankieta  
błędy, 142, 143  
przygotowanie, 207  
wiarygodność wyników, 141, 143, 155  
zalety, 151, 153, 157  
ANOVA, *Patrz:* analiza wariancji  
architektura, 64, 73  
implementacja, 74  
luźno sprzężona, 76, 77, 192, 201  
mikrouslug, 77  
testowalność, 75  
wdrażalność, 75  
zorientowana na usługi, 77, 79  
automatyzacja testów, *Patrz:* test automatyzacja

awaria, 56, 67  
czas wymagany do odzyskania  
sprawności, *Patrz:* stabilność  
współczynnik, *Patrz:* współczynnik  
awaryjności zmian

## B

badania, 131, 132, 133  
bezpieczeństwo, 79  
informacji, 71, 81, 82  
OWASP Top 10, 82  
przesuwanie „w lewo”, 63, 82, 99, 192  
Blank Steve, 91  
błąd, 142  
systematyczny, 208  
wspólnej metody, 208  
ból wdrażania, 77, 97, 98, 200, 203  
eliminowanie, 63, 65, 71, 100, 104  
pomiar, 47, 99  
brygada, 172  
zebranie, 174, 175

## C

center of expertise, *Patrz:* centrum wiedzy  
centrum wiedzy, 173  
CI, *Patrz:* integracja ciągła  
cluster analysis, *Patrz:* analiza skupień  
CMB, *Patrz:* błąd wspólnej metody  
CMV, *Patrz:* wariancja wspólnej metody  
Cockcroft Adrian, 112  
common method bias, *Patrz:* błąd wspólnej  
metody

common method variance, *Patrz:*  
wariancja wspólnej metody  
continuous integration, *Patrz:* integracja  
ciągła  
Conway Melvin, 76  
Corman Josh, 84  
cykl Deminga, 176

## D

dane  
ankietowe, 156, 157  
    wiarygodność, *Patrz:* ankieta  
    wiarygodność danych  
obiektywne, 157  
pozyskiwanie, 162, 207  
systemowe, 150, 156  
    nieprzetworzone, 152  
    ograniczenia, 154  
testowe, 22, 63, 70, 99, 191  
Deming W. Edwards, 60  
Deminga  
    cykl, *Patrz:* cykl Deminga  
    postulaty, *Patrz:* zarządzanie postulaty  
    Deminga  
deployment pain, *Patrz:* ból wdrażania  
DevOps, 77, 81  
    metodyka, 28  
    minikonferencja, 125  
    odporny, 84  
    pomiar, 29  
    zalety, 32  
DevSecOps, 84  
DiRT, *Patrz:* testowanie przywracania  
    poawaryjnego  
Disaster Recovery Testing, *Patrz:*  
    testowanie przywracania poawaryjnego  
dzień gry, 126

## E

employee Net Promoter Score, *Patrz:* eNPS  
eNPS, 22, 108, 109, 123, 204  
extreme programming, *Patrz:*  
    programowanie ekstremalne

## G

GitHub Flow, 70, 71

## H

historyjka, 36  
hoshin kanri, 176

## I

information security, *Patrz:* zespół  
    ds. bezpieczeństwa informacji  
infosec, *Patrz:* zespół ds. bezpieczeństwa  
    informacji  
integracja ciągła, 61, 62, 99, 191

## K

koło sukcesu, 92, 111, 112  
konstrukt, 56  
    ciągłego dostarczania oprogramowania  
    pierwszego rzędu, 64  
ukryty, 143, 144, 145, 146  
    dane systemowe, 150  
    testowany statystycznie, 148, 149  
    trafność, 147  
    Westruma, 53  
kontrola wersji, 61, 62, 63, 68, 89, 99,  
    154, 189, 199  
korelacja, 135, 137, 197, 209, 210  
Krishnan Kripa, 127  
kultura, 49, *Patrz też:* organizacja  
    generatywna, 65  
    mierzenie, 51, 145  
    modelowanie, 49  
    organizacyjna, 49, 92, 104, 147, 157  
    typologia, 143, 144, 145

## L

Lean management, *Patrz:* zarządzanie  
    szczupłe  
Lean practice, *Patrz:* metoda szczupła  
Leek Jeffrey, 133  
lider, 119, 169, 204  
    służebny, 121  
    technologiczny, 125  
    transformacyjny, 120, 123, 124  
    umiejętności, 125

## M

manewr odwrotny Conwaya, 76  
manifest  
  odporny, 84  
  programowania zwinnego, 59  
Maslach Christina, 102, 103  
menedżer, 124, 125, *Patrz też:* lider  
metoda  
  kuli śnieżnej, 164, 207  
  mapowania Wardleya, 47  
  Scrum, 59  
  szczupła, 28, 40  
  Vanguard, 68  
metodologia  
  bramkowana, *Patrz:* metodologia  
  kaskadowa  
  kaskadowa, 18  
  zorientowana na plan, *Patrz:*  
  metodologia kaskadowa  
metodologia zwinnego wytwarzania  
  oprogramowania, *Patrz:* oprogramowanie  
  wytwarzanie zwinne  
metryka  
  DevOps, *Patrz:* DevOps metryka  
  szczupła, 28, 40  
minimal viable product, *Patrz:* MVP  
model  
  biznesowy, 91  
  dojrzałości, 29, 30, 31  
  pięciopłaszczyznowy, 120  
  pomiarowy, 209  
  Westruma, 50, 51, 54, 55, 87, 200, 203  
  zdolności, 29, 30, 31  
MVP, 92, 93, 193

## N

narzędzia, 64, 80  
  monitorowania wydajności, 86  
  wybór, 79  
Net Promoter Score, *Patrz:* NPS  
NPS, 108, 109, 143

## O

obszar produktowy główny, 172  
oprogramowanie, 27, 28  
  bazujące na  
  długoterminowych rozgałęzieniach  
  funkcji, 71

  gałęzi głównej, 70, 191, 200  
  krótkoterminowych  
  rozgałęzieniach funkcji, 71  
dostarczanie, 21, 35, 38, 76, 94  
ciągłe, 28, 57, 59, 60, 61, 62, 63,  
  64, 66, 68, 71, 192, 199  
wydajność, 31, 32, 37, 39, 40, 41,  
  42, 45, 46, 47, 55, 56, 65, 70, 71,  
  73, 82, 88, 92, 104, 122, 154, 198  
jakość, 66, 71  
komercyjne, 74  
kontrola wersji, *Patrz:* kontrola wersji  
niestandardowe, 74  
niestrategiczne, 47  
projektowanie, 35, 38  
strategiczne, 47  
typu mainframe, 74  
wdrażanie  
  automatyzacja, 191  
  ból, *Patrz:* ból wdrażania  
  częstotliwość, 39, 48, 56, 138  
  niepowodzenie, 100  
wytwarzanie, 21, 38  
  metodami eksperymentalnymi, 91  
  produktywność, 35, 36  
  szczupłe, 37, 39, 91, 92  
  wydajność, 35  
  zwinne, 91, 93

organizacja  
  biurokratyczna, 50  
  generatywna, 50, 65, 87, 112, 194  
  niskowydajna, 67, 74, 78, 138, 199,  
  200, 204  
  patologiczna, 50  
  średniowydajna, 78, 138  
  wysokowydajna, 67, 68, 78, 79, 82,  
  84, 108, 122, 138, 181, 198, 199,  
  200, 201, 204

## P

Pal Topo, 84  
PCA, *Patrz:* analiza głównych składowych  
Plan, Do, Check, Act, *Patrz:* cykl Deminga  
populacja, 134  
praca  
  w toku, 86, 174, 194  
  z małymi zadaniami, 92, 93, 94, 193

pracownik  
lider, *Patrz:* lider  
lojalność, 107, 108  
płeć, 114, 116  
rekomendacja, 22  
rotacja, 158  
satysfakcja, 107, 112, 113  
wypalenie zawodowe, 77, 87, 92, 101, 200, *Patrz też:* zespół wypalenie zawodowe  
objawy, 101  
pomiar, 103  
przyczyny, 101, 102, 105  
zapobieganie, 101, 102, 104  
zaangażowanie, 107, 108  
predykcja, 197  
principal components analysis, *Patrz:* analiza głównych składowych  
product area lead, *Patrz:* obszar produktowy główny  
produkt  
o minimalnej funkcjonalności, *Patrz:* MVP  
zarządzanie, *Patrz:* zarządzanie produktem  
prognoza, *Patrz:* predykcja  
programowanie  
bazujące na długoterminowych rozgałęzieniach funkcji, 63  
gałęzi głównej, 22, 63, 99  
bazujące na testach, 59  
ekstremalne, 59  
parami, 88  
zorientowane na testowanie, *Patrz:* TDD  
zwinne, 59  
projekt  
o otwartym kodzie źródłowym, 71  
zarządzanie, 22  
projektowanie dedukcyjne, 197  
prototyp, 91, 93, 193  
próba, 134  
próbkowanie  
nieprobabilistyczne, 163  
probabilistyczne, 163  
referencyjne, 164  
przepychanie sondażowe, 142  
przyczynowość, 135

przywództwo, 169  
służebne, 121  
technologiczne, 119  
transformacyjne, 120, 121, 122, 123  
psychometria, 141

## R

raport Forrestera, 28, 29, 133  
regresja, 211  
return of investment, *Patrz:* ROI  
Ries Eric, 91  
ROI, 45  
ruch  
DevOps, *Patrz:* DevOps  
szczypty, 28, 40, 57, *Patrz też:* zarządzanie szczyple  
zwinny, 57, 59  
Rugged Manifesto, *Patrz:* manifest odporny  
rzetelność, 53, 210

## S

Schwartz Mark, 54  
Seddon John, 68  
service-oriented architecture, *Patrz:* architektura zorientowana na usługi  
Shook John, 57  
skala Likerta, 21, 51, 63, 111, 132  
SOA, *Patrz:* architektura zorientowana na usługi  
spójność wewnętrzna, *Patrz:* rzetelność  
squad, *Patrz:* brygada  
stabilność, 32, 40, 47, 48, 56, 64, 66, 88, 138  
strumień wartości, 172, 193  
system  
kontroli wersji, *Patrz:* kontrola wersji  
rejestracji danych, 74  
zaangażowania, 74

## T

TDD, 69  
technologia, 28, 80  
teoria  
kolejek, 37  
Westruma, *Patrz:* model Westruma



test, 75  
A/B, 47, 93, 137  
akceptacyjny, 62, 69  
automatyczny, 69  
automatyzacja, 62, 69, 72  
chi-kwadrat, 208  
dane, *Patrz:* dane testowe  
eksploracyjny, 62  
jednoczynnikowy Harmana, 208  
jednostkowy zautomatyzowany, 62  
korelacji markera, 208  
pakiet kwarantannowy, 69  
t, 208  
Tukeya, 212  
zamiennik, *Patrz:* zamiennik testowy  
test-driven development, *Patrz:* TDD  
tester, 70  
testowanie  
ciągłe, 62  
przywracania poawaryjnego, 127  
trafność, 53

**U**

utyliczacja, 37

**V**

Vigen Tyler, 135

**W**

Wardley Simon, 47  
Wardleya metoda, *Patrz:* metoda mapowania Wardleya  
wariancja  
przeciętna wyodrębniona, 209  
wspólnej metody, 208  
Weinberg Jerry, 66  
Westrum Ron, 50, 144  
Westruma  
konstrukt, *Patrz:* konstrukt Westruma  
model, *Patrz:* model Westruma  
Wickett James, 81, 84  
WIP, *Patrz:* zapasy w toku produkcji, praca w toku  
wirtualizacja, 77  
Work in Process, *Patrz:* zapasy w toku produkcji, praca w toku

wskaźnik  
lojalności, 108  
rekomendacji netto, 108, 109, 143  
pracowników, *Patrz:* eNPS  
współczynnik  
awaryjności zmian, 32, 37, 41, 45, 48, 56, 66, 138  
korelacji Pearsona, 135, 210  
wydajność  
dostarczania oprogramowania, *Patrz:* oprogramowanie dostarczanie wydajność  
organizacyjna, 45, 46, 55, 200, 203  
wyświetlacz, 86

## X

XP, *Patrz:* programowanie ekstremalne

## Y

Yegge Steve, 79

## Z

zamiennik testowy, 77  
zapasy w toku produkcji, 86, 174, 194  
zarządzanie  
danymi testowymi, *Patrz:* dane testowe  
postulaty Deminga, 60  
produktem, 22, 92  
szczupłe, 94, 202  
projektem, 22  
ryzykiem w kontekście zmian, 90  
szczupłe, 22, 85, 86, 87, 202, 203  
zespół, 124  
badawczy, 131  
ds. bezpieczeństwa informacji, 81, 82, 83  
produktywność, 36  
przekrojowy, 126  
rotacja, 158  
wypalenie zawodowe, 47, 59, 63, 71, 97, *Patrz też:* pracownik wypalenie zawodowe  
zróźnicowanie, 114, 117, 204  
mniejszości, 116  
płeć, 114, 116  
zwrot z inwestycji, *Patrz:* ROI



# PROGRAM PARTNERSKI

— GRUPY 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>

GRUPA  
**Helion**

# Lean. Przygotuj się na prawdziwe przyspieszenie!

Tworzenie oprogramowania to wymagający obszar branży technologicznej — trzeba szybko reagować na zmieniające się uwarunkowania i w każdej chwili być gotowym na kolejną innowację, a wszystko po to, by budować przewagę konkurencyjną. Rozwijanie skomplikowanych systemów informatycznych powoli przypomina ciągły wyścig z czasem. Często ceną za skrócenie czasu pisania kodu i wdrożenia oprogramowania bywa zmniejszenie stabilności lub większa awaryjność systemów w środowiskach produkcyjnych. Czy musisz wybierać między szybkością dostarczania rozwiązań a stabilnością pracy systemu? Już nie! Pogodzisz je dzięki odpowiedniej kulturze organizacyjnej i należytej jakości zarządzania pracą zespołu.

Ta książka zawiera odpowiedzi na pytania, które zadaje sobie każdy szef zespołu programistów. Jakie czynniki wpływają na terminowe dostarczanie produktu? Jak wydobyć twórczy potencjał z zespołów technicznych? I wreszcie: w jaki sposób przekształcić firmę w wysokowydajną organizację, która przebojem zdobędzie rynek za pomocą technologii? Przedstawiono tu wyniki badań przeprowadzonych w latach 2014 – 2017. Pokazano, jakie działania pozwolą podnieść efektywność dostarczania oprogramowania, a w efekcie — szybciej modyfikować strategię, dostarczać nowe funkcjonalności, reagować na zmiany zgodności i zabezpieczeń oraz korzystać z zalet ścisłej współpracy z użytkownikami. Wdróż te techniki, a rozwój Twojej organizacji imponująco przyspieszy!

## Poznaj kluczowe dla sukcesu czynniki zgrupowane w pięciu kategoriach:

- ciągłe dostarczanie oprogramowania
- architektura
- produkt i proces
- zarządzanie metodą lean
- kultura

**Dr Nicole Forsgren** jest wybitną badaczką i konsultantką w dziedzinie zarządzania wiedzą i systemami IT. Przeprowadziła obszerne studia nad DevOps, jest autorką licznych publikacji naukowych na ten temat.

**Jez Humble** nigdyś pracował w start-upie, po czym na 10 lat związał się z ThoughtWorks, gdzie zajmował się rozwijaniem produktów i konsultacjami. Obecnie wykłada na UC Berkeley.

**Gene Kim** jest utalentowanym dyrektorem ds. technologii, badaczem i autorem książek. Specjalizuje się w doskonaleniu przepływu pracy programistów w celu podniesienia jakości tworzonego kodu.

	<i>Sprawdź nasze szkolenia!</i>	<b>KOD KORZYŚCI</b> <i>Sięgnij po więcej!</i>	
 <b>helion.pl</b>	 AKADEMIA IT & BUSINESS	ISBN 978-83-283-5806-5	
 <b>HELION SA</b> ul. Kościuszki 1c 44-100 Gliwice tel.: 32 230 98 63 helion@helion.pl	<a href="http://WWW.SZKOLENIA.HELION.PL">WWW.SZKOLENIA.HELION.PL</a>	 9 788328 358065	
<b>INFORMATYKA W NAJLEPSZYM WYDANIU</b>		Cena: 49,00 zł	