Basiskennis IT

De eerste stap van een leven lang leren

Bart de Best

Onder redactie van
Louis van Hemmen

Colofon

Meer informatie over deze en andere uitgaven kunt u verkrijgen bij:
Leonon Media
(0)572 - 851 104

Algemene vragen : info@leonon.nl
Sales vragen : verkoop@leonon.nl
Manuscript / auteur : redactie@leonon.nl

© 2021 Leonon Media

Omslagontwerp : Eric Coenders, IanusWeb, Nijmegen
Productie : Printforce B.V., Culemborg

Titel : Basiskennis IT
Sub titel : De eerste stap van een leven lang leren
Datum : 7 november 2021
Auteur : Bart de Best
Uitgever : Leonon Media
ISBN13 : 978 94 92618 573
Druk : Eerste druk, 7 november 2021

©2021, Leonon Media
Niets uit deze uitgave mag worden verveelvoudigd en/of openbaar gemaakt door middel van druk, fotokopie, microfilm, of welke andere wijze dan ook, zonder voorafgaande schriftelijke toestemming van de uitgever.

TRADEMARK NOTICES
ITIL® and PRINCE2® are registered trademarks of Axelos Limited.
Scaled Agile Framework and SAFe are registered trademarks of Scaled Agile, Inc.

"De enige wijsheid die echt telt is weten dat je niets weet."

by Socrates

Inhoudsopgave

1 INTRODUCTIE 1
1.1 DOEL 1
1.2 DOELGROEP 1
1.3 STRUCTUUR 1
1.3.1 HOOFDSTUK 2: IT LANDSCHAP 1
1.3.2 HOOFDSTUK 3: SOFTWARE ONTWIKKELEN 2
1.3.3 HOOFDSTUK 4: SOFTWARE BEHEREN 2
1.3.4 HOOFDSTUK 5: TRENDS IN DE IT 3
1.4 BIJLAGEN 3
1.5 LEESWIJZER 4

2 IT LANDSCHAP 5
2.1 BASISCONCEPTEN 5
2.2 AFKORTINGEN 7
2.3 ONTSTAANSGESCHIEDENIS 7
2.4 DE STACK 8
2.5 SOFTWARE 9
2.6 TOEPASSINGSPROGRAMMATUUR 10
2.6.1 MAATWERKPROGRAMMATUUR 10
2.6.2 SOFTWAREPAKKETTEN 10
2.7 SYSTEEMPROGRAMMATUUR 11
2.7.1 BESTURINGSSYSTEEM 12
2.7.2 DATABASE MANAGEMENT SYSTEEM 12
2.8 STORAGE 12
2.9 HARDWARE 13
2.10 FIRMWARE 14
2.11 NETWERKEN 14
2.12 COMMUNICATIE 15
2.13 VIRTUALISERING 15
2.14 CLOUD SERVICES 16
2.15 CLOUD SERVICE PROVIDER 17
2.16 DATA CENTERS 19
2.17 DE SAMENHANG 20

3 SOFTWARE ONTWIKKELEN 21
3.1 BASISCONCEPTEN 21
3.2 AFKORTINGEN 21
3.3 ONTSTAANSGESCHIEDENIS 22
3.4 SOFTWARE ONTWIKKELPROCES 23
3.4.1 WATERVAL ONTWIKKELPROCES 23
3.4.2 AGILE ONTWIKKELPROCES 24
3.5 ARCHITECTUUR 25
3.5.1 MONOLIET 26
3.5.2 MICROSERVICE 27
3.5.3 REST API 27
3.5.4 ENTERPRISE SERVICE BUS 27
3.5.5 CONTAINERS 29
3.6 ONTWERP 29
3.6.1 FUNCTIONEEL ONTWERP 29

3.6.2 Technische ontwerp 30
3.7 Testen 31
3.8 Programmeren 32
3.9 Governance 33
3.10 Programmeertalen 33
3.11 Apps versus Applicaties 34
3.12 Samenhang 35

4 SOFTWARE BEHEREN 37
4.1 Basisconcepten 37
4.2 Afkortingen 38
4.3 Ontstaansgeschiedenis 39
4.4 Drievoudig beheermodel 40
4.5 Beheermodel ITIL v2 40
4.5.1 Service Support Set 41
4.5.2 Service Delivery Set 42
4.6 Beheermodel ITIL v3 44
4.6.1 Service strategy 44
4.6.2 Service Design 46
4.6.3 Service Transition 47
4.6.4 Service Operation 49
4.6.5 Continual Service Improvement (CSI) 49
4.7 Beheermodel ITIL 4 50
4.8 Ontwikkeling en beheer 51
4.9 Housing en hosting 53
4.10 On premise 53
4.11 Demand / supply 53
4.12 Outsourcing, insourcing, nearsourcing, offsourcing 53
4.13 Samenvatting 54

5 TRENDS IN DE IT 55
5.1 Basisconcepten 55
5.2 Afkortingen 55
5.3 Top 5 ontwikkelingen 55
5.4 Kennis up-to-date 56
5.5 Artificial Intelligence 56
5.6 Machine learning 57
5.7 Digitalisering 58
5.8 Robotisering 58
5.9 Internet of things 59
5.10 Samenhang 59

BIJLAGE A, LITERATUURLIJST 63
BIJLAGE B, BEGRIPPENLIJST 67
BIJLAGE C, AFKORTINGEN 83
BIJLAGE D, AGILE TOOLS 87
BIJLAGE E, WEBSITES 91
BIJLAGE F, INDEX 93

Figuren

FIGUUR 1-1, DE VIER PERSPECTIEVEN VAN IT. 1
FIGUUR 1-2, BASISKENNIS - IT LANDSCHAP. 2
FIGUUR 1-3, BASISKENNIS - SOFTWARE ONTWIKKELEN. 2
FIGUUR 1-4, BASISKENNIS - SOFTWARE BEHEREN. 3
FIGUUR 1-5, BASISKENNIS – TRENDS IN DE IT. 3
FIGUUR 2-1, DE STACK. 9
FIGUUR 2-2, STORAGE. 13
FIGUUR 2-3, COMMUNICATIE SERVICES. 15
FIGUUR 2-4, CLOUD SERVICES. 16
FIGUUR 2-5, VOORBEELDEN VAN IAAS, PAAS, SAAS EN PAAS. 17
FIGUUR 2-6, CLOUD SERVICE PROVIDER. 17
FIGUUR 2-7, CLOUD BROKER 18
FIGUUR 2-8, AZURE SERVICES. 18
FIGUUR 2-9, AWS SERVICES. 19
FIGUUR 2-10, DATA CENTERS. 19
FIGUUR 2-11, IT LANDSCHAP – DE SAMENHANG. 20
FIGUUR 3-1, HET WATERVAL ONTWIKKELPROCES. 24
FIGUUR 3-2, HET AGILE ONTWIKKELPROCES. 25
FIGUUR 3-3, REFERENTIEARCHITECTUUR MODEL. 26
FIGUUR 3-4, MONOLIET. 26
FIGUUR 3-5, MICROSERVICES. 27
FIGUUR 3-6, REST API. 27
FIGUUR 3-7, APPLICATIECOMMUNICATIE. 28
FIGUUR 3-8, ENTERPRISE SERVICE BUSS COMMUNICATIE. 28
FIGUUR 3-9, CONTAINER PATTERN. 29
FIGUUR 3-10, PROGRAMMEERTALEN. 34
FIGUUR 3-11, SAMENHANG SOFTWARE ONTWIKKELEN. 35
FIGUUR 4-1, DRIEVOUDIG BEHEERMODEL, BRON [LOOIJEN 2014]. 40
FIGUUR 4-2, ITIL V2. 40
FIGUUR 4-3, DE ITIL SERVICE SUPPORT SET – BRON: [OGC - 2000], [FIGUUR VAN THIADENS]. 41
FIGUUR 4-4, DE ITIL SERVICE DELIVERY SET – BRON: [OGC - 2000], [FIGUUR VAN THIADENS]. 42
FIGUUR 4-5, ITIL V3. 44
FIGUUR 4-6, ITIL SERVICE VALUE SYSTEM. 50
FIGUUR 4-7, ITIL SERVICE VALUE SYSTEM. 51
FIGUUR 4-8, DEVOPS LEMNISCAAT. 52
FIGUUR 4-9, SAMENHANG SOFTWARE BEHEREN. 54
FIGUUR 5-1, CONTINUOUS LEARNING MODEL. 56
FIGUUR 5-2, AI MODELLEN. 57
FIGUUR 5-3, SAMENHANG. 59

Tabellen

TABEL 1-1, BIJLAGEN. 4
TABEL 2-1, IT LANDSCHAP AFKORTINGEN. 7
TABEL 2-2, SYSTEEMPROGRAMMATUUR. 11
TABEL 2-3, HARDWARE COMPONENTEN. 14
TABEL 2-4, RANDAPPARATUUR. 14
TABEL 3-1, SOFTWARE ONTWIKKELEN AFKORTINGEN. 22
TABEL 3-2, SOFTWARE ONTWIKKELEN AFKORTINGEN. 23
TABEL 4-1, SOFTWAREBEHEER AFKORTINGEN. 39

TABEL 5-1, TRENDS IN DE IT AFKORTINGEN 55

Bijlagen

BIJLAGE A, LITERATUURLIJST 63
BIJLAGE B, BEGRIPPENLIJST 67
BIJLAGE C, AFKORTINGEN 83
BIJLAGE D, AGILE TOOLS 87
BIJLAGE E, WEBSITES 91
BIJLAGE F, INDEX 93

Voorwoord

Dit boek is samengesteld op basis van mijn ervaringen in de IT. De grootste uitdaging bij de samenstelling van dit boek is de kunst van het weglaten. Er is zoveel te vertellen over IT, maar wat zijn nu de dingen die er toe doen en hoe kunnen deze zo tijdloos mogelijk worden uitgelegd zonder betekenisloos te worden.

Gezien de snelheid waarmee de wereld van IT zich ontwikkelt en de behoefte om u met zo min mogelijk tekst zo veel mogelijk beelden te geven wat IT inhoudt heb ik besloten dit boek Agile te houden. Dit houdt in dat het zeer beknopt beschrijft wat belangrijke inzichten zijn, die ik heb opgedaan gedurende mijn rol als consultant, trainer, coach en examinator ten aanzien van IT. Waar van toepassing verwijs ik naar bronnen die ik zelf heb geraadpleegd om mij verder te bekwamen. Hierbij besef ik dat de gepresenteerde basiskennis IT een momentopname is die wellicht door de toenemende snelheid van innovatie weer snel achterhaald is.

Veel van mijn ervaringen heb ik ook al gedeeld in de artikelen op www.ITpedia.nl. Tevens heb ik de kennis en kunde vertaald naar diverse trainingen die ik verzorg. Deze zijn te vinden op www.dbmetrics.nl.

Hierbij dank ik de volgende personen van harte voor hun inspirerende bijdrage aan dit boek en de fijne samenwerking!

- D. (Dennis) Boersen — Argis IT Consultants
- J.A.E. (Jane) ten Have — -
- Dr. L.J.G.T. (Louis) van Hemmen — BitAll B.V.
- F. (Freek) de Cloe — smartdocs.com
- J.W. (Jan-Willem) Hordijk — Digital & Transformation Manager Nordics TKE
- N (Niels) Talens — www.nielstalens.nl
- W. (Willem) Kok — Argis IT Consultants

Ik wens u veel plezier toe bij het lezen van dit boek en vooral veel succes bij het toepassen van de Basiskennis IT binnen uw eigen organisatie.

Mocht u vragen of opmerkingen hebben, aarzel dan vooral niet om met mij contact op te nemen. Er is veel tijd besteed om dit boek zo compleet en consistent mogelijk te maken. Mocht u toch tekortkomingen aantreffen, dan zou ik het op prijs stellen als u mij daarvan in kennis stelt, dan kunnen deze zaken in de volgende editie verwerkt worden.

1 Introductie

Leeswijzer:
Dit hoofdstuk beschrijft het doel van dit boek (1.1) de beoogde doelgroep (1.2), de achtergrond van dit boek (1.3), de structuur (1.4) de bijlagen (1.5) ten slotte enkele tips voor het hanteren van dit boek in (1.6).

1.1 *Doel*

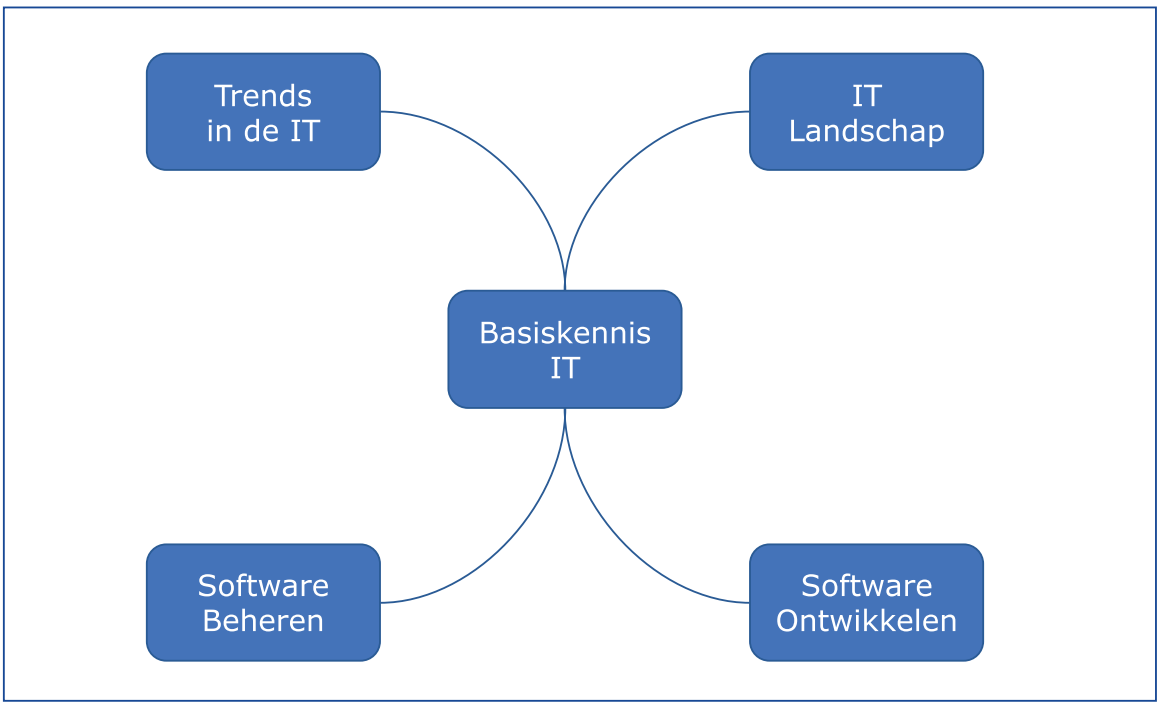
De doelstelling van dit boek is het bespreken van de basiskennis ten aanzien van Informatie Technology (IT). De scope van de basiskennis is het landschap, het ontwikkelen en beheren van software en de onderliggende infrastructuur en de trends in de IT.

1.2 *Doelgroep*

De doelgroep van dit boek zijn alle mensen die pas beginnen in de IT zoals studenten en mensen die zich willen omscholen

1.3 *Structuur*

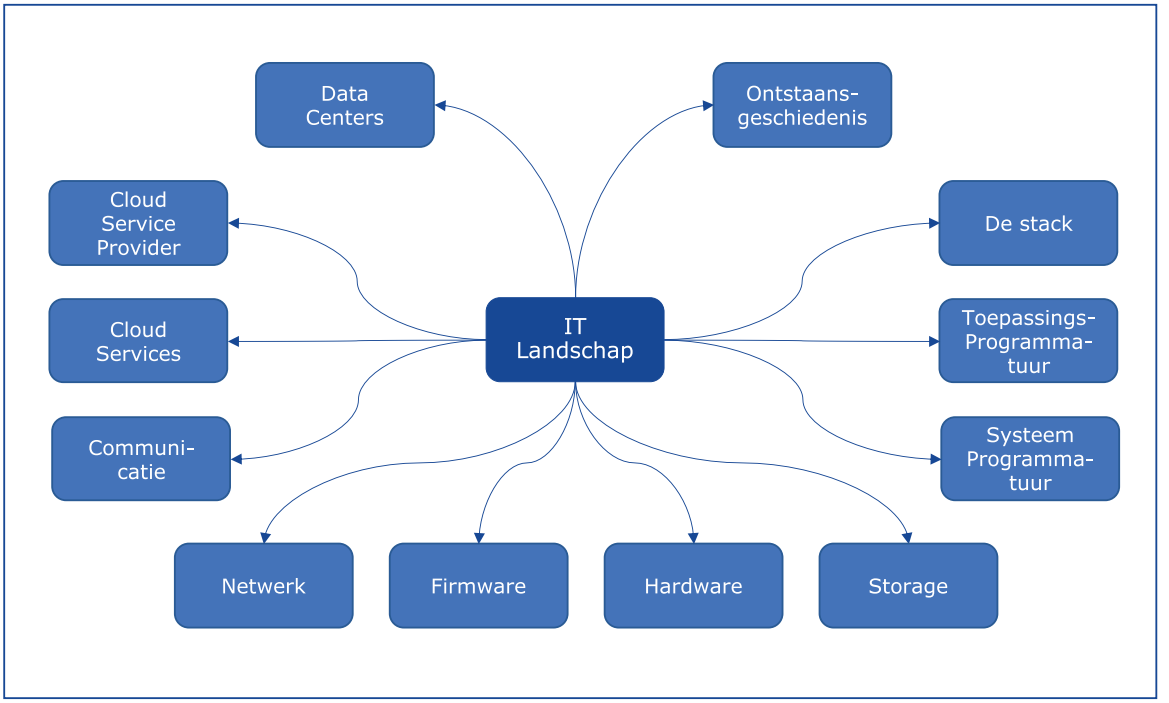
Dit boek beschrijft de IT vanuit vier perspectieven zoals weergegeven in Figuur 1-1.



Figuur 1-1, De vier perspectieven van IT.

1.3.1 *Hoofdstuk 2: IT Landschap*

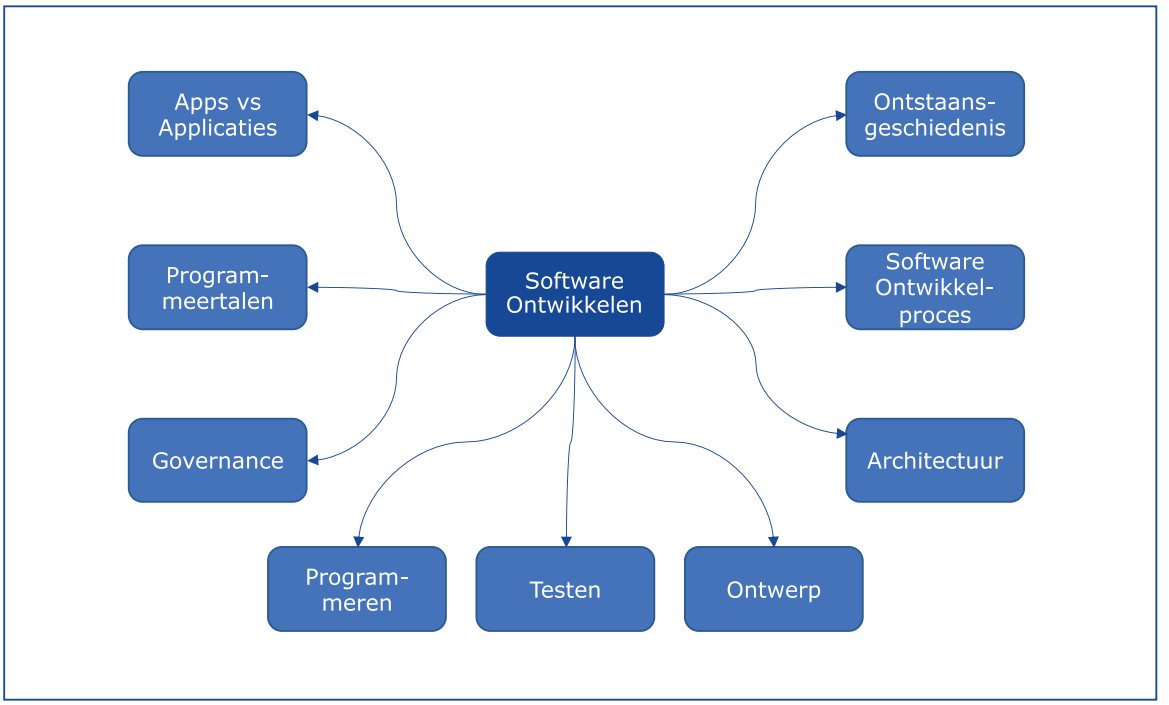
Dit hoofdstuk bespreekt wat er zo al aan objecten in de wereld van IT wordt onderkend zonder daarin uitputtend te willen zijn. De nadruk ligt op de software en de objecten die nodig zijn om de software te laten werken. Daarnaast wordt een blik geworpen op de Cloud oplossingen. In Figuur 1-2 is een overzicht gegeven van de betrokken onderwerpen.



Figuur 1-2, Basiskennis - IT Landschap.

1.3.2 *Hoofdstuk 3: Software ontwikkelen*

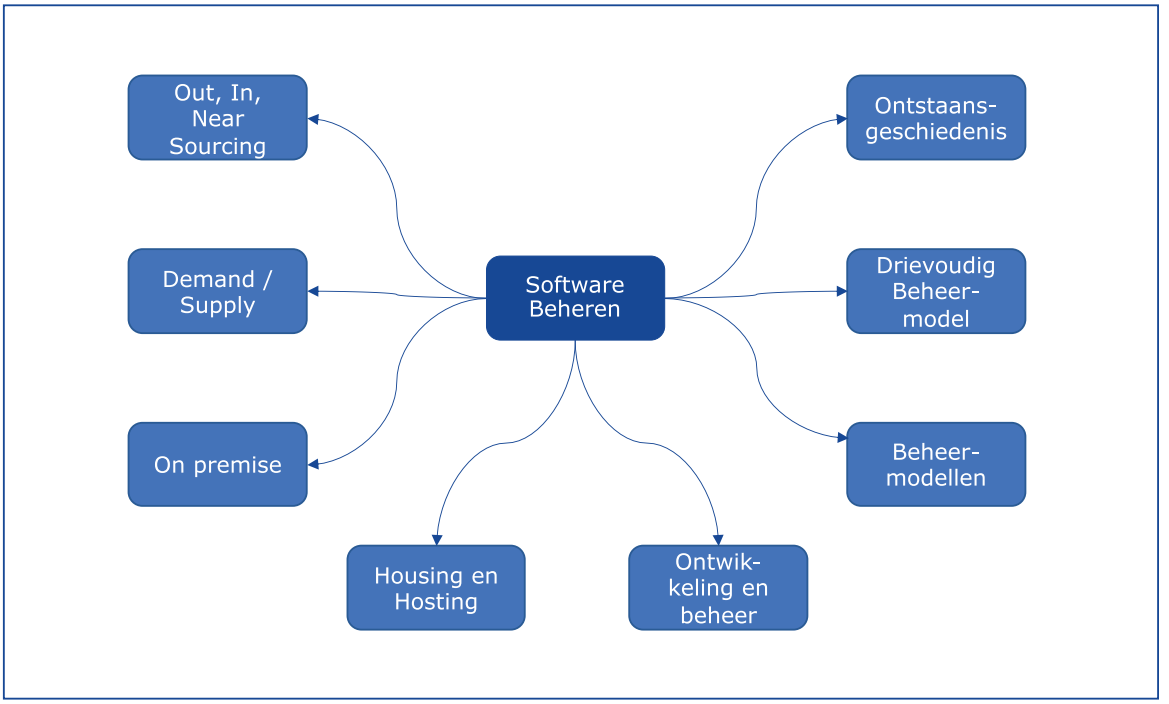
In Figuur 1-3 zijn de onderwerpen weergegeven die behandeld worden ten aanzien van het ontwikkelen van software. Er wordt vanuit een historisch perspectief gekeken hoe software te ontwikkelen.



Figuur 1-3, Basiskennis - Software Ontwikkelen.

1.3.3 *Hoofdstuk 4: Software beheren*

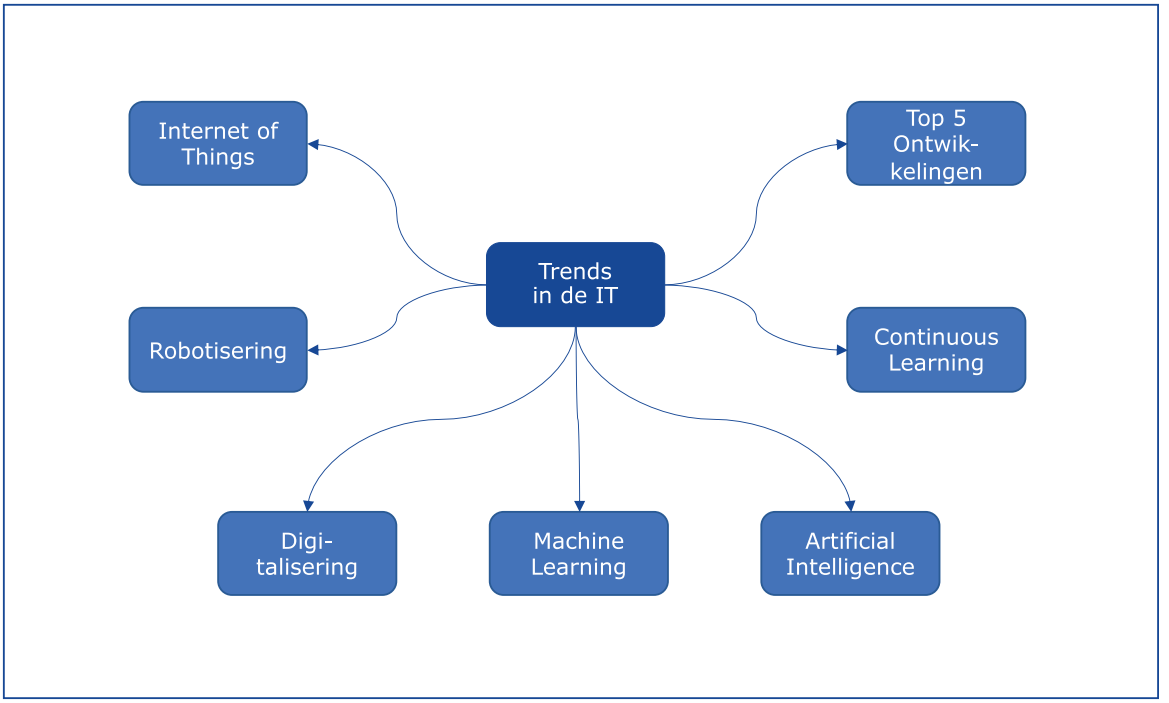
Software moet niet alleen ontwikkeld of gekocht worden maar ook beheerd worden. Hiertoe zijn een aantal beheermodellen beschikbaar die uitleggen wat er zoal gedaan moet worden. De software vereist ook beheer op andere gebieden. Deze worden meegenomen in dit hoofdstuk. In Figuur 1-4 is een overzicht gegeven van de gerelateerde onderwerpen.



Figuur 1-4, Basiskennis - Software Beheren.

1.3.4 *Hoofdstuk 5: Trends in de IT*

In Figuur 1-5 is een overzicht gegeven van trends in de IT. Dit hoofdstuk bespreekt deze onderwerpen en legt de relaties daartussen.



Figuur 1-5, Basiskennis – Trends in de IT.

1.4 *Bijlagen*

De bijlagen bevatten informatie die helpt bij het beter leren kennen van het IT vakgebied.

Bijlage A, Literatuurlijst

In Tabel A-1 is een overzicht gegeven van boeken die direct of indirect zijn gerelateerd aan DevOps.

Referenties	Publicaties
Best 2019 ADN	B. de Best, "Agile Design", Dutch language, Leonon Media, 2020, ISBN 13: 978 94 92618 375. English language, Leonon Media, 2019, ISBN13: 978 94 92618 184.
Best 2014 ACC	B. de Best, "Acceptatiecriteria", Dutch language, Leonon Media, 2014, ISBN 13: 978 90 71501 784.
Best 2011 SLA	B. de Best, "SLA best practices", Dutch language, Leonon Media 2011, ISBN13: 978 90 71501 456.
Best 2012 Q&A	B. de Best, "Quality Control & Assurance", Dutch language, Leonon Media 2012, ISBN13: 978 90 71501 531.
Best 2014 ASM	B. de Best, "Agile Service Management met Scrum", Dutch language, Leonon Media, 2014, ISBN13: 978 90 71501 807. English language, Leonon Media, 2014, ISBN13: 978 94 9261 8085.
Best 2014 CSL	B. de Best, "Cloud SLA, Leonon Media, Dutch language, 2014, ISBN13: 978 90 71501 739. English language, 2014, ISBN13: 978 94 9261 8009.
Best 2015 ASM	B. de Best, "Agile Service Management met Scrum in de Praktijk", Dutch language Leonon Media, 2015, ISBN13: 978 90 71501 845. English language Leonon Media, 2015, ISBN13: 978 94 9261 8177.
Best 2015 KTB	B. de Best, "Ketenbeheer in de Praktijk", Dutch language, Leonon Media 2015, ISBN13: 978 90 71501 852.
Best 2015 PRI	B. de Best, "Prestatie Indicatoren", Dutch language, Leonon Media 2015, ISBN13: 9789071501470.
Best 2017 SLT	B. de Best, "SLA templates", English language, Leonon Media 2017, ISBN13: 978 94 92618 030.
Best 2017 BOA	B. de Best, "Beheren onder Architectuur", Dutch language, Leonon Media, 2017, ISBN13: 978 90 71501 913.
Best 2017 DOB	B. de Best, "DevOps best practices", English language, Leonon Media, 2017, ISBN13: 978 94 92618 078.
Best 2019 DOA	B. de Best, "DevOps Assessments", Dutch language, Leonon Media, 2019, ISBN13: 978 90 71501 814.
Best 2019 DOD	B. de Best, "DevOps Development", Dutch language, Leonon Media, 2019, ISBN13: 978 94 92618 054. English language, Leonon Media, 2019, ISBN13: 9789492618184.
Best 2019 DOR	B. de Best, "DevOps Architectuur", Dutch language, Leonon Media, 2019, ISBN13: 978 94 92618 061. English language, Leonon Media, 2019, ISBN13: 978 90 71501 579
Best 2015 BKI	B. de Best, "Basiskennis IT", Dutch language, Leonon Media, 2015, ISBN13: 978 94 92618 573.
Best 2021 CP	B. de Best. "Continuous Planning", Dutch Language, Leonon Media, 2021, ISBN 13: 978 94 92618 504
Best 2021 CN	B. de Best. "Continuous Design", Dutch Language, Leonon Media, 2021, ISBN 13: 978 94 92618 481
Best 2021 CT	B. de Best. "Continuous Testing", Dutch Language, Leonon Media, 2021, ISBN 13: 978 94 92618 450

Referenties	Publicaties
Best 2021 CI	B. de Best. "Continuous Integration", Dutch Language, Leonon Media, 2021, ISBN 13: 978 94 92618 467
Best 2021 CD	B. de Best. "Continuous Deployment", Dutch Language, Leonon Media, 2021, ISBN 13: 978 94 92618 511
Best 2021 CM	B. de Best. "Continuous Monitoring", Dutch Language, Leonon Media, 2021, ISBN 13: 978 94 92618 498
Best 2021 CL	B. de Best. "Continuous Learning", Dutch Language, Leonon Media, 2021, ISBN 13: 978 94 92618 528
Best 2021 CR	B. de Best. "Continuous Robotics", Dutch Language, Leonon Media, 2021, ISBN 13: 978 94 92618 535
Best 2021 CA	B. de Best. "Continuous Auditing", Dutch Language, Leonon Media, 2021, ISBN 13: 978 94 92618 542
Best 2021 CS	B. de Best. "Continuous Assessment", Dutch Language, Leonon Media, 2021, ISBN 13: 978 94 92618 474
Bloom 1956	Benjamin S. Bloom, "Taxonomy of Educational Objectives (1956)", Allyn and Bacon, Boston, MA. Copyright (c) 1984 by Pearson Education.
Boehm	Boehm B. Software Engineering Economics, Prentice Hall, 1981
Caluwé 2011	L. de Caluwé en H. Vermaak, "Leren Veranderen", Kluwer, 2011, tweede druk, ISBN13: 978 90 13016 543.
Davis 2016	Jennifer Davis, Katherine Daniels, "Effective DevOps Building a Culture of Collaboration, Affinity, and Tooling at Scale", O'Reilly Media; 1 edition, ISBN-13: 978 14 91926 307, 2016.
Deming 2000	W. Edwards Deming, "Out of the Crisis. MIT Center for Advanced Engineering Study", 2000, ISBN13: 978 02 62541 152.
Downey 2015	Allen. B. Downey, "Think Python", O'Reilly Media, Inc, Usa; Druk 2, ISBN-13: 978 14 91939 369, 2015
Galbraith 1992	Galbraith, J.R. "Het ontwerpen van complexe organisaties", 1992, Alphen aan de Rijn: Samson Bedrijfsinformatie.
Humble 2010	Jez Humble, David Farley "Continuous Delivery Reliable Software Releases through Build, Test, and Deployment Automation", Addison-Wesley Professional; 1 edition, ISBN-13: 978 03 21601 919, 2010.
Kim 2014	Gene Kim, Kevin Behr, George Spafford "The Phoenix Project", IT Revolution Press, ISBN-13: 978 09 88262 508, 2014.
Kim 2016	Gene Kim, Jez Humble "The DevOps Handbook: How to Create World-Class Agility, Reliability, and Security in Technology Organizations, Patrick Debois, John Willis", IT Revolution Press, ISBN-13: 978 19 42788 003, 2016.
Kotter 2012	John P. Kotter, "Leading Change", Engels 1e druk, november 2012, ISBN13: 978 14 22186 435.
Kaplan 2004	R. S. Kaplan en D. P. Norton, "Op kop met de Balanced Scorecard", 2004, Harvard Business School Press, ISBN13: 978 90 25423 032.
Layton 2017	Mark C. Layton Rachele Maurer, "Agile Project Management for Dummies", tweede druk, John Wiley & Sons Inc, 2017, ISBN13: 978 11 19405 696.
Looijen 2014	M. Looijen, L. van Hemmen, "Beheer van Informatiesystemen", zevende druk, Academic Service, 2014, ISBN13: 978 94 62450 936.
MAES	R. Maes, "Visie op informatiemanagement", www.rikmaes.nl

Referenties	Publicaties
McCabe	McCabe T. "A Complexity Measure" in: IEEE Transactions on Software Engineering 1976, vol. 2, nr. 4.
Michael Porter	M.E. Porter 'Competitive Advantage: Creating and Sustaining Superior Performance, Simon & Schuster ,1998, ISBN13: 978 06 84841 465
Oirsouw 2001	R.R. van Oirsouw, J. Spaanderman, C. van Arendonk, "Informatiserings-economie", ISBN 90 395 1393 7, 2001.
Scrum	Ken Schwaber and Jeff Sutherland, "The Scrum Guide™", 2017, www.scrumguides.org.
Schwaber 2015	K. Schwaber, "Agile Project Management with Scrum", Microsoft Press, ISBN13: 978 07 35619 937.
Toda 2016	(Luke) Toda, President Strategic Staff Services Corporation and Director of TPS Certificate Institution Nobuyuki Mitsui, CTO of Strategic Staff Services Corporation, "Success with Enterprise DevOps Koichiro" "White Paper", 2016.

Tabel A-1, Literatuurlijst.

Bijlage B, Begrippenlijst

In Tabel B-1 is een begrippenlijst opgenomen. Deze is in het Engels opgesteld omdat zeer veel termen uit de Engelse taal afkomstig zijn en de uitleg makkelijker leest als de hele uitleg in het Engels is opgenomen.

Begrip	Betekenis
5S	Japan's principle of order and cleanliness. These Japanese terms with their Dutch equivalent are: Seiri (整理): Sort Seiton (整頓): Arrange Seisō (清掃): Cleaning Seiketsu (清潔): Standardize Shitsuke (躾): Hold or Systematize [Wiki]
A/B testing	A/B testing means that two versions of an application or webpage are taken into production to see which performs better. Canary releasing can be used, but there are also other ways to perform A/B testing.
Acceptance test	For DevOps engineers the acceptance testcases gives the answer "How do I know when I am done?". For the users the acceptance testcases gives the answer "Did I get what I wanted?". Examples of acceptance testcases are Functional Acceptance Testcases (FAT), User Acceptance Testcases (UAT) and Production Acceptance Testcases (PAT). The FAT and UAT should be expressed in the language of the business.
Affinity	DevOps is about collaboration and affinity. Where collaboration is focused on the relationship between individuals in a DevOps team, affinity goes one step further. This DevOps pillar is about shared organizational goals, empathy and learning between different groups of people by sharing stories and learn from each other.
Agile Infrastructure	Within DevOps both Development and Operations work in an Agile way. This requires an Agile Infrastructure that can be changed with the same pace as the application is changed through the deployment pipeline. A good example of an Agile Infrastructure is the use of Infrastructure as Code.
Alternate path	See happy path.
Andon cord	In the Toyota manufacturing plant, above every work centre a cord is installed. Every worker and manager are trained to pull when something goes wrong; for example, when a part is defective, when a required part is not available, or even when work takes longer than planned. When the Andon cord is pulled, the team leader is alerted and immediately works to resolve the problem. If the problem cannot be resolved within a specified time (e.g., fifty-five seconds), the production line is stopped so that the entire organization can be mobilized to assist with problem resolution until a successful countermeasure has been developed [Kim 2016].
Anomaly detection techniques	Not all data that needs to be monitored has a Gaussian (normal) distribution. The anomaly detection techniques make it possible to find noteworthy variances using a variety of methods for data that has no Gaussian distribution. These techniques are either used in monitoring tools or require people with statistical skills.

Begrip	Betekenis
Antifragility	This is the process of applying stress to increase resilience. This term is introduced by author and risk analyst Nassim Nicholas Taleb.
Anti-pattern	An anti-pattern is an example of the wrong interpretation of a pattern. The anti-pattern is often used to explain the value of the pattern.
Artefact	An artefact is a product that is manufactured. Within DevOps the output of the commit phase are binaries, reports and meta data. These products are also referred to as artefacts.
Artefact repository	The central storage of artefacts is called the artefact repository. The artefact repository is used to managed artefacts and their dependencies.
Automated tests	Testcases should be automated as much as possible to reduce waste and to increase velocity and quality of the products that are to be delivered.
Bad apple theory	People that believe in the 'Bad Apple Theory' think that a system is basically safe if it were not for those few unreliable people in it. By removing these people, the system will be safe. This results in the anti DevOps pattern of 'name, blame, shame'.
Bad paths	A 'bad path' is a situation where the application does not follow the 'happy path' or 'the alternate' path. In other words, something goes wrong. This exception must be handled and should be monitorable.
Behavior driven development	The development of software requires that the users are asked to define the (non) functional requirements. Behavior driven development is based on this concept. The difference however is that the acceptance criteria of these requirements should be written in the customer's expectation of the behavior of the application. This can be accomplished by formulating the acceptance criteria in the Given – When – Then format.
Binary	A compiler is used to transform source code to objectcode. The objectcode is also known as a binary. The source code is readable for human being, the objectcode however is only readable for computers since they have been written in hexadecimals.
Blamelessness	This approach is about learning rather than punishing. Within DevOps this is one of the basic ideas of learning from mistakes. The energy of the DevOps team is spending on learning from the mistake, rather than on finding the one to blame.
Blameless post mortem	Blameless post-mortem is a term coined by John Allspaw. It helps to examine "mistakes in a way that focuses on the situational aspects of a failure's mechanism and the decision-making process of individuals proximate to the failure." [Kim 2016].
Blue-Green deployment pattern	Blue and green refer to two identical production systems. One is used for the final acceptance of a new release. If this acceptance is successful, then this environment becomes the new production environment. In case of a failure of the production system, the other system can be used instead. This mitigates the risk of downtime since the switchover is likely to be less than a second.
Broken build	A build that fails due to an error in the application source code.
Brown field	There are two scenarios' for applying DevOps best practices: green field and brown field. In case of a green field scenario the whole DevOps organisation has to be established from scratch. The opposite scenario is where there is already a DevOps organisation, but improvements are needed.

Begrip	Betekenis
	The colour green refers to the situation that a factory is built on a clean grass field. The colour brown refers to the situation that a factory is to be built on a place where there has already been a factory that poisoned the ground. In order to build on a brown field, the poison needs to be removed.
Business value	Applying DevOps best practices results in increasing the business value. Research of Puppet Labs (State Of DevOps Report) proofs that high-performing organizations using DevOps practices are outperforming their non–high performing peers in many following areas [Kim 2016].
Canary releasing pattern	Normally a release is offered to every user at once. Canary releasing is the approach in which a small set of users is receiving the new release. If this small scope release works fine than the release can be deployed to all users. The term canary refers to the old habit to have a canary in the coal mines to detect toxic gas.
Change categories	Changes can be categorized into standard changes, normal changes and urgent changes.
Change schedules	Changes can be scheduled in order to defined in which order they have to be applied.
Cloud configuration files	Cloud configuration files are used to initiate a cloud service before using it. In this way cloud service providers enable customers to configure the cloud environment for their needs.
Cluster immune system release pattern	The cluster immune system expands upon the canary release pattern by linking our production monitoring system with our release process and by automating the roll back of code when the user-facing performance of the production system deviates outside of a predefined expected range, such as when the conversion rates for new users drops below our historical norms of 15%–20% [Kim 2016].
Code branch	See branching.
Code review forms	Code review can be performed in several ways like "over the shoulder", pair programming, email pass-around and tool-assisted code review.
Codified NFR	A list of Non-Functional Requirements (NFR) that are categorised in categories like availability, capacity, security, continuity et cetera.
Collaboration	One of the four pillars of DevOps is collaboration. Collaboration refers to the way the individuals of a DevOps team works together to achieve the common goal. There are many forms in which this collaboration comes to expression like: • peer to peer programming; • demonstrating weekly progress; • documentation; et cetera.
Commit code	Committing code is the action in which the DevOps engineer adds the changed source code to the repository, making these changes part of the head revision of the repository [Wiki].
Commit stage	This is the phase in the CI/CD secure pipeline where the source code is compiled to the objectcode. This includes the performance of the unit testcases.

Begrip	Betekenis
Compliance checking	The manual action of a security officer to make sure that the system is built in accordance with the agreed standards. This is the opposite of security engineering where the DevOps teams works together with the security officer in order to embed the agreed standards in the deliverables and enable continuous monitoring of the standard in the whole lifecycle of the product.
Compliancy officer	The compliancy officer is a DevOps role. The compliancy officer is responsible for ensuring compliance with agreed standards throughout the whole life cycle of a product.
Configuration management	Configuration Management refers to the process by which all artefacts, and the relationships between them, are stored, retrieved, uniquely identified and modified.
Containers	A container is an isolated structure that is used by DevOps engineers to build their application independently from the underlying operating system or hardware. This is accomplished by interfaces in the container that are used by DevOps engineers. Instead of installing the application in an environment, the complete container is deployed. This saves a lot of dependencies and prevents configuration errors to occur.
Conway's law	The following statement of Melvin Conway is called the Conway's law: "organizations which design systems ... are constrained to produce designs which are copies of the communication structures of these organizations." [Wiki].
Cultural debt	There are three forms of debt. Cultural debt, technical debt and information debt. This form of debt refers to the decision to keep flaws in the organization structure, hiring strategy, values et cetera. This debt costs interest and will result in less maturity growth of the DevOps teams. Cultural debt can be recognized by the exitance of extensive silos, workflow constraints, miscommunications, waste et cetera.
Culture, Automation Measurement, Sharing (CAMS)	CAMS is the abbreviation for Culture, Automation, Measurement and Sharing. • Culture: Culture relates to the people and process aspects of DevOps. Without the right culture, automation attempts will be fruitless. • Automation: Release management, configuration management, and monitoring and control tools should enable automation. • Measurement: 'If you can't measure it, you can't manage it.' & 'If you can't measure it, you can't improve it'. • Sharing: Culture of sharing ideas and problems is critical to help organizations to improve. Creates feedback loop.
Cycle time (flow time)	Cycle time measures more the completion rate or the work capability of a system overall, and a shorter cycle time means that less time is being wasted when a request has been made but no progress or work is getting done.
Cycle time (lean)	The average time between two successive units leaving the work or manufacturing process.
Declarative programming	This is a programming paradigm that expresses the logic of a computation without describing its control flow. An example are the database query languages for example TSQL and PSQL.

Begrip	Betekenis
Defect tracking	Defect tracking is the process of tracking the logged defects in a product from beginning to closure and making new versions of the product that fix the defects [Wiki].
Development rituals	The Agile Scrum rituals of development are the sprint planning, daily stand-up, sprint execution, review and the retrospective.
Development	Development is an activity that is performed by the DevOps role 'DevOps engineer'. A DevOps engineer is responsible for the complete lifecycle of a configuration item. Within DevOps there is no difference anymore between designer, builder or tester.
Downward spiral	Gene Kim explains in his book [Kim 2016] that the downward spiral in Information Technology (IT) has three acts. • The first act begins in IT Operations where technical debt results in jeopardizing our most important organizational promises. • The second act starts with compensating the latest broken promise by promising a bigger, bolder feature or an even larger revenue target. As a result, Development is tasked with another urgent project which results in even more technical debt. • The third stage is where the deployments are getting slower and slower, and outages are increasing. The business value continuously decreases.
E-mail pass-around	E-mail pass-around is a review technique where the source code management system emails code to reviewers automatically after the code is checked in [Kim 2016].
Error path	See happy path.
Fast feedback	Fast feedback refers to the second way of the three ways of Gene Kim. The second way is about having feedback on the functionality and quality of the product that is created or modified as soon as possible in order to maximize the business value.
Feature toggles	A feature toggle is a mechanism that makes it possible to enable or disable a part of the functionality of an application released in production. Feature toggles enables testing the effect of changes on users in production. Feature Toggles are also referred to as Feature Flags, Feature Bits or Feature Flippers.
Feed forward	Feed forward within the context of DevOps is the mechanism by which experiences in the present value stream are used to improve the future value stream. Feed forward is the opposite of feedback since feedback is focused on the past and feed forward on the future.
Feedback	Feedback within the context of DevOps is the mechanism by which errors in the value stream are detected as soon as possible and is used to improve the product and if necessary to improve the value stream as well.
Feed forward	Feed forward within the context of DevOps is the mechanism by which experiences in the present value stream are used to improve the future value stream. Feed forward is the opposite of feedback since feedback is focused on the past and feed forward on the future.
Gaussian distribution	In probability theory, the normal (or Gaussian) distribution is a very common continuous probability distribution. Normal distributions are important in statistics and are often used in the natural and social sciences to represent real-valued random variables whose distributions are not known. A random variable with a Gaussian distribution is said to be normally distributed and is called a normal deviate [Wiki].

Begrip	Betekenis
Given-When-Then	The Given-When-Then format is used to define acceptance criteria in a way that the stakeholders understand how the functionality actually will work. GIVEN – the fact that… WHEN – I do this… THEN – this happens
Green field	See brown field.
Hand-off Readiness Review (HRR)	The HRR term is introduced by Google. An HRR is set of safety checks for a critical stage of releasing new services. HRR is performed when a service is transitioned from a developer-managed state to an OPS-managed state (usually months after the LRR). HRR makes service transition easier and more predictable and helps create empathy between upstream and downstream work centers.
Happy path	An application supports a business process by receiving, editing, storing and providing information. The assumed steps in which the information processing is performed is called the happy path. The steps in alternate ways are called the alternate path. In that case, the same result will be achieved via another navigation path. The crawl of the application that causes an error is called an error path.
Holocrazy	In this type of organization all decisions are made through self-organizing teams rather than through a traditional management hierarchy.
Horizontal splitting of features	A feature can be splitted into stories. Horizontal splitting refers to the result of a feature splitting in which more DevOps teams must work tightly together. They have to align their work continuously in order to deliver together the feature.
Idempotent	Continuous delivery requires that a component can always to be brought fully automatically to the desired status regardless of the component's initial state and regardless of the number of times the component is configured. The characteristic of a component to always be able to get back into the desires is called idempotent.
Imperative programming	This is a programming paradigm that uses statements that change a program's state. Imperative programming focuses on how a program should operate and consists of commands for the computer to perform. Examples are COBOL, C++, BASIC et cetera. The term is often used in contrast to declarative programming, which focuses on what the program should accomplish without specifying how the program should achieve the result.
Infrastructure as Code (IaC)	Normally infrastructure components have to be configured in order to perform the requested functionality and quality for example a rule set for a firewall or the allowed IP addresses for a network. These configurations normally are stored in configuration files which enable the operators to manage the functionality and the quality of the infrastructure components. Infrastructure as code (IaC) makes it possible to programme these infrastructure component settings and deploy these settings through the CI/CD secure pipeline by the use of machine-readable definition files, rather than physical hardware configuration or interactive configuration tools.
Infrastructure management	Infrastructure management consists of the lifecycle management of all infrastructure products and services in order to support the correct working of the applications that run on top of the infrastructure.

Begrip	Betekenis
Independent, Negotiable, Valuable, Estimable, Small, and Testable (INVEST)	Independent, Negotiable, Valuable, Estimable, Small, and Testable. • Independent: The product backlog item should be self-contained, in a way that there is no inherent dependency on another product backlog item. • Negotiable: Product backlog items, up until they are part of an iteration, can always be changed, rewritten or even discarded. • Valuable: Product backlog item must deliver value to the stakeholders. • Estimable: The size of a product backlog item must always estimable. • Small: Product backlog items should not be so big as to become impossible to plan / task / prioritize with a certain level of certainty. Testable: The product backlog item or its related description must provide the necessary information to make test development possible.
Information radiators	An Information Radiator is a visual display that a team places in a highly visible location so that all team members can see the latest information at a glance.
Infrastructure as Code (IaC)	Normally infrastructure components have to be configured in order to perform the requested functionality and quality for example a rule set for a firewall or the allowed IP addresses for a network. These configurations normally are stored in configuration files which enable the operators to manage the functionality and the quality of the infrastructure components. Infrastructure as code (IaC) make it possible to programme these infrastructure component settings and deploy these settings through the deployment pipeline by the use of machine readable definition files, rather than physical hardware configuration or interactive configuration tools.
Infrastructure management	Infrastructure management consists of the lifecycle management of all infrastructure products and services in order to support the correct working of the applications that run on top of the infrastructure.
Infrastructure as code	Infrastructure as code (IaC) is a software-based approach to the ICT infrastructure, whereby the systems can be rolled out and adapted in a consistent manner through templates. If a change has to be made, it is implemented in the template which is then rolled out again.
I-shaped, T-shaped, E-shaped	I-shaped, T-shaped, E-shaped are the categories to indicate the knowledge and special skills of a person. An I-shaped person is a pure specialist in one area. The T-shaped person has special skills in one field and broad general knowledge. The E-shaped person has special skills in more than one field and broad general knowledge.
Infosec	A team that is responsible for securing systems and data.
Just In Time (JIT)	JIT means building up a stream-lined supply chain with one-piece flow.
Ji-Kotei-Kanketsu (JKK)	JKK which means 100% completion of an item. This quality way of working means: • clear understanding of the goals; • understanding the right way to work; • ensure high quality of work;

Begrip	Betekenis
	• getting the work right for 100% completion, never pass defects to the next process; • Definition of Done (DoD) is vital; • and then maintaining the required quality without inspections.
Kaizen	Kaizen is Japanese for "improvement". Kaizen is used to improve production systems. The goals of kaizen are: • elimination of waste (muda's); • JIT; • standardization of production; • cycle of continuous improvements. Continuous improvement means circulate the Plan-Do-Check-Act (PDCA) cycle daily, weekly. This can be accomplished by finding the root cause of a failure by asking "Why" 5 times. The following steps can be followed: • defining problems with supporting data; • making sure everybody recognizes the problems clearly; • setting a hypothesis on the problems found; • defining countermeasure actions to verify the hypothesis; • defining countermeasure actions be in daily based activities; • measuring a weekly KPI so people can feel a sense of accomplishment.
Kaizen Blitz (or Improvement Blitz)	A Kaizen Blitz is a rapid improvement workshop designed to produce results / approaches to discrete process issues within a few days. It is a way for teams to carry out structured, but creative problem solving and process improvement, in a workshop environment, over a short timescale.
Kaizen in advance	Kaizen in advance goes one step further than Kaizen. Not only the own activities are improved but also the activities that are performed upstream and that lead to problems downstream. In this way a feedback loop of problems is created which improves the system as a whole.
Kanban	This is system to signal when something is needed. Kanban is a system for managing the logistics production chain. Kanban was developed by Taiichi Ohno, at Toyota, to find a system that made it possible to achieve a high level of production. Kanban is often used for application management. One of the characteristics of Kanban is that it is pull oriented which means that there is not stock of material to be used during the production. Kanban can be used to implement JIT in production systems.
Kata	A kata is any structured way of thinking and acting (pattern of behavior) that is practiced until the pattern becomes a second nature. Four steps can be recognised to accomplish this second nature: • direction (target); • current condition (IST situation); • target condition (SOLL situation); • PDCA (Deming wheel). From an architectural viewpoint the migration path might be added to Kata as well. The migration path shows the way to go in order to achieve the Soll situation.
Kibana dashboards	A Kibana dashboard displays a collection of saved visualizations.

Begrip	Betekenis
Latent defects	Problems that are not visible yet. Latent defects can be made visible by injecting faults into the system.
Launching guidance	To prevent the possibility of problematic, self-managed services going into production and creating organizational risk, launch requirements may be defined that must be met in order for services to interact with real customers and be exposed to real production traffic [Kim 2016].
Launch Readiness Review (LRR)	The LRR term is introduced by Google. An LRR is a set of safety checks for a critical stage of releasing new services. It is performed and signed off before a service is made publicly available and receive live production traffic. LRR is self-reported by the project teams. LRR is used in the development-managed state.
Lead Time (LT)	Lead time is the time from when a request is made to when the final result is delivered, or the customer's point of view on how long something takes to complete.
Lean tools	• A3 thinking (problem solving) • Continuous flow (eliminates waste) • Kaizen • Kanban • KPI (Key Performance Indicator) • Plan Do Check Act (PDCA) • Root cause analysis • Specific, Measurable, Accountable, Realistic, Timely (SMART) • Value stream mapping (depict the flow) • JKK (No defects are passed to next process)
Learning culture	A learning culture is a collection of organizational conventions, values, practices and processes. These conventions encourage employees and organizations to develop knowledge and competence. An organization with a learning culture encourages continuous learning and believes that systems influence each other. Since constant learning elevates an individual as a worker and as a person, it opens opportunities for the establishment to transform continuously for the better.
Light weight ITSM	This variant of Information Technology (IT) Service Management (ITSM) is strictly focused on business continuity with a set of Minimum Required Information (MRIs). The MRI set for each organization depends on their business.
Logging levels	Within monitoring systems there are several levels of logging recognized: • Debug level: Information at this level is about anything that happens in the program, most often used during debugging. • Info level: Information at this level consists of actions that are user-driven or system specific. • Warn level: Information at this level tells us of conditions that could potentially become an error. Error level: Information at this level focuses on error conditions Fatal level: Information at this level tells us when we must terminate.
Loosely coupled architecture	Loosely coupled architectures enables that changes can be made safely and with more autonomy, increasing developer productivity.
Microservice	Microservices are a variant of the service-oriented architecture (SOA) architectural style that structures an application as a collection of loosely coupled services.

Begrip	Betekenis
	In a microservices architecture, services should be fine-grained, and the protocols should be lightweight [Wiki].
Micro service architecture	This architecture consists of a collection of services where each service provides a small amount of functionality, and the total functionality of the system is derived from composing multiple versions of a service in production simultaneously and to roll back to a prior version relatively easily.
Mini pipeline	In rare cases more than one deployment pipeline is required in order to produce the entire application. This can be accomplished by the use of a pipeline per application component. All these components are then assembled in a central pipeline which puts the entire application through acceptance tests, non-functional tests, and then deploys the entire application to testing, staging, and production environments.
Monitoring Framework	A framework of components that together form a monitor facility that is capable to monitor business logic, applications, and operating systems. Events, logs and measures are routed by the event router to destinations [Kim 2016].
Monolithic	A monolithic architecture is the traditional programming model, which means that elements of a software program are interwoven and interdependent. That model contrasts with more recent modular approaches such as a micro service architecture (MSA).
MTTR	Mean Time To Repair (MTTR) is a basic measure of the maintainability of repairable items. It represents the average time required to repair a failed component or device.
Muda	This is a Japanese word for waste. It is used in relationship to production systems.
Non-Functional Requirement (NFR)	NFR are requirements that define the quality of a product like maintainability, manageability, scalability, reliability, testability, deploy ability and security. NFR are also referred to as operational requirements.
Non-Functional Requirement (NFR) testing	NFR testing is the testing aspect that focusses on the quality of the product.
Obeya	Obeya is a war room which serves two purposes: • information management; • and on-the-spot decision making.
One piece flow	The Lean approach means that the DevOps team only works at one item at a time as a team with a fast pace and smooth flow. This is also used in the first way of the three ways of Gene Kim.
OPS liaison	An OPS liaison is an operation employee who is assigned to a development team in order to facilitate the development team for their infrastructural demands.
Operations	Operations is the team often responsible for maintaining the production environment and helping to ensure that required service levels are met [Kim 2016].
Operations stories	The work that has to be done by Ops can be written in stories. In that way that can be prioritized and managed.
Organization archetypes	There are three organization archetypes: functional, matrix, and market. They are defined by Dr. Roberto Fernandez as follows: • Functional: Functional-oriented organizations optimize for expertise, division of labour, or reducing cost.

Begrip	Betekenis
	• Matrix: Matrix-oriented organizations attempt to combine functional and market orientation. • Market: Market-oriented organizations optimize for responding quickly to customer needs.
Organizational typology model	This a model of Dr. Ron Westrum in which he defined three types of culture: 'pathological', 'bureaucratic', 'generative'. These organization types can be recognized by the following characteristics: • Pathological organizations are characterized by large amounts of fear and threat. • Bureaucratic organizations are characterized by rules and processes. • Generative organizations are characterized by actively seeking and sharing information to better enable the organization to achieve its mission. Dr. Westrum observed that in healthcare organizations, the presence of "generative" cultures was one of the top predictors of patient safety.
Over-the-shoulder	This is a review technique where the author walks through his code while another developer gives feedback.
Packages	A set of individual files or resources which are packed together as a software collection that provides certain functionality as part of a larger system.
Pair programming	This is review technique where two developers work together using one computer. While one developer writes the code the other reviews it. After one hour they exchange their role.
Peer review	This is a review technique where developers review each other's code.
Post mortems	After a major incident a post-mortem meeting can be organized in order to find out what the root-cause is of the incident and how to prevent it in the future.
Product owner	The Product Owner is a DevOps role. The Product Owner is the internal voice of the business. The Product Owner is the owner of the product backlog and determines the priority of the product backlog items in order to define the next set of functionalities in the service.
Programming paradigm	A style of building the structure and elements of computer programs.
Pull request process	This is a form of peer review that span Dev and Ops. It is the mechanism that lets engineers tell others about changes they have pushed to a repository.
QA	Quality Assurance (QA) is the team responsible for ensuring that feedback loops exist to ensure the service functions as desired [Kim 2016].
Reduce batch size	The size of a batch has an influence on the flow. Small batch sizes results in a smooth and fast flow. Large batch sizes results in high Work In Progress (WIP) and increases the level of variability in flow.
Reduce number of handoffs	In terms of a software process a handoff means that the work that is performed in order to produce software is stopped and handed over to another team. Each time the work passes from one team to another team, this requires all sorts of communication using different tools and filling up queues of work. To less handoffs the better.

Begrip	Betekenis
Release managers	This a DevOps role. The release manager is responsible for managing and coordinating the production deployment and release processes.
Release patterns	There are two patterns of releases to be recognized [Kim 2016]: • Environment-based release patterns: In this pattern there are two or more environments that receive deployments, but only one environment is receiving live customer traffic. • Application-based release patterns: In this pattern the application is modified in order to make selectively releases possible and to expose specific application functionality by small configuration changes.
Sad path	A specific type of a 'bad path' is called a 'sad path'. This is the case if the 'bad path' results in a security-related error condition.
Safety checks	Safety checks are performed during a release of a product. They are typical part of an HRR of an LRR.
SBAR	This technique offers guidelines for making sure concerns or critiques are expressed in a productive manner. In this situation the people who concerns it have to follow the following steps: • situational information to describe what is happening; • background information or context; • an assessment of what they believe the problem is; • recommendations for how to proceed.
Security testing	Security testing is one of many types of tests. Within DevOps security testing is integrated in the deployment pipeline by using automated tests as early as possible in the flow.
Self service capability	One way of integrating Ops in Dev is the usage of infrastructure self-services.
Shared goals	Delivering value to the customer requires that Dev and Ops are working together in value streams and have shared goals and practices.
Shared Operations Team (SOT)	A SOT is a team that is responsible for managing all the DTAP environments performing daily deployments into those development and test environments, as well as doing periodically production deployments. The reason to use a SOT is to have a team that focusses only on deployments. This results in automation of repeatable work and learning how to fix occurring problems very fast.
Shared version control repository	In order to be able to use trunk-based development DevOps engineers need to share their source code. The source code must be committed into a single repository that also supports version control. Such a repository is called a shared version control repository.
Simian army	Simian Army consists of services (Monkeys) for generating various kinds of failures, detecting abnormal conditions, and testing the ability to survive them. The goal is to keep the cloud service safe, secure, and highly available. Currently there are 3 Monkeys in the Simian Army: • Janitor Monkey (unused resources); • Chaos Monkey (try to shut down a service); Conformity Monkey (non-conformance to rules).
Single repository	A single repository is used to facilitate trunk-based development.
Smoke testing	Smoke testing is one of the test types that is used to determine whether or not the basics of a new or adjusted service works. Only a few testcases are needed to indicate whether or not at least the most important functions are working properly.

Begrip	Betekenis
	This test type origins from the hardware manufacturers where engineers tested circuits by powering on the system and checking for smoke which was an alarm of malfunctioning hardware.
Standard deviation	In statistics, the standard deviation (SD, also represented by the Greek letter sigma σ or the Latin letter s) is a measure that is used to quantify the amount of variation or dispersion of a set of data values. A low standard deviation indicates that the data points tend to be close to the mean (also called the expected value) of the set, while a high standard deviation indicates that the data points are spread out over a wider range of values [Wiki].
Standard operations	The standard operations is the situation in which the system performs as designed. Deviations of the standard operations need to be detected as early as possible.
Static analysis	Static analysis is a type of testing that is performed in a non-runtime environment, ideally in the deployment pipeline. Typically, a static analysis tool will inspect program code for all possible run-time behaviours and seek out coding flaws, back doors, and potentially malicious code [Kim 2016].
System of Engagement (SoE)	SoE's are decentralized Information Communication Technology (ICT) components that incorporate communication technologies such as social media to encourage and enable peer interaction [What-is].
System of Information (SoI)	The term SOI includes are all the tools that are used to process and visualize information from SoR systems. Typically, examples are Business Intelligence (BI) systems.
System of Records (SoR)	A SoR is an ISRS (information storage and retrieval system), that is the authoritative source for a particular data element in a system containing multiple sources of the same element. To ensure data integrity, there must be one -- and only one -- system of record for a given piece of information [What-is].
Swarming	David Bernstein explains how swarming helps to build an effective team which is able to focus and solve complex problems: "When swarming, the whole team works together on the same problem. It helps to know each other and work well together. Generally, groups need to go through the phases of forming (getting to know each other) and storming (having conflicts and resolving them) before they get to performing (being a highly functional team), so give everyone the space to become a team.". According to Dr. Spear, the goal of swarming is to contain problems before they have a chance to spread, and to diagnose and treat the problem so that it cannot recur. "In doing so," he says, "they build ever-deeper knowledge about how to manage the systems for doing our work, converting inevitable up-front ignorance into knowledge." [Kim 2016].
Technology adaption curve	It takes time for new technology to get adapted in the market. The technology adaption curve indicates the stages of market penetration in time.
Technology executives	This is a DevOps role also named 'value stream manager'. The value stream manager is someone who is responsible for "ensuring that the value stream meets or exceeds the customer (and organizational) requirements for the overall value stream, from start to finish" [Kim 2016].

Begrip	Betekenis
Test driven development	Test driven development is the approach in which the source code is written after the completion of the test case definition and execution. The source code is written and adjusted until the test case conditions are met.
Test harness	Software constructed to facilitate integration testing. Where test stubs are typically components of the application under development and are replaced by working components as the application is developed (top-down integration testing), test harnesses are external to the application being tested and simulate services or functionality not available in a test environment.
The ideal testing automation pyramid	The ideal testing automation pyramid is a way of testing that can be characterized as follows: • Most of the errors are found using unit tests as early as possible. • Run faster-running automated tests (e.g., unit tests) before slower-running automated tests (e.g., acceptance and integration tests), which are both run before any manual testing. • Any errors should be found with the fastest possible category of testing.
The non-ideal testing automation inverted pyramid	The non-ideal testing automation pyramid is a way of testing that can be characterized as follows: • Most of the investment is in manual and integration testing. • Errors are found later in the testing. • Slower running automated tests are performed first.
Test driven development	Test driven development is the approach in which the source code is written after the completion of the testcase definition and execution. The source code is written and adjusted until the testcase conditions are met.
Test harness	Software constructed to facilitate integration testing. Where test stubs are typically components of the application under development and are replaced by working components as the application is developed (top-down integration testing), test harnesses are external to the application being tested and simulate services or functionality not available in a test environment.
The Agile Manifesto	The Agile Manifesto (Manifesto for Agile Software Development) was set up during an informal meeting of seventeen software DevOps engineers. This meeting took place from 11 to 13 February 2001 at "The Lodge" in Snowbird, Utah. The charter and the principles formed an elaboration of ideas that had arisen in the mid-nineties, in response to methods traditionally classed as waterfall development models. Those models were experienced as bureaucratic, slow, and narrow-minded and would hinder the creativity and effectiveness of DevOps engineers. The seventeen people who have drawn up the Agile Manifesto together represented the various Agile movements. After the publication of the charter, several signatories set up the "Agile Alliance" to further convert the principles into methods [Wiki].
The Lean movement	An operating philosophy that stresses listening to the customer, tight collaboration between management and production staff, eliminating waste and boosting production flow. Lean is often heralded as manufacturers' best hope for cutting costs and regaining their innovative edge.
The Simian Army	The Simian Army is a collection of open-source cloud testing tools created by the online video streaming company, Netflix.

Begrip	Betekenis
	The tools allow engineers to test the reliability, security, resiliency and recoverability of the cloud services that Netflix runs on Amazon Web Services (AWS) infrastructure [Whatis]. Within this Simian Army the following monkeys are recognized: Chaos Gorilla, Chaos Kong, Conformity Monkey, Doctor Monkey, Janitor Monkey, Latency Monkey and Security Monkey.
Theory of constraints	This is a methodology for identifying the most important limiting factor that stands in the way of achieving a goal and then systematically improving that constraint until it is no longer the limiting factor.
The three ways	The three ways are introduced in 'The Phoenix Project: A Novel About IT, DevOps, And Helping Your Business Win' by Gene Kim, Kevin Behr and George Spafford. The Three Ways are an effective way to frame the processes, procedures and practices of DevOps, as well as the prescriptive steps. • The first way – flow understand and increase the flow of work (left to right); • The second way – feedback create short feedback loops that enable continuous improvement (right to left); • The third way – Continuous Experimentation and Learning (Continuous Learning).
Tool-assisted code review	This is a review technique where authors and reviewers use specialized tools designed for peer code review or facilities provided by the source code repositories [Kim 2016].
Toyota Kata	Toyota Kata is a management book by Mike Rother. The book explains the Improvement Kata and Coaching Kata, which are a means for making the Continual improvement process as observed at the Toyota Production System teachable [Wiki].
Transformation team	Introducing DevOps requires a defined transformation strategy. Based on their research, Dr. Govindarajan and Dr. Trimble assert that organizations need to create a dedicated transformation team that is able to operate outside of the rest of the organization that is responsible for daily operations (which they call respectively the "dedicated team" and "performance engine"). The lessons learned from this transformation team can be used to apply in the rest of the organization.
Value stream	The process required to convert a business hypothesis into a technology-enabled service that delivers value to the customer [Kim 2016].
Value Stream Mapping (VSM)	Value stream mapping is a Lean tool that depicts the flow of information, materials, and work across functional silos with an emphasis on quantifying waste, including time and quality.
Vertical splitting of features	A feature can be splitted into stories. Vertical splitting refers to the result of a feature splitting in which more DevOps teams can work independently on their own stories. Together they realize the feature. See also Horizontal splitting of features.
Virtualized environment	An environment that is based on virtualization of hardware platforms, storage devices and network resources. In order to create a virtualized environment usually VMware is used.
Visualization	In computing, virtualization refers to the act of creating a virtual (rather than actual) version of something, including virtual computer hardware platforms, storage devices, and computer network resources.

Begrip	Betekenis
	Virtualization began in the 1960s, as a method of logically dividing the system resources provided by mainframe computers between different applications. Since then, the meaning of the term has broadened [Wiki].
Walking skeleton	Walking skeleton means doing the smallest possible amount of work to get all the key elements in place.
Waste	Waste comprises the activities that are performed in the manufacturing process that are not adding value to the customer. Examples in the context of DevOps are: • Unnecessary software features. • Communication delays. • Slow application response times. • Overbearing bureaucratic processes.
Waste reduction	Minimization of waste at its source is to minimize the quantity required to be treated and disposed of, achieved usually through better product design and/or process management. Also called waste minimization [Businessdictionary].
Work In Progress (WIP	Material that has entered the production process but is not yet a finished product. Work in progress (WIP) therefore refers to all materials and partly finished products that are at various stages of the production process.
WIP limit	This is a Key Performance Indicator (KPI) that is used in the Kanban process to maximize the number of items that has been started but that is not completed. Limiting the amount of WIP is an excellent way to increase throughput in your software development pipeline.

Tabel B-1, Begrippenlijst.

Bijlage C, Afkortingen

Afkorting	Betekenis
%C/A	Percent Complete / Accurate
AI	Artificial Intelligence
ASL	Application Services Library
AVG	Algemene Verordening Gegevensbescherming
AWS	Amazon Web Services
BDD	Behavior Driven Development
BI	Business Intelligence
BISL	Business Information Services Library
CAB	Change Advisory Board
CAMS	Culture, Automation, Measurement and Sharing
CD	Continuous Delivery
CD	Continuous Deployment
CDO	Chief Data Officer
CDO	Chief Digital Officer
CE	Continuous Everything
CEO	Chief Executive Officer
CFO	Chief Financial Officer
CI	Continuous Integration
CI	Configuration Item
CIA	Confidentiality, Integrity & Accessibility (or Availability)
CIO	Chief Information Officer
CISO	Chief Information Security Officer
CMDB	Configuration Management DataBase
CMMI	Capability Maturity Model Integration
CMS	Configuration Management System
COO	Chief Operating Officer
CoP	Communities of Practice (CoP)
CPU	Central Processing Unit
CRO	Chief Risk Officer
CSI	Continual Service Improvement
CT	Component Test
CTO	Chief Technical Officer
DAP	Dossier Afspraken en Procedures
DAS	Direct Attached Storage
DevOps	Development & Operations
DL	Deep Learning
DML	Definitive Media Library
DNS	Domain Name System
DoD	Definition of Done
DoR	Definition of Ready
DTAP	Development, Test, Acceptance and Production
E2E	End-to-End

Afkorting	Betekenis
ERD	Entity Relation Diagram
ERP	Enterprise Resource Planning
ESA	Epic Solution Approach
ESB	Enterprise Service Buss
ETL	Extract Transform & Load
EUX	End User eXperience Monitoring
FAT	Functionele AcceptatieTest
FSA	Feature Solution Approach
GAT	Gebruiker AcceptatieTest
GCC	General Computer Controls
GDPR	General Data Protection Regulation
Git	Global Information Tracker
GSA	Generieke & Specifieke Acceptatiecriteria
GUI	Graphical User Interface
GWT	Given-When-Then
HDD	Hard Disk Drive
HRM	Human Resource Management
HRR	Hand-off Readiness Review
IAAS	Infrastructure as a Service
IaC	Infrastructure as Code
ICT	Information Communication Technology
ID	Identifier
INVEST	Independent, Negotiable, Valuable, Estimatable, Small and Testable
IoT	Internet of Things
ISMS	Information Security Management System
ISVS	Information Security Value System
IT	Information Technology
ITIL	Information Technology Infrastructure Library
ITSM	Information Technology Service Management
JIT	Just In Time
JKK	Ji-Kotei-Kanketsu
JVM	Java Virtual Machine
KPI	Key Performance Indicator
KSF	Kritieke Succes Factor
LCM	LifeCycle Management
LDAP	Lightweight Directory Access Protocol
LRR	Launch Readiness Review
LT	Lead Time
ML	Machine Learning
MOF	Microsoft Operations Framework
MRI	Minimum Required Information
MT	Module Test
MTBF	Mean Time Between Failure

Afkorting	Betekenis
MTBSI	Mean Time Between System Incidents
MTTR	Mean Time To Repair
MVP	Minimal Viable Product
NAS	Network Attached Storage
NFR	Non-Functional Requirement
OLA	Operational Level Agreement
OTAP	Ontwikkel-, Test-, Acceptatie- en Productieomgeving
PaaS	Platform as a Service
PAT	Productie AcceptatieTest
PBI	Productie Backlog Item
PDCA	Plan Do Check Act
POR	Project or Organisational Risk
PPT	People, Process & Technology
PSQL	ProgreSQL
PST	Performance StressTest
PT	Processing Time
QA	Quality Assurance
QC	Quality Control
RACI	Responsibility, Accountable, Consulted and Informed
RASCI	Responsibility, Accountable, Supporting, Consulted and Informed
RBAC	Role-based access control
REST API	REpresentational State Transfer Application Programming Interface
RFC	Request for Change
ROI	Return On Investment
RUM	Real User Monitoring
S-CI	Software Configuration Item
SA	Strategic IS Architecture
SaaS	Software as a Service
SAFe	Scaled Agile Framework
SAN	Storage Area Network
SAT	Security AcceptatieTest
SBAR	• Situational information to describe what is happening; • Background information or context; • An assessment of what they believe the problem is; • Recommendations for how to proceed.
SBB	System Building Block
SBB-A	System Building Block Application
SBB-I	System Building Block Information
SBB-T	System Building Block Technology
SIT	Systeemintegratietest
SLA	Service Level Agreement
SM	Strategic Match
SMART	Specific, Measurable, Accountable, Realistic, Timely

Afkorting	Betekenis
SME	Subject Matter Expert
SNMP	Simple Network Management Protocol
SoE	System of Engagement
SoI	Systems of Information
SoR	System of Records
SoX	Sarbanes Oxley
SQL	Structured Query Language
SRG	Standards Rules & Guidelines
SSL	Secure Sockets Layer
ST	Systeemtest
SVS	Service Value System
TCO	Total Cost of Ownership
TCP	Transmission Control Protocol
TDD	Test Driven Development
TFS	Team Foundation Server
TPS	Toyota Production System
TSQL	Transact Structured Query Language
TTM	Time To Market
TU	Technical Uncertainty
TVB	Taken, Verantwoordelijkheden en Bevoegdheden
UAT	User Acceptance Test
UML	Unified Modeling Language
UT	Unit Testing
UX design	User eXperience Design
VOIP	Voice over Internet Protocol
VSM	Value Stream Mapping
WAN	Wide Area Network
WIP	Work In Progress
WMI	Windows Management Instrumentation
WoW	Way of Working
XP	eXtreme Programming

Tabel C-1, Afkortingen.

Bijlage D, Agile tools

Op basis van een onderzoek dat bij tien organisaties is uitgevoerd [Best 2015a] is een lijst samengesteld van de ingezette tools en de inzet van deze tools. De inzet van tools blijken zeer veelvuldig te worden aangepast. Figuur D-1 is daarom een momentopname. Het model kan wel goed gebruikt worden om de eigen DevOps tools op compleetheid te beoordelen. In Tabel D-1 is tevens per tool een URL opgenomen om meer informatie over de tools te geven.

	Agile Project Management								Test Mngt.				Service Management								
Tool	Feature / User story admin	Sprint backlog	Dashboard	Software configuration management	Code repository	Code review	Version control / Baselining	Continuous Integration	Issue tracking	Test cases / Test scripts	Test automation	Regression testing	Security management	Incident management	Problem management	Change Management	Configuration management	Release & deployment management	Continuous Delivery	DTAP	Reporting	Totals
#define	1			1		1			1	1			1							1		7
Bamboo								1														1
CCCQ				1	1	1	1							1	1		1					7
Cherwell													1	1	1							3
ClearCase				2	1	1	1										1					6
ClearQuest							1		1					1	1		1					5
Clientele									1				2	1		1	1	1				7
Enterprise Architect	2	1																1				4
FitNesse										1												1
GIT				1	2		2		1													6
HP ALM	1								2	3	2	3			1		1					13
HP SC				1									1	1	1	1		1				6
HP Service Manager													1	1	1	1						4
IBM Tivoli Service Management													1	1								2
Ice-Scrum	1																					1
Installshield																	1					1
Jenkins								5												1	1	7
Jira	3	4	3				1		2						1							14
JMeter											2	4										6
Leankit	2	1	1										1		1							6
Maven				1				2			1									1	1	6
Mavim								1														1
McAfee																		1				1
Mediawiki															1							1
MS Excel										1							1	1				3
MSI				1	1	1	1	1							1	1	1			1	1	10
MTM										1	1	1										3
Nexus				1				1														2
NPS			1																			1
Octopus Deploy								1														1
Omnitracker													1	1	1		1					4
OTRS									1				1									2
Other																		1				1
Powershell																				1	1	2
ReportServer			1															1				2
Robot Framework												1										1
ROSS																	1			1	1	3
ScrumWise	1	1	1																			3
Selenium										1	4	4										9
Serena Dimensions					1																	1
SOAtest										2	1	2										5
Sonar						2		2														4
Subversion					2	2	3															7
TestComplete										1	1	1										3
TFS	2	2	2	1	2	2	3	1	1	1	1	1			1		1	1		1	1	24
TOPdesk													1	1	2	1	2	1				8
Twist											1	1										2
Totals	13	9	9	10	10	10	13	15	10	11	15	18	11	9	14	5	13	7	2	7	6	

Figuur D-1, Gebruik van tools door tien organisaties.

Short name	Long name	Website
#define	issue-tracker	(proprietary)
Bamboo	Bamboo	www.atlassian.com
CCCQ	IBM® Rational® ClearCase® ClearQuest	www.ibm.com
Cherwell	Cherwell Software	www.cherwell.com

Short name	Long name	Website
ClearCase	IBM® Rational® ClearCase	www.ibm.com
Clientele	Mproof Clientele IT Service Management	www.mproof.com
Enterprise Architect	Sparx Systems Enterprise Architect	www.sparxsystems.com
FitNesse	FitNesse	www.fitness.org
Git	Global Information Tracker	www.git-scm.com
HP ALM	HP Application Lifecycle Management	www8.hp.com
HP QC	HP Quality Center	www8.hp.com
HP SC	HP ServiceCenter Software	H10076.www1.hp.com
HP Service Manager	HP Service Manager	www8.hp.com
IBM Tivoli Service Management Suite	IBM Tivoli Service Management Suite	www-01.ibm.com
Ice-Scrum	Ice-Scrum	www.icescrum.org
Installshield	Installshield	www.installshield.com
IT Service Management	IT Service Management (Mexon)	www.mexontechnology.com
Jenkins	Jenkins	www.jenkins-ci.org
Jira	Jira	www.atlassian.com
JMeter	Apache JMeter™	jmeter.apache.org
LeanKit	LeanKit™	www.Leankit.com
Maven	Apache Maven Project	maven.apache.corg
Mavim	Mavim	www.mavim.com
McAfee	McAfee®	www.mcafee.com
MRM	Microsoft Release Manager	www.ibm.com/software
MS Excel	MicroSoft Excel	www.microsoft.com
MSI	MicroSoft Installer	msdn.microsoft.com
MTM	Microsoft Test Manager	msdn.microsoft.com
Nexus	Nexus	www.ichrome.com/solutions/nexus
NPS	Net Promoter Score	www.netpromoter.com
Octopus Deploy	Octopus Deploy	www.octopusdeploy.com
Omnitracker	Omnitracker	www.omnitracker.com
OTRS	Open-source Ticket Request System	www.otrs.org
Powershell	Automated Powershell script	microsoft.com/powershell
ReportServer	SQL Server Reporting Services (SSRS)	msdn.microsoft.com
Robot Framework	Robot Framework	www.robotframework.org
ROSS	RepliWeb Operations Suite for SharePoint	www.repliweb.com
ScrumWise	ScrumWise	www.scrumwise.com
Selenium	Selenium	docs.seleniumhq.org
Serena Dimensions	Serena Dimensions	www.serena.com
SOAtest	Parasoft SOAtest	www.parasoft.com
SonarQube	SonarQube™	www.sonarSource.org
Subversion	Apache SubVersioN (SNV)	subversion.apache.org
TestComplete	TestComplete	SmartBear Software
TFS	Team Foundation Server	msdn.microsoft.com

Short name	Long name	Website
TOPdesk	TOPdesk	www.topdesk.nl
Twist	Twist	www.thoughtworks.com
WinMerge	winMerge	www.winmerge.com

Tabel D-1, Tools.

Bijlage E, Websites

Name	Reference	URL
Betekenis	[Betekenis]	https://www.betekenis-definitie.nl
bigpanda	[Bigpanda]	https://www.bigpanda.io/blog/event-correlation/
Bullseye	[Bullseye]	https://www.bullseye.com/minimum.html
Businessdictionary	[Businessdictionary]	http://www.businessdictionary.com
CEO	[CEO]	https://www.cio.com
Collabnet	[CollabNet]	https://www.collab.net
CleanArchitecture	[CleanArchitecture]	https://www.freecodecamp.org/news/a-quick-introduction-to-clean-architecture-990c014448d2/
CleanCode	[CleanCode]	https://cvuorinen.net/2014/04/what-is-clean-code-and-why-should-you-care/
Consultancy	[Consultancy]	https://www.consultancy.nl/nieuws/30410/de-vijf-belangrijkste-it-trends-van-2020-en-2021
dbmetrics	[dbmetrics]	http://www.dbmetrics.nl
dbmetrics	[dbmetrics publicaties]	https://www.dbmetrics.nl/wp-content/uploads/2021/07/dbmetrics_best-practice-publications_2021-07-22_900.pdf
DevOps	[DevOps]	http://DevOps.com
DDD	[DDD]	https://www.slideshare.net/skillsmatter/ddd-in-agile
doxygen	[doxygen]	http://www.doxygen.nl/manual/docblocks.html
doxygen voorbeeld	[doxygen voorbeeld]	http://www.doxygen.nl/manual/examples/qtstyle/html/class_q_tstyle___test.html#a0525f798cda415a94fedeceb806d2c49
EXIN	[Exin]	http://www.exin.nl
Gartner	[Gartner]	https://www.gartner.com/en/information-technology/glossary/cio-chief-information-officer
Gladwell	[GLADWELL]	http://www.gladwill.nl
IIR	[IIR]	http://www.IIR.nl
Investopedia	[Investopedia]	https://www.investopedia.com
ITMG	[ITMG]	http://www.ITMG.nl
ITPedia	[ITPEDIA]	http://www.itpedia.nl
Patrick Cousot	[Patrick Cousot]	https://www.di.ens.fr/~cousot/abstract_interpret.shtml
Passioned	[Passioned]	https://www.passionned.nl
Porter	[Porter]	https://medium.com/@sniloy/value-chain-analysis-value-stream-mapping-and-business-process-mapping-what-is-the-difference-431589d27ea8
Tiobe	[Tiobe]	www.tiobe.com/content/paperinfo/DefinitionOfConfidenceFactor.html
UnitTest	[UnitTest]	https://docs.python.org/3/library/unittest.html
Wiki	[Wiki]	http://nl.wikipedia.org / wiki / Cloud_computing

Name	Reference	URL
Wiki docgen	[Wiki docgen]	https://en.wikipedia.org/wiki/Comparison_of_documentation_generators

Tabel E-1, Websites.

Bijlage F, Index

%

%C/A · 83

A

A/B testing · 67
acceptatiecriterium · 68, 72
acceptatieperiode · 48
acceptatietest · 67
access management · 16, 49
account · 9, 12
administratiepakket · 10, 11
affinity · 67
Agile · 4, 21, 24, 25, 31, 33, 35, 37, 39, 55, 56, 67, 71, 80, 85
  - aanpak · 33
  - infrastructure · 67
  - Manifesto · 24, 80
  - Scrum · 33, 39, 56, 71
  - werken · 21, 55
AI · 11, 37, 55, 56, 57, 58, 59, 83
AI tool · 57
algemeen directeur · 7
Algemene Verordening Gegevensbescherming · Zie AVG
Algol · 22, 23
algoritme · 57, 58
alternate path · 67
Amazon Web Services · Zie AWS
AmazonRDS · 12
analyse · 49, 50
Andon cord · 67
Android · 12
anomaly detection technique · 67
antifragility · 68
anti-pattern · 68
app · 23, 34
Apple · 8, 12, 23, 34, 68
Apple macOS · 12
applicatiebeheer · 37, 74
applicatiecomponent · 76
applicatieontwikkeling · 51
Application Services Library · Zie ASL
apps · 35
architectuur · 46
architectuurmodel · 25
architectuurprincipe · 25
artefact · 68, 70
artefact repository · 68
Artificial Intelligence · Zie AI
ASL · 38, 39, 45, 83
assessment · 78, 85
Atanasoff · 8
authenticatie · 24, 58
authenticatie en autorisatie · 30
authenticiteit · 47
automated test · 68
autorisatie · 24, 58
availability · 69
availability management · 43, 47
AVG · 83
AWS · 6, 7, 9, 16, 19, 81, 83
Azure · 16

B

back-end · 25, 33
backlog item · 77
back-up · 24
bad apple theory · 68
bad path · 68
BASIC · 22, 23, 72
batchverwerking · 43
BDD · 21, 32, 51, 68, 83
bedrijfsproces · 10, 12, 16, 42, 43, 46, 55
bedrijfsregel · 27
beeldscherm · 5, 13, 31
beeldscherm formaat · 30
Behavior Driven Development · Zie BDD
beheer · 31, 37, 39
  - organisatie · 43
  - proces · 10, 41, 43, 44, 46, 47, 49, 50
  - taak · 37, 46
  - werkzaamheden · 37, 38
behoefte · 8, 10, 31, 37, 45, 47
beschikbaarheid · 43, 47, 55
beschikbaarheidsnorm · 43
best practice · 37, 38, 69
besturingslogica · 5
besturingssysteem · 11, 12, 16
beveiliging · 18, 47
beveiligingsbeleid · 47
BI · 10, 11, 55, 56, 79, 83
binary · 68
BIOS · 9, 14
BiSL · 38, 39, 45
BISL · 83
BI-tool · 10, 11
blameless post mortem · 68
blamelessness · 68
blockchain · 58
blue/green deployment · 68
boundary · 58
branching · 69
broken build · 68
broker · 17, 18
brown field · 68
budgetting · 43
build · 68, 69, 70, 79
business
  - continuity management · 43
  - logica · 5, 8, 10, 12, 24
  - relationship management · 44, 45
  - servicecatalogus · 46
  - value · 69, 71

Business DevOps · 40
Business Information Services Library · Zie BISL
Business Intelligence · Zie BI
Business Process as a Service · 16
business rule · 30

C

C# · 10, 23
C++ · 72
CA · 38
CAB · 83
CAMS · 70, 83
canary releasing · 69
capability · 47, 70
Capability Maturity Model Integration · Zie CMMI
capaciteit · 43
capacity · 69
capacity management · 43, 46, 47
card reader · 14
CD · 14, 37, 38, 51, 69, 72, 83
CDO · 6, 83
CE · 38, 83
Central Processing Unit · Zie CPU
Centrale Bank · 7
CEO · 6, 7, 83
CFO · 6, 83
change · 42, 48
Change Advisory Board · Zie CAB
change category · 69
change management · 42, 48
change schedule · 69
charging · 43
Chief Data Officer · Zie CDO, Zie CDO
Chief Executive Officer · Zie CEO
Chief Financial Office · Zie CFO
Chief Information Officer · Zie CIO
Chief Information Security Officer · Zie CISO
Chief Operating Officer · Zie COO
Chief Risk Officer · Zie CRO
Chief Technology Officer · Zie CTO
Chrome book · 8
CI · 21, 37, 41, 42, 46, 47, 48, 51, 69, 72, 83
CIA · 38, 83
CIO · 6, 7, 83
CISO · 6, 83
CL · 38
cloud · 69
  - computer · 19
  - computing · 16
  - configuration file · 69
  - oplossing · 1
  - provider · 6, 20, 37
  - service · 6, 17, 69, 78
  - software · 21
cluster immune system release pattern · 69
CM · 38
CMDB · 41, 83
CMMI · 38, 83
CMS · 83
CO · 21
COBOL · 8, 10, 22, 23, 72
code branch · 69
code review form · 69
codified NFR · 69
collaboration · 69
commit code · 69
commit stage · 69
communicatiemiddel · 5
communicatieprotocol · 15
Communities of Practice · Zie CoP
competentie · 31, 56
Completeness / Accurateness · Zie %C/A
compliance · 70
compliance checking · 70
compliancy · 70
compliancy officer · 70
component · 13, 14, 15, 19, 26, 29, 30, 31, 32, 46, 50, 72, 76, 79, 80
Component Test · Zie CT
computer · 12, 15
  - capaciteit · 6
  - chip · 8
  - processor · 59
  - ruimte · 53
concurrentievoordeel · 10
Confidentiality, Integrity & Accessibility · Zie CIA
configuration file · 10
Configuration Item · Zie CI
configuration management · 70
Configuration Management DataBase · Zie CMDB
Configuration Management System · Zie CMS
connectiviteit · 9, 53
container · 29, 70
container omgeving · 29
contingency maatregel · 24
Continual Service Improvement · Zie CSI
continuity · 69, 75
Continuous
  - Delivery · Zie CD
  - Deployment · Zie CD
  - Everything · Zie CE
  - Integration · Zie CI
  - Learning · 81
control · 39, 46, 53, 70, 78
Conway's law · 70
COO · 6, 7, 83
CoP · 21, 83
counter measure · 74
CP · 38
CPU · 13, 14, 83
CRO · 6, 7, 83
CSI · 49, 83
CT · 38, 83
CTO · 6, 7, 83
cultural debt · 70
Culture, Automation, Measurement and Sharing · Zie CAMS

Cybersecurity · 55
cycle time · 70

D

DAP · 48, 83
DAS · 12, 13, 83
data
  - center · 5
  - gedreven · 6
  - mining tool · 11
  - opslagplaats · 27
  - security · 55
database · 5, 9, 11, 12, 16, 23, 24, 30, 31, 33, 41, 55, 56, 70
database managementsysteem · 16
debt · 70
declarative programming · 70
deep blue · 56
Deep Learning · Zie DL
defect · 75
defect tracking · 71
Definition of Done · Zie DoD
Definition of Ready · Zie DoR
Definitive Media Library · Zie DML
deliverable · 30, 31, 70
demand / supply · 40, 53
demand management · 44
demand zijde · 53
Demming wheel · 74
deployment · 67
deployment pipeline · 69
design · 70, 82
design coordination · 46
desktop image scanner · 14
development · 67, 68, 71, 73, 75, 76, 78, 80, 82
Development & Operations · Zie DevOps
development ritual · 71
development tool · 30, 31
Development, Test, Acceptance and Production · Zie DTAP
device · 12, 13, 14, 34, 76
DevOps · 32, 37, 38, 39, 51, 52, 55, 56, 63, 67, 68, 69, 70, 71, 72, 76, 77, 78, 79, 80, 81, 82, 83
  - engineer · 67, 70, 78, 80
  - team · 67, 68, 69, 70, 72, 76, 81
DevOps Lemniscaat · 51, 52
DFA · 43, 45
digitalisering · 55, 56, 58
Direct Attached Storage · Zie DAS
directeur · 6, 7
discriminator · 34
DL · 11, 55, 56, 57, 58, 83
DL-algoritm · 57
DL-methode · 57
DML · 38, 83
DNS · 83
DoD · 21, 74, 83
doelstelling · 1
Domain Name System · Zie DNS
doorbelasten · 43
doorlooptijd · 42
DoR · 21, 83
Dossier Afspraken en Procedures · Zie DAP
downward spiral · 71
DRAM · 8
DTAP · 38, 78, 83
DTAP environments · 78

E

E2E · 21, 83
e-commerce bedrijf · 25
electriciteit · 53
e-mail pass around · 71
emerging architectuur · 25
End User eXperience Monitoring · Zie EUX
End-to-End · Zie E2E
energiecentrale · 20
energieconsumptie · 20
engine · 29, 30
Eniac · 8
Enterprise Resource Planning · Zie ERP
Enterprise Service Bus · 27, Zie ESB
Entity Relation Diagram · Zie ERD
epic · 84
Epic Solution Approach · Zie ESA
EPROM · 14
ERD · 84
ERP · 11, 58, 84
error path · 71
ESA · 84
ESB · 21, 27, 28, 84
E-shaped · 73
ethernet · 8
ETL · 30, 31, 84
ETL techniek · 31
EUX · 84
evaluation · 48
event · 49, 76
event management · 49
exclusiviteit · 18
extern geheugen · 9
external hard drive · 14
externe speaker · 13
Extract Transform & Load · Zie ETL
eXtreme Programming · Zie XP

F

failure · 68
FAT · 31, 32, 38, 67, 84
feature · 71, 72, 81, 82
Feature Driven Development · 33
Feature Solution Approach · Zie FSA
feature toggle · 71
feed forward · 71
feedback · 23, 32, 51, 52, 58, 70, 71, 74, 77, 81
feedforward · 71
file system · 12

financial management for IT services · 44
firewall · 15
firmware · 5, 9, 14
flash geheugen · 14
floppy disk · 8
flow · 70, 73, 75, 76, 77, 78, 80, 81
Fortran · 8, 22
fossielbrandstof · 20
foutboodschap · 29
framework · 21, 31, 33, 76
front-end · 25
FSA · 84
functie · 103
functionaliteit · 11, 24, 26, 32, 33, 34, 37, 45, 47, 48, 49, 52
Functionele AcceptatieTest · Zie FAT
fysieke computer · 15

G

GAT · 31, 32, 38, 84
Gaussian distribution · 67, 71
GCC · 84
GDPR · 55, 84
gebruiker · 45
Gebruiker AcceptatieTest · Zie GAT
gebruikersinterface · 27
gebruikersorganisatie · 49
gebruikersperspectief · 30
gebruiksgemak · 32
gedragsrequirement · 32
gegevensdefinitie · 30
geheugen · 16
General Computer Controls · Zie GCC
General Data Protection Regulation · Zie GDPR
Generieke & Specifieke Acceptatiecriteria · Zie GSA
gestructureerde data · 55
Gherkin language · 30, 32
Git · 21, 84
GIT · 30, 32
Given When Then · 72, Zie GWT
Global Information Tracker · Zie Git
Go · 23
Gordon Moore · 8
governance · 21, 33, 37, 38
Graphical User Interface · Zie GUI
green field · 72
GSA · 38, 84
GUI · 7, 8, 24, 27, 30, 84
GWT · 21, 30, 32, 72, 84

H

hacking · 59
Hand-off Readiness Review · Zie HRR
happy path · 67, 68, 72
Hard Disk Drive · Zie HDD
hardware · 5, 8, 9, 12, 13, 14, 15, 16, 19, 41, 42, 53, 70, 72, 73, 79, 81
HDD · 14, 84
helptekst · 29
holocrazy · 72
horizontal splitting of feature · 72, 81
hosting · 6, 53
housing · 6
HRM · 11, 84
HRR · 38, 72, 78, 84
HTML · 8
Human Resource Management · Zie HRM
HW · 13

I

IAAS · 6, 7, 16, 53, 84
IaC · 22, 67, 72, 73, 84
ICT · 73, 84
  - component · 48
  - infrastructuur · 41, 48
  - middel · 43
  - organisatie · 41, 43
  - service · 47, 49
  - serviceverlening · 41, 43
ID · 22, 84
ideal test pyramid · 80
idempodent · 72
IDentifier · Zie ID
IDentity Management · *Zie* IDM
IDM · 49
impact · 46
impactanalyse · 24, 25
imparative programming · 72
implementeren · 44
incident management · 42, 49
incrementeel · 24, 32, 33
Independent, Negotiable, Valuable, Estimatable, Small and Testable · Zie INVEST
informatie · 50
  - beheer · 37
  - behoefte · 37
  - beveiliging · 47
  - beveiligingsbeleid · 6
  - gestuurd · 6
  - systeem · 37, 40, 52
  - verwerkingsketens · 25
  - voorziening · 37, 39, 53
Information Communication Technology · Zie ICT
information radiator · 73
Information Security Management System · Zie ISMS
Information Security Value System · Zie ISVS
Information Technology · Zie IT
Information Technology Infrastructure Library · Zie ITIL
Information Technology Service Management · Zie ITSM
Infosec · 73
Infrastructure as a Service · 16, Zie IAAS
Infrastructure as Code · Zie IaC

infrastructure component · 72, 73
infrastructure management · 72, 73
infrastructuurbeheer · 37, 53
infrastructuurobject · 5
instructie · 9, 13, 14, 23
integratie · 32, 37, 38
integratietest · 32
integratietesten · 31
integriteit · 47, 55
integriteiteis · 30
interne cloud service · 16
interne opslag · 12
interne speaker · 13
internet · 5, 6, 8, 14, 16, 20, 31, 53, 55, 59
Internet of Things · Zie IoT
INVEST · 22, 73, 84
inwijkplan · 43
IoT · 55, 58, 59, 84
IP address · 72, 73
I-shaped · 73
ISMS · 38, 84
IST · 74
ISVS · 38, 84
IT · 1, 71, 75, 81, 84
ITIL · 37, 38, 39, 40, 41, 42, 43, 44, 45, 47, 48, 49, 50, 51, 84
ITSM · 75, 84

J

Java · 8, 10, 23, 84
Java Virtual Machine · Zie JVM
Ji-Kotei-Kanketsu · Zie, Zie JKK
JIT · 73, 74, 84
JKK · 73, 84
Just In Time · Zie JIT
JVM · 22, 84

K

Kaizen · 74, 75
Kaizen Blitz (or Improvement Blitz) · 74
Kaizen in advance · 74
Kanban · 33, 74, 75, 82
kennis · 46
kennisontwikkeling · 20
kennisprofiel · 46
kerncompetentie · 6
kerntaak · 7
Key Performance Indicator · Zie KPI
keyboard · 13
kibana dashboard · 74
klant · 10, 11, 16, 17, 23, 30, 42, 43, 45, 46, 48, 49, 53, 55
klantperspectief · 45
knowledge management · 48
known error · 47
Koppeling · 30
kostenallocatie · 43
kosteneffectief · 45
kosteneffectiviteit · 49
kostprijs · 43
KPI · 38, 50, 74, 75, 82, 84
Kritieke Succes Factor · Zie KSF
KSF · 84
Kubernetes · 29
kwaliteit · 37, 42, 45, 47, 48, 49, 52, 54, 59
kwaliteit management · 49
kwaliteitsverhoging · 32
kwantum computer · 8

L

LAN · 14
Laravel · 31, 33
latent defect · 75
lauching guidance · 75
Launch Readiness Review · Zie LRR
LCM · 84
LDAP · 84
Lead Time · Zie LT
Lean · 75, 80, 81
  - production aanpak · 52
  - software ontwikkelaanpak · 52
  - tool · 75
learning culture · 75
leespen · 14
leverancier · 10, 11, 16, 18, 34, 53
levering · 45, 49
leveringsvoorwaarden · 46
lifecycle · 71, 72
LifeCycle Management · Zie LCM
Lightweight Directory Access Protocol · Zie LDAP
limitation · 58
Linux · 12
log · 76
logging en monitoring · 30
logging level · 75
logisch datamodel · 30
loosely coupled architecture · 75
loosely coupled services · 75
LRR · 75, 84
LT · 75, 84

M

maatwerkprogrammatuur · 5, 10
Machine Learning · 56, 57, Zie ML
malware · 59
management practice · 50
managementteam · 7
manufacturing process · 82
Mean Time Between Failure · Zie MTBF
Mean Time Between System Incidents · Zie MTBSI
Mean Time To Repair · Zie MTTR
menskracht · 32
meta-data · 68
microservice · 27, 75

microservice architecture · 76
Microsoft Operations Framework · Zie MOF
Microsoft SQL Server · 12
Microsoft Windows · 12
mini pipeline · 76
miniaturisering · 8, 13
Minimal Viable Product · Zie MVP
Minimum Required Information · Zie MRI
ML · 11, 55, 56, 57, 58, 59, 84
mobile devices · 34
module · 24
Module Test · Zie MT
MOF · 38, 39, 84
molecular informatics · 8
MongoDB · 12
monitoring · 76
monoliet · 26
monolithic · 76
MRI · 75, 84
MS DOS · 23
MT · 22, 84
MTBF · 84
MTBSI · 85
MTTR · 76, 85
muda · 76
multi-functional · 14
mutatiegraad · 33
MVP · 22, 85
MySQL · 12

N

NAS · 12, 13, 85
netwerk · 5, 8, 9, 12, 14, 15, 16, 19, 24, 43, 57
netwerkaansluiting · 13
netwerkcomponent · 15
netwerkvoorziening · 53
Network Attached Storage · Zie NAS
neurale netwerkarchitectuur · 57
NFR · 22, 69, 76, 85
Non Functional Requirement · Zie NFR
norm · 41

O

obeya · 76
objectcode · 9, 21, 68
OLA · 43, 85
on premise · 10, 16, 19
onderhandelen · 43
one piece flow · 76
ongestructureerde data · 55
online payment system · 10
on-premise · 16
ontstaansgeschiedenis · 5, 7, 22, 23, 33, 39
Ontwikkel-, Test-, Acceptatie- en Productieomgeving · Zie OTAP
ontwikkelproces · 23, 24, 25, 33
ontwikkelteam · 21
operating system · 5, 9, 11, 12, 23, 29, 34, 70
operation · 24, 38, 67, 71, 76, 78, 79, 81, 83
operation · 51
Operational Level Agreement · Zie OLA
operationeel beleid · 7
operations story · 76
Ops liaison · 76
optical disc drive · 14
Oracle RDBMS · 12
organisational typology model · 77
organization archetype · 76
OSI-model · 15
OTAP · 85
outcome · 37, 45, 50, 52
over-the-shoulder · 77

P

PaaS · 6, 7, 16, 53, 85
package · 77
pair programming · 69, 77
password · 12
PAT · 31, 32, 38, 67, 85
pattern · 29, 68, 78
PBI · 22, 85
PDCA · 74, 75, 85
peer review · 77
peer to peer programming · 69
People, Process & Technology · Zie PPT
performance · 69, 82, 85
  - eis · 23, 30, 32, 43
  - engine · 81
  - management · 43
  - verlies · 27
Performance StressTest · Zie PST
PHP · 23, 33
pipeline · 67, 72, 73, 76, 78, 79, 82
Plan Do Check Act · Zie PDCA
Platform as a Service · 16, Zie PaaS
plug-in · 10
poort · 13
POR · 85
portfolio · 46
post mortem · 77
PostgresSQL · 12
power aansluiting · 13
Powershell · 23
powersupply · 19
PPT · 22, 85
prijs · 46
prijs prestatieverhouding · 48
prijsstelling · 43
printer · 14
prioriteit · 21, 47
private cloud · 16, 18, 55
problem management · 49
procedure · 30, 41
procesmanager · 78
procespattern · 78
process mining tool · 11

Processing Time · Zie PT
processor · 16
product backlog · 77
product backlog item · 73
product owner · 77
Productie AcceptatieTest · Zie PAT
Productie Backlog Item · Zie PBI
productieomgeving · 46, 47
production environment · 76
programmatuur · 5, 10, 33, 37
programmeertaal · 8, 10, 22, 23, 33, 34
programmeur · 11, 21, 32, 33
programming paradigm · 77
ProgreSQL · Zie PSQL
Project or Organisational Risk · Zie POR
Prolog · 22, 23
PSQL · 70, 85
PST · 31, 32, 38, 85
PT · 39, 85
public cloud · 16, 18, 55
public cloud service · 16
pull request process · 77
Python · 8, 10, 23

Q

QA · 39, 77, 85
QC · 39, 85
Quality Assurance · Zie QA
Quality Control · Zie QC

R

RACI · 39, 85
rack · 19, 53
randapparatuur · 9, 13
rapportage · 48
RASCI · 39, 85
RBAC · 85
Reactive AI · 56, 57
Real User Monitoring · Zie RUM
recovery · 24
reduce batch size · 77
reduce number of handoffs · 77
regressieve model · 58
rekencentrum · 16
rekenkracht · 16, 23
relatiebeheer · 47
release · 48, 78
release and deployment management · 48
release management · 42
repository · 68, 69, 77, 78
REpresentational State Transfer Application Programming Interface · 27, Zie REST API
Request for Change · Zie RFC
request fulfilment · 49
requirement · 23, 24, 29, 30, 32, 50, 52, 53, 68, 75, 76, 79, 85
resource planning · 48
Responsibility, Accountable, Consulted and Informed · Zie RACI
Responsibility, Accountable, Supporting, Consulted and Informed · Zie RASCI
REST API · 15, 22, 27, 85
retrospective · 71
Return On Investment · Zie ROI
review · 71
RFC · 48, 85
richtlijn · 46
risico · 5, 7, 21, 39, 46, 48, 68, 75
risicoanalyse · 24, 25
risicobeheer · 48
robotisering · 55
ROI · 22, 85
Role-based access control · Zie RBAC
ROM · 14
rootcause analyse · 75
router · 5, 15, 76
RPG III · 22, 23
RUM · 85

S

SA · 85
SaaS · 6, 7, 10, 16, 37, 85
sad path · 78
SAFe · 22, 85
safety check · 78
SAN · 12, 13, 85
SAP Sybase ASE · 12
Sarbanes Oxley · Zie SoX
SAT · 39, 85
SBAR · 78, 85
SBB · 22, 85
SBB-A · 85
SBB-I · 85
SBB-T · 85
Scaled Agile Framework · Zie SAFe
scanner · 14
schoningsgegevens · 30
S-CI · 85
Secure Sockets Layer · Zie SSL
security · 69, 70, 76, 78, 81
Security AcceptatieTest · Zie SAT
Self service capability · 78
server blade · 53
serverblade · 19
service · 85
- afspraak · 43
- beschrijving · 46
- catalogue management · 46
- design · 46
- desktool · 41
- hiërarchie · 46
- level · 41, 43
- levenscyclus · 48
- norm · 41, 42
- operation · 49
- portfolio · 44, 46
- provider · 49
- specificatie · 47

- strategie · 45
- strategy · 44
- transition · 47
- validation and testing · 48
- value chain · 50
- verlening · 37, 42, 43, 45, 50, 52, 53
Service Level Agreement · Zie SLA
Service Value System · Zie SVS
shared code repository · 51
shared goals · 78
silo · 81
Simian army · 78, 80, 81
Simple Network Management Protocol · Zie SNMP
SIT · 22, 85
skill · 67, 73
SLA · 32, 39, 41, 42, 43, 45, 46, 47, 48, 49, 50, 63, 85
- afspraak · 43
- norm · 32, 41
- normafwijking · 48
- rapportage · 41, 42
SM · 85
SMART · 10, 39, 75, 85
SMART phone · 10
SME · 39, 86
smoke testing · 78
SNMP · 86
SoE · 22, 79, 86
software · 16, 41, 42, 68, 80, 82
- ontwikkeling · 21, 38, 55
- ontwikkelproces · 21, 52
- pakket · 10, 12
Software as a Service · 16, Zie SaaS
Software Configuration Item · Zie S-CI
SoI · 22, 79, 86
SOLL · 74
SoR · 22, 79, 86
sourcecode · 9, 11, 21, 30, 33, 35, 68, 69, 71, 78, 80, 81
SoX · 39, 86
Specific, Measurable, Accountable, Realistic, Timely · Zie SMART
specificatie · 48
sprint · 24, 25, 33, 51, 71
- backlog · 24
- execution · 71
- planning · 71
SQL · 9, 11, 12, 22, 23, 30, 86
SRG · 22, 86
SSD · 14
SSL · 86
ST · 22, 86
stack · 5, 8, 9, 15, 16, 27, 53
stakeholder · 72, 73
standaard change · 49
standaard functie · 21
standard deviation · 79
standard operations · 79
Standard Rules & Guidelines · Zie SRG
stand-up · 71
State Transfer · 27
static analysis · 79
Static analysis · 79
storage · 5, 12, 13, 19, 68, 79, 81
Storage Area Network · Zie SAN
Strategic IS Architecture · Zie SA
Strategic Match · Zie SM
strategy management for IT services · 44
Structured Query Language · Zie SQL
Subject Matter Expert · Zie SME
supplier management · 47
Supply zijde · 53
support · 47
support organisatie · 43
support set · 41
SVS · 50, 86
SW · 13
Swift · 23
switch · 5, 15
systeem
- ontwikkeling · 25, 56
- programmatuur · 5, 6, 10, 11, 15, 53
- software · 30
System Building Block · Zie SBB
System Building Block Application · Zie SBB-A
System Building Block Infrastructure · Zie SBB-I
System Building Block Technology · Zie SBB-T
System Integration Test · Zie SIT
System of Engagement · Zie SoE
System of Records · Zie SoR
System Test · Zie ST
Systems of Information · Zie SoI

T

taalgebruik · 34
tabel · 30
Taken, Verantwoordelijkheden en Bevoegdheden · Zie TVB
takenpakket · 6
task · 73
TCO · 39, 86
TCP · 86
TDD · 22, 32, 51, 80, 86
Team Foundation Server · Zie TFS
technical debt · 70, 71
Technical Uncertainty · Zie TU
technisch ontwerp · 30, 31, 32
technology adaption curve · 79
technology executive · 79
test
- case · 31, 32, 33, 67, 68, 69, 78
- er · 71
- harness · 80
- plan · 48
- resultaat · 48
Test Driven Development · 32, 86, Zie TDD
TFS · 22, 86
the 7-step improvement process · 49
The Agile Manifesto · 80
the ideal testing automation pyramid · 80

The Lean movement · 80
the non-ideal testing automation inverted pyramid · 80
The Three Ways · 76, 81
theory of constraints · 81
Time To Market · Zie TTM
toepassingsprogrammatuur · 5, 9, 10, 12, 53
Toepassingsprogrammatuur · 5, 10
tool-assisted code review · 81
top-down · 80
Total Cost of Ownership · Zie TCO
Toyota Kata · 81
Toyota Production System · Zie TPS
TPS · 86
Transact Structured Query Language · Zie TSQL
transactie · 8
transformatie · 56
transformation team · 81
transistor · 8
transitie · 47
Transmission Control Protocol · Zie TCP
trend · 1, 55
trends · 3
trigger · 31
trunk · 78
T-shaped · 73
TSQL · 70, 86
TTM · 22, 86
TU · 86
TVB · 39, 86

U

UAT · 39, 67, 86
UCMDB · 46
uitwijkplan · 43
UML · 86
Unified Modeling Language · Zie UML
Unit Test · Zie UT
unit testcase · 32
Universal Configuration Management Data Base · *Zie* UCMDB
up front architectuur · 25
USB poort · 13
use case · 29
User Acceptance Test · Zie UAT
User eXperience Design · Zie UX design
userstory · 30, 32
UT · 22, 86
utility · 45
UX-design · 29, 30, 86

V

vakgebied · 38, 40
value stream · 37, 38, 50, 52, 56, 58, 71, 75, 78, 79, 81, 86
Value Stream Mapping · Zie VSM
veilig · 48
velocity · 68
verandertraject · 47
verdienmodel · 34
verkeer · 15
versie · 42
verstoring · 42
vertical splitting of feature · 81
vertrouwelijkheid · 47, 55
verwachting · 48
virtual machine · 15, 29
virtualisatie · 5, 15, 29, 37
virtualized environment · 81
visualisatie · 81
Voice over Internet Protocol · Zie VOIP
VOIP · 86
VSM · 22, 81, 86

W

waardecreatie · 46
walking skeleton · 82
WAN · 14, 86
war room · 76
warranty · 45
waste · 68, 70, 74, 75, 76, 80, 81, 82
waste reductie · 52, 82
waterval · 21, 23, 33, 35, 56
waterval aanpak · 33
Way of Working · Zie WoW
werklast · 53
Wide Area Network · Zie WAN
wijzigingsverzoek · 41, 42
windmolenpark · 20
Windows · 8, 11, 34, 86
Windows Management Instrumentation · Zie WMI
WIP · 22, 77, 82, 86
WMI · 86
Work In Progress · Zie WIP
workflow · 70
WoW · 22, 86

X

XP · 86
XP programming · 33

Z

zelfsturing · 21

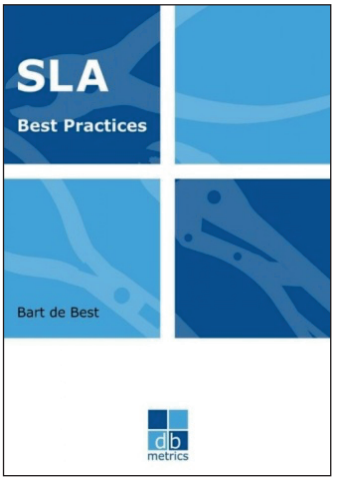
Nawoord

Mijn ervaring is dat de denkbeelden die ik vastleg in een artikel of een boek zich blijven evolueren. In geval u met een bepaald onderwerp uit dit boek aan de slag gaat in uw eigen DevOps organisatie, dan raad ik u aan om even met mij contact op te nemen. Wellicht zijn er aanvullende artikelen of ervaringen op dit gebied die ik met u kan delen. Dit geldt ook omgekeerd evenredig. Als u bepaalde ervaringen hebt die een aanvulling zijn op hetgeen in dit boek is beschreven, dan nodig ik u uit om dit met mij te delen. U kunt mij bereiken via mijn e-mail adres bartb@dbmetrics.nl.

Over de auteur

Drs. Ing. B. de Best RI is vanaf 1985 werkzaam in de ICT. Hij heeft voornamelijk bij de top 100 van het Nederlandse bedrijfsleven en de overheid gewerkt. Hierbij heeft hij gedurende 12 jaar functies vervuld in alle fasen van de systeemontwikkeling, inclusief exploitatie en beheer. Daarna heeft hij zich toegelegd op het service management vakgebied. Momenteel vervult hij als consultant alle aspecten van de kennislevenscyclus van service management, zoals het schrijven en geven van trainingen aan ICT-managers en service managers, het adviseren van beheerorganisaties bij het richting geven aan de beheerorganisatie, de beheerinrichting, het verbeteren van beheerprocessen, het uitbesteden van (delen van) de beheerorganisatie en het reviewen en auditen van beheerorganisaties. Hij is op zowel HTS-niveau als Universitair niveau afgestudeerd op het beheervakgebied.

Andere boeken van deze auteur



SLA Best Practices

Het volledige ABC van service level agreements.

Het belangrijkste bij het leveren van een service is dat de klant tevreden is over de geleverde prestaties. Door deze tevredenheid verkrijgt de leverancier heraankopen, wordt hij gepromote in de markt en is de continuïteit van het bedrijf geborgd. Wellicht nog het belangrijkste aspect van deze klanttevredenheid voor een leverancier is dat de betrokken medewerkers een drive krijgen om hun eigen kennis en kunde verder te ontwikkelen om nog meer klanten tevreden te stellen. Dit boek beschrijft de best practices om erachter te komen wat de Prestatie-Indicatoren (PI's) zijn die gemeten moeten worden om de tevredenheid van de klant te borgen. Het tweede deel beschrijft de documenten die van toepassing zijn om de afspraken in vast te leggen. Het opstellen, afspreken, bewaken en evalueren van serviceafspraken is een vak op zich.

Het derde deel geeft de gereedschappen om hier adequaat invulling aan te geven. De werkzaamheden rond serviceafspraken herhalen zich in de tijd. Deel vier van dit boek beschrijft hoe deze werkzaamheden in een proces gevat kunnen worden en hoe dit proces het beste in de organisatie kan worden vormgegeven. Tot slot geeft bespreekt dit boek een aantal raakvlakken van serviceafspraken en een tweetal artikelen met SLA best practices.

Auteur : Bart de Best
Uitgever : Leonon Media, 2011
ISBN (NL) : 978 90 7150 1456

Cloud SLA

De best practices van cloud service level agreements

Steeds meer organisaties kiezen voor het vervangen van de traditionele ICT-services door cloud services. Het opstellen van doelmatige SLA's voor traditionele ICT-services is voor veel organisaties een ware uitdaging. Met de komst van cloud services lijkt dit in eerste instantie veel eenvoudiger, maar al snel komen de moeilijke vragen aan bod zoals data eigenaarschap, informatiekoppelingen en beveiliging.

Dit boek beschrijft wat cloud services zijn. Daarbij wordt ingegaan op de risico's die organisaties lopen bij het aangaan van contracten en SLA's. Op basis van een lange lijst van risico's en tegenmaatregelen geeft dit boek tevens aanbevelingen voor de opzet en inhoud van de diverse service level management documenten voor cloud services. Dit boek definieert eerst het begrip 'cloud' en beschrijft daarna diverse aspecten zoals cloud patronen en de rol van een cloud broker. De kern van het boek betreft het bespreken van de contractaspecten, service documenten, service designs, risico's, SLA's en cloud governance. Om de lezer gelijk aan de slag te kunnen laten gaan met cloud SLA's zijn in het boek tevens checklists opgenomen van de volgende documenten: Underpinning Contract (UC), Service Level Agreement (SLA), Dossier Financiële Afspraken (DFA), Dossier Afspraken en Procedures (DAP), External SpecSheets (ESS) en Internal Specsheets (ISS).

Auteur : Bart de Best
Uitgever : Leonon Media, 2014
ISBN (NL) : 978 90 7150 1739
ISBN (UK) : 978 94 9261 8009

SLA Templates

A complete set of SLA templates

The most important thing in providing a service Is that the customer is satisfied with the delivered performance. With this satisfaction, the supplier gets re-purchasing's, promotions in the market and is the continuity of the company ensured. Perhaps the most important aspect of this customer satisfaction for a supplier is that the employees in question get a drive to further develop their own knowledge and skills to satisfy even more customers. This book describes the templates for Service Level Agreements in order to agree with the customer on the required service levels. This book gives both a template and an explanation for this template for all common service level management documents.

The following templates are included in this book:

- Service Level Agreement (SLA)
- Underpinning Contract (UC)
- Operational Level Agreement (OLA)
- Document Agreement and Procedures (DAP)
- Document Financial Agreements (DFA)
- Service Catalogue
- External Spec Sheet (ESS)
- Internal Spec Sheet (ISS)
- Service Quality Plan (SQP)
- Service Improvement Program (SQP)

Author	Bart de Best
Publisher	Leonon Media, 2017
ISBN (UK)	978 94 92618 030
ISBN (Pocket Guide)	978 94 92618 320

ICT Prestatie-indicatoren

De beheerorganisatie meetbaar gemaakt.

De laatste jaren is het maken van concrete afspraken over de ICT-serviceverlening steeds belangrijker geworden. Belangrijke oorzaken hiervoor zijn onder meer de stringentere wet- en regelgeving, de hogere eisen die gesteld worden vanuit regie-voering over uitbestede services en de toegenomen complexi-teit van informatiesystemen. Om op de gewenste service-normen te kunnen sturen, is het belangrijk om een Performance Measurement System (PMS) te ontwikkelen. Daarmee kunnen niet alleen de te leveren ICT-services worden gemeten, maar tevens de benodigde ICT-organisatie om de ICT-services te verlenen. Het meten van prestaties is alleen zinvol als bekend is wat de doelen zijn van de opdrachtgever.

Daarom start dit boek met het beschrijven van de bestuurlijke behoefte van een organisatie en de wijze waarop deze vertaald kunnen worden naar een doeltreffend PMS. Het PMS is hierbij samengesteld uit een meetinstrument voor de vakgebieden service management, project management en human resource management. Voor elk van deze gebieden zijn tevens tal van prestatie-indicatoren benoemd. Hiermee vormt dit boek een onmisbaar instrument voor zowel ICT-managers, kwaliteitsmanagers, auditors, service managers, project managers, programma managers, proces managers, als human resource managers.

Auteur	: Bart de Best
Uitgever	: Leonon Media, 2011
ISBN (NL)	: 978 90 7150 1470

Quality Control & Assurance

Kwaliteit op maat.

De business stelt steeds hogere eisen aan de ICT-services die ICT-organisaties leveren. Niet alleen nemen de eisen van de overheid toe in de vorm van wet- en regelgeving, ook de dynamiek van de markt wordt hoger en de levenscyclus van business producten korter. De reactie van veel ICT-organisaties hierop is het hanteren van kwaliteitsmodellen zoals COBIT, ITIL, TOGAF en dergelijke.

Helaas verzandt het toepassen van de best practices van deze modellen vaak omdat het model als doel wordt verklaard, hierdoor ontstaat veel overhead. Nut en noodzaak worden niet onderscheiden. In het beste geval is de borging van kwaliteit een golfbeweging met pieken en dalen waarop maar weinig grip op te krijgen is. Dit boek bespreekt op welke wijze de keuze voor kwaliteit concreet en kwantitatief gemaakt kan worden alsmede hoe de kwaliteit in de ICT-organisatie verankerd kan worden. De voorgestelde aanpak omvat zowel Quality Control (opzet en bestaan) als Quality Assurance (werking) voor ICT-processen. Hierbij worden de eisen die aan de ICT-organisatie worden gesteld vertaald naar procesrequirements (opzet) en worden deze binnen ICT-processen geborgd (bestaan). Periodiek worden deze gemeten (werking). Door requirementste classificeren naar tijd, geld, risicobeheersing en volwassenheid kan het management een bewuste keuze maken voor de toepassing van requirements. Hierdoor wordt kwaliteit meetbaar en blijft de overhead beperkt. Dit boek is een onmisbaar instrument voor kwaliteitsmanagers, auditors, lijnmanagers en proces managers.

Auteur	: Bart de Best
Uitgever	: Leonon Media, 2012
ISBN (NL)	: 978 90 7150 1531

Acceptatiecriteria

Naar een effectieve en efficiënte acceptatie van producten en services in de informatietechnologie.

Acceptatiecriteria zijn een meetinstrument voor zowel gebruiker als beheerder om te bepalen of nieuwe of gewijzigde informatiesystemen voldoen aan de afgesproken requirements ten aanzien van functionaliteit, kwaliteit en beheerbaarheid. Er komt heel wat bij kijken om acceptatiecriteria te verankeren in beheerprocessen en systeem-ontwikkelingsprojecten. Het opstellen en het hanteren van acceptatiecriteria voor ICT-producten en ICT-services geschiedt bij veel organisaties met wisselend succes. Vaak worden acceptatiecriteria wel opgesteld, maar niet effectief gebruikt en verworden ze tot een noodzakelijk kwaad zonder kwaliteitsborgende werking. Dit boek geeft een analyse van de oorzaken van dit falen van de kwaliteitsbewaking. Als remedie worden drie stappenplannen geboden voor het afleiden, toepassen en invoeren van acceptatiecriteria. De doelgroep van dit boek omvat alle partijen die betrokken zijn bij de acceptatie van ICT-producten en ICT-services: de klanten, de leveranciers en de beheerders. Ook is er nog een doelgroep die niet accepteert, maar vaststelt of correct is geaccepteerd; hiertoe behoren kwaliteitsmanagers en auditors die het boek als normenkader kunnen gebruiken. In dit boek is een aantal casussen opgenomen die diverse manieren laten zien voor het effectief en efficiënt omgaan met acceptatiecriteria.

Auteur	: Bart de Best
Uitgever	: Leonon Media, 2014
ISBN (NL)	: 978 90 7150 1784

Ketenbeheer in de praktijk

De keten meetbaar gemaakt.

Ketenbeheer is een fenomeen dat vele gezichten heeft. Voor de een is het een aaneenschakeling van informatiesystemen, voor de ander een samenhangend geheel van beheerprocessen of bedrijfsprocessen. Dit boek definieert ketenbeheer als de verzameling van beheerprocessen en beheerproducten die ervoor zorgdragen dat een set bij elkaar horende en aansluitende bedrijfsprocessen als geheel efficiënt en effectief functioneren. Hierbij worden drie ketens onderkend: op beheerprocesniveau, informatiesysteemniveau en bedrijfsprocesniveau. De nadruk ligt op het verkrijgen van synergie door het zoveel mogelijk uitwisselen van statusinformatie over de ketens die zich op deze drie niveaus bevinden: integraal ketenbeheer.

Ketenbesturing is nog jong. Dit boek geeft in een vogelvlucht de stand van zaken weer aan de hand van een onderzoek, in opdracht van Qforce door de auteur uitgevoerd, naar de functionaliteit van vier innovatieve leveranciers van tools voor ketenbeheer met elk hun eigen specialisatie. Tegelijkertijd is het boek een bruikbare hands-on, als aanvulling op de management theorieën over dit onderwerp. Een goed beeld van ketenbeheer in de praktijk is de in dit boek opgenomen casus waarin Fortis Nederland ketens monitort. Deze organisatie heeft ook een praktijkvoorbeeld gegeven van het inrichten van een ketenbeheerorganisatie. Dit eerste werk in Nederland over integraal ketenbeheer is een onmisbaar hulpmiddel bij de integratie van bedrijfs- en beheerprocessen.

Auteur	: Bart de Best
Uitgever	: Leonon Media, 2015
ISBN (NL)	: 978 90 7150 1852

Agile Service Management met Scrum

Op weg naar een gezonde balans tussen de dynamiek van het ontwikkelen en de stabiliteit van het beheren van de informatievoorziening.

Het toepassen van Agile software development neemt een grote vlucht. De termen Scrum en Kanban zijn al ingeburgerd bij menig organisatie. Agile software development stelt andere eisen aan de invulling van beheer van programmatuur. Veel organisaties zijn dan ook bezig om zich over deze nieuwe uitdaging te buigen. Vooral de interactie tussen het Scrum-ontwikkelproces en het beheren van de programmatuur die het Scrum-ontwikkelproces heeft opgeleverd is hierbij een belangrijk aspectgebied. Dit boek bespreekt juist deze interactie. Voorbeelden van onderwerpen die hierbij ter sprake komen zijn het service portfolio, SLA's en de afhandeling van incidenten en wijzigingsverzoeken. Dit boek definieert eerst de risicogebieden bij het invoeren van Scrum en Kanban. Daarna worden de diverse Agile begrippen en concepten besproken. De invulling van Agile service management is zowel op organisatieniveau als op procesniveau beschreven. Hierbij zijn per beheerproces de relevante risico's benoemd. Tevens is aangegeven hoe hier binnen de context van Scrum invulling aan gegeven kan worden.

Auteur	Bart de Best
Uitgever	Leonon Media, 2014 (NL), 2018 (UK)
ISBN (NL)	978 90 7150 1807
ISBN (UK)	978 94 9261 8085

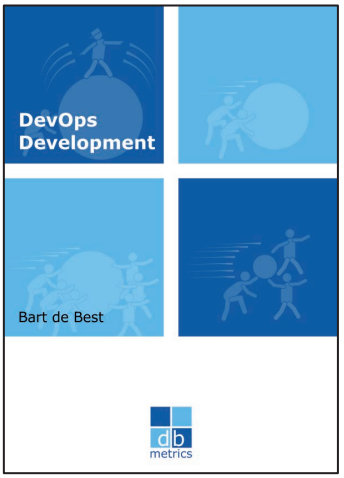
Agile Service Management met Scrum in de Praktijk

Op weg naar een gezonde balans tussen de dynamiek van het ontwikkelen en de stabiliteit van het beheren van de informatievoorziening.

Veel bedrijven zijn bezig om Agile softwareontwikkeling toe te gaan passen in de vorm van Scrum of Kanban of hebben het nieuwe ontwikkelproces al in gebruik genomen. Vroeg of laat komt dan de vraag hoe dit ontwikkelproces zich verhoudt tot de beheerprocessen. In het boek 'Agile Service Management met Scrum' is al naar deze interface gekeken en zijn een aantal risico's per beheerproces onderkend. Tevens zijn tegen-maatregelen gedefinieerd die genomen kunnen worden. In een onderzoek bij tien organisaties zijn deze risico's voorgelegd en is gevraagd hoe zij met deze risico's zijn omgegaan.

Tevens is onderzocht welke Agile aspecten worden toegepast en in het bijzonder die van Scrum of Kanban. Tot slot is door elke organisatie een volwassenheidsassessment uitgevoerd voor zowel het Agile ontwikkelproces als het change management proces. Dit boek is het rapport over het onderzoek naar de samenwerking van Agile software ontwik-keling en beheer-processen in de praktijk. De doelgroep van dit boek omvat alle partijen die betrokken zijn bij de toepassing van Agile software ontwikkeling en die graag eens willen weten hoe collega's deze cruciale interface voor een succesvolle serviceverlening hebben vormgegeven. In dit boek is tevens van elke organisatie een korte beschrijving gegeven over de wijze waarop het Agile ontwikkelproces is vormgegeven.

Auteur : Bart de Best
Uitgever : Leonon Media, 2015 (NL), 2018 (UK)
ISBN (NL) : 978 90 7150 1845
ISBN (UK) : 978 94 9261 8177



DevOps Development

Een set van best practices voor het ontwikkelen van software in een DevOps omgeving

De wereld van softwareontwikkeling is snel aan het veran-deren. Development (Dev) en Operations (Ops) worden meer en meer geïntegreerd zodat oplossingen sneller aan klanten kunnen worden geboden met een hogere kwaliteit. Veel DevOps engineers vragen zich af welke nieuwe inzichten er zijn op het gebied van softwareontwikkeling om invulling te geven aan de hogere time to market en een hogere kwaliteit. Dit boek voorziet in een antwoord op deze vraag.

Dit boek bestaat uit vier kernhoofdstukken te weten Continuous Planning, Continuous Documentation, Continuous Testing and Continuous Assessment.
Elk van deze kernhoofdstukken beschrijft een aantal best practices aan de hand van een definitie, de te beheersen risico's, de anti-pattern en de pattern. Op deze wijze bent u snel weer up-to-date op het gebied van de DevOps ontwikkelingen.

Author : Bart de Best
Publisher : Leonon Media, 2021 (NL), 2021 (UK)
ISBN (NL) : 978 94 92618 054
ISBN (UK) : 978 94 92618 184

DevOps Best Practices

Best Practices for DevOps

In recent years, many organisations have experienced the benefits of using Agile approaches such as Scrum and Kanban. The software is delivered faster whilst quality increases and costs decrease. The fact that many organisations that applied the Agile approach did not take into account the traditional service management techniques, in terms of information management, application management and infrastructure management, is a major disadvantage. The solution to this problem has been found in the Dev (Development) Ops (Operations) approach. Both worlds are merged into one team, thus sharing the knowledge and skills. This book is about sharing knowledge on how DevOps teams work together.

For each aspect of the DevOps process best practices are given in 30 separate articles. The covered aspects are Plan, Code, Build, Test, Release, Deploy, Operate and Monitor. Each article starts with the definition of the specifically used terms and one or more concepts. The body of each article is kept simple, short, and easy to read.

Author : Bart de Best
Publisher : Leonon Media, 2017 (NL), 2018 (UK)
ISBN (UK) : 978 94 92618 078
ISBN (Pocket Guide) : 978 94 92618 306

DevOps Architectuur

DevOps Architectuur Best Practices

De wereld van systeemontwikkeling is in een snel tempo aan het veranderen. Daarbij worden Development (Dev) en Operations (Ops) steeds meer geïntegreerd zodat oplossingen sneller en kwalitatief beter aan de klant kunnen worden aangeboden. De vraag is hoe binnen deze nieuwe zienswijze van DevOps plaats is voor Agile architectuur. Dit boek geeft een antwoord op deze vraag door het geven van vele voorbeelden van architectuurprincipes- en modellendie richting geven aan de inrichting en de verrichting van een DevOps organisatie. In het hele boek wordt zo veel als mogelijk per paragraaf een toelichting gegeven op basis van een denkbeeldig bedrijf Assuritas.

Dit boek bestaat uit verschillende onderdelen hetgeen het boek modulair maakt. Het hoeft dus niet van A to Z gelezen te worden. Na de korte schets van het casusbedrijf volgt de bespreking van de DevOps organisatie vanuit een architectuurperspectief. Daarna wordt de DevOps beheervoorziening besproken. Beide verhandelingen worden aan de hand van het casusbedrijf inzichtelijk gemaakt. Na de behandeling van de integratie van de Dev- en Ops-rollen volgen twee handige analysetools om de volwassenheid van DevOps te bepalen. Het boek sluit af met een casus waarin op basis van architectuurprincipes en -modellen de keuze voor een Agile documentatie wordt gemaakt. Dit werk over DevOps architectuur is een onmisbaar hulpmiddel bij de vormgeving en uitvoering van een DevOps serviceorganisatie.

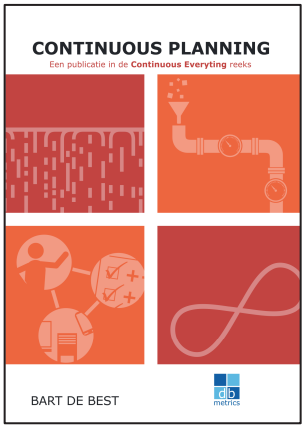
Author : Bart de Best
Publisher : Leonon Media, 2019
ISBN (NL) : 978 94 92618 061
ISBN (UK) : 978 90 71501 579

Beheren onder Architectuur
Het richting geven aan de inrichting van beheerorganisaties.

Veel organisaties zijn al jaren bezig met het vormgeven van de beheerorganisatie door vanaf de werkvloer te kijken wat er fout gaat en op basis daarvan verbetervoorstellen te formuleren. Hierbij wordt meestal gebruik gemaakt van beheermodellen, zoals ITIL, ASL en BiSL, omdat deze veel best practices bevatten. Deze bottom-up benadering werkt een lange tijd goed. De afstemming van de beheerorganisatieinrichting op de behoefte van de business is daarmee echter nog geen feit. Het wezenlijke verschil met een top-down benadering is dat er eerst een kader gesteld wordt dat richting geeft aan de inrichting van de beheer-organisatie. Dit kader bestaat uit beleidsuitgangspunten, architectuurprincipes en architectuur-modellen. Deze richtinggevendheid is ook van toe passing op de projectorganisatie waarin de projecten worden vormgegeven die beheerd moeten gaan worden. Dit boek ondersteunt dit gedachtegoed door stap voor stap aan deze richtinggevendheid invulling te geven. Het eerste deel van dit boek positioneert het gedachtegoed van beheren onder architectuur binnen de wereld van de informatievoorzieningsarchitectuur. Het tweede deel beschrijft een stappenplan om invulling te geven aan dit gedachtegoed aan de hand van vele best practices en checklists. Het derde deel beschrijft hoe beheren onder architectuur in de organisatie kan worden ingebed. Tot slot geeft het vierde deel een negental casussen van organisaties die het aangereikte stappenplan al hebben toegepast. Zoals de casus van het agentschap BPR (Basisadministratie Persoonsgegevens en Reisdocumenten), waarin beschreven is hoe zij een blauwdruk gebruiken voor de inrichting van de beheerorganisatie. Dit boek is primair gericht op beheerarchitecten, ICT-architecten, business architecten, service managers, ICT-managers, programmamanagers, beleidsmakers, en overige functionarissen die betrokken zijn bij het beheren onder architectuur.

Auteur : Bart de Best
Uitgever : Leonon Media, 2017
ISBN (NL) : 978 90 7150 1913

Continuous Everything boeken

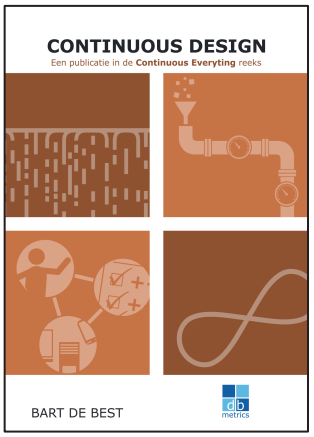


Continuous Planning

Een uitgave in de Continuous Everything reeks.

Continuous Planning is een aanpak om grip te krijgen op veranderingen die aangebracht worden in de informatievoorziening teneinde de outcome verbetering van de bedrijfsprocessen te realiseren en daarmee de bedrijfsdoelen te behalen. De aanpak is gericht op meer niveaus waarbij voor elke niveau een Agile planningstechniek wordt aangereikt die de hoger liggende planning verfijnt. Op deze manier kan er zowel op strategisch, tactisch als operationeel niveau een planning worden gemaakt en wel op een Agile wijze die zo min mogelijk overhead en zoveel mogelijk waarde creëren. Dit boek is een publicatie in de Continuous Everything reeks. De content bestaat uit de bespreking van de planningstechnieken zoals de balanced scorecard, enterprise architectuur, product vision, roadmap, epic one pager, product backlog management, release planning en sprint planning. Tevens wordt aangegeven hoe deze technieken aan elkaar zijn gerelateerd. Daarnaast geeft dit boek aan hoe Continuous Planning in te richten in uw organisatie op basis van het paradigma van de verander-manager en architectuurprincipes en -modellen. Met deze integrale Agile benadering van planning heeft u een krachtig gereedschap in handen om de strategie van uw organisatie planmatig op te pakken en daarmee uw bedrijfsdoelen te realiseren.

Author	: Bart de Best
Publisher	: Leonon Media, 2021
ISBN (NL)	: 978 94 92618 504
ISBN (UK)	: -



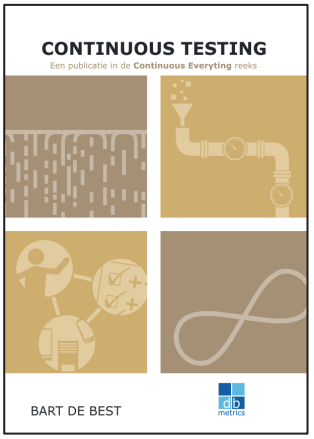
Continuous Design

Een uitgave in de Continuous Everything reeks.

Continuous Design is een aanpak die beoogt om DevOps teams vooraf kort na te laten denken over de contouren van het te realiseren informatiesysteem en tijdens het Agile project het design te laten groeien (emerging design). Hierdoor worden interface risico's voorkomen en wordt essentiële kennisoverdracht geborgd ter ondersteuning van beheer en het nakomen van wet- en regelgeving. Elementen die de continuïteit van een organisatie waarborgen. Dit boek is een publicatie in de Continuous Everything reeks. De content bestaat uit het Continuous Design Pyramid model waarin de volgende design views zijn gedefinieerd: business, solution, design, requirements, test en code view.

Het Continuous Design omvat de gehele lifecycle van het informatiesysteem. De eerste drie views worden ingevuld op basis van moderne ontwerptechnieken zoals de value streammapping en use cases. De nadruk van het effectief toepassen van een Continuous Design ligt echter in de realisatie van het informatiesysteem en wel door het design te integreren in de Behaviour Driven Development en Test Driven Development alsmede in Continuous Documentation. Met deze Agile benadering van een design heeft u een krachtig gereedschap in handen om grip te krijgen op een Agile ontwikkelproject.

Author	: Bart de Best
Publisher	: Leonon Media, 2021
ISBN (NL)	: 978 94 92618 481
ISBN (UK)	: -



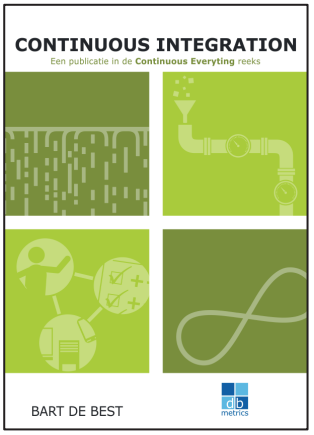
Continuous Testing

Een uitgave in de Continuous Everything reeks.

Continuous Testing is een aanpak die beoogt om een snelle feedback te geven in het software ontwikkelproces door de 'wat'- en' hoe'-vragen te definiëren als testcases voordat gestart wordt met de bouw van de oplossing. Hierdoor worden de concepten requirements, testcases en acceptatiecriteria geïntegreerd in één aanpak. Het begrip 'continuous' verwijst naar het toepassen van test management in alle fasen van de deployment pipeline, dus van requirements tot en met het in productie nemen. Tevens omvat het begrip 'continuous' de aspecten People, Process en Technology. Daarmee wordt test management dus holistisch.

Dit boek is een publicatie in de Continuous Everything reeks. De content bestaat uit het behandelen van Continuous Testing aan de hand van een definitie, business case, architectuur, ontwerp en best practices. Begrippen die hierbij worden besproken zijn: het verander-paradigma, de Ideal Test Pyramid, test meta data, Behaviour Driven Development, Test Driven Development, test policies, test technieken, test tools en de rol van unit testcases in Continuous Testing. Op deze wijze bent u snel weer up-to-date op het gebied van de DevOps ontwikkelingen en op het gebied van Continuous Testing.

Author : Bart de Best
Publisher : Leonon Media, 2021
ISBN (NL) : 978 94 92618 450
ISBN (UK) : -



Continuous Integration

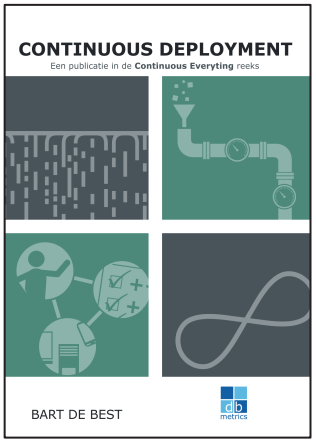
Een uitgave in de Continuous Everything reeks.

Continuous Integration is een holistische Lean software ontwikkel-aanpak die beoogt om op een incrementele en iteratieve wijze continue software te produceren en in productie te nemen waarbij waste reductie hoog in het vaandel staat.

Het woord 'holistisch' verwijst naar de PPT-begrippen: People (multiple expert), Process (kennis van bedrijfs- en beheer-processen) en Technology (applicatie en infrastructuur program-meren). Door de incrementele en iteratieve werkwijze wordt fast feedback mogelijk doordat functionaliteiten eerder in productie kunnen worden genomen. Hierdoor wordt waste gereduceerd omdat gebreken eerder worden gevonden en sneller kunnen worden hersteld.

Dit boek is een publicatie in de Continuous Everything reeks. De content bestaat uit het behandelen van Continuous Integration aan de hand van een definitie, business case, architec-tuur, ontwerp en best practices. Begrippen die hierbij worden besproken zijn het verander-paradigma, het toepassen van Continuous Integration, gebruik repositories, code kwaliteit, green code, green build, refactoring, security based development en built-in failure mode. Op deze wijze bent u snel weer up-to-date op het gebied van de DevOps ontwikkelingen met betrekking tot Continuous Integration.

Author : Bart de Best
Publisher : Leonon Media, 2021
ISBN (NL) : 978 94 92618 467
ISBN (UK) : -



Continuous Deployment

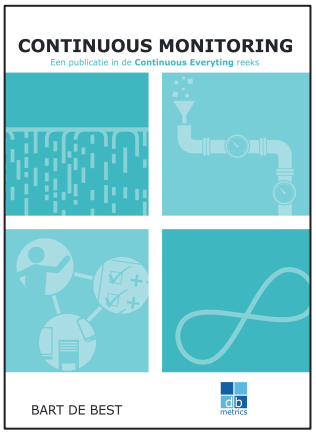
Een uitgave in de Continuous Everything reeks.

Continuous Deployment is een holistische Lean production aanpak die beoogt om op een incrementele en iteratieve wijze continue software te deployen en te releasen waarbij time to market en hoogwaardige kwaliteit hoog in het vaandel staan.

Het woord 'holistisch' verwijst naar de PPT-begrippen: People (multiple expert), Process (kennis van bedrijfs- en beheerprocessen) en Technology (applicatie en infrastructuur programmeren. Door de incrementele en iteratieve deployments wordt fast feedback mogelijk omdat fouten eerder in productie van de CI/CD secure pipeline worden waargenomen. Hierdoor zijn herstelacties sneller en goedkoper hetgeen leidt tot een waste reductie.

Dit boek is een publicatie in de Continuous Everything reeks. De content bestaat uit het behandelen van Continuous Deployment aan de hand van een definitie, business case, architectuur, ontwerp en best practices. Begrippen die hierbij worden besproken zijn het verander-paradigma, het toepassen van Continuous Deployment, een stappenplan voor het planmatig inregelen van Continuous Deployment en vele patterns om deployments te laten plaatsvinden. Op deze wijze bent u snel weer up-to-date op het gebied van de DevOps ontwikkelingen op het gebied van Continuous Deployment.

Author : Bart de Best
Publisher : Leonon Media, 2021
ISBN (NL) : 978 94 92618 511
ISBN (UK) : -



Continuous Monitoring

Een uitgave in de Continuous Everything reeks.

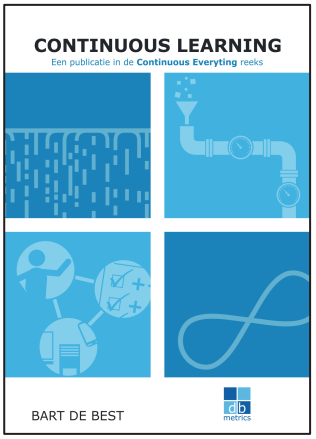
Continuous Monitoring is een aanpak om grip te krijgen op zowel core value streams (business processen) als enable value streams die deze core value streams ondersteunen. Continuous Monitoring onderscheidt zich van de klassieke monitoring door de focus op de outcome verbetering en de holistisch scope waarmee value streams worden gemeten te weten de gehele CI/CD secure pipeline voor alle drie de perspectieven van PPT: People, Process en Technology.

De aanpak omvat People, Process en Technology, hetgeen het mogelijk maakt om de bottlenecks in uw value streams in kaart te brengen en te elimineren of te mitigeren.

Dit boek is een publicatie in de Continuous Everything reeks. De content bestaat uit de bespreking van de monitorfuncties die zijn gedefinieerd in het continuous monitoring lagenmodel. Dit lagenmodel classificeert de op de markt beschikbare monitortools. Elk monitor archetype wordt in dit boek gedefinieerd qua definitie, doelstelling, meetattributen, requirements, voorbeelden en best practices. Tevens geeft dit boek aan hoe Continuous Monitoring in te richten in uw organisatie op basis van het paradigma van de verander-manager en architectuurprincipes en -modellen. Met deze integrale agile benadering van monitoring heeft u een krachtig gereedschap in handen om de controls in te regelen voor de besturing van uw value streams.

Author : Bart de Best
Publisher : Leonon Media, 2021
ISBN (NL) : 978 94 92618 498
ISBN (UK) : -

Continuous Learning



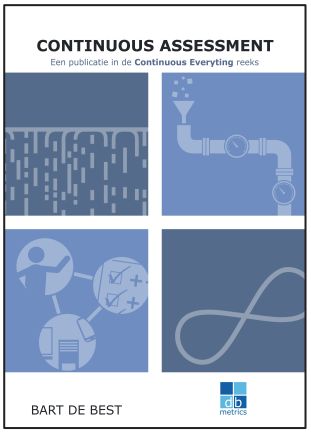
Een uitgave in de Continuous Everything reeks.

Continuous Learning is een aanpak om grip te krijgen op de competenties die nodig zijn om de strategie van uw organisatie te realiseren.

Continuous Learning biedt Human Resource Management hiertoe een aanpak die stap voor stap de organisatiebehoeften competenties verkent en deze behoeften omzet in competentie-profielen. Een competentieprofiel is hierbij gedefinieerd als de set van kennis, kunde en gedrag op een bepaald Bloom level die een bepaald resultaat oplevert. Competentieprofielen worden vervolgens samengevoegd in rollen die op hun beurt functies vormen.

Op deze wijze wordt een Agile functiehuis verkregen. Dit boek is een publicatie in de Continuous Everything reeks. De content bestaat uit de bespreking van het Continuous Learning model dat u helpt om stap voor stap een value chain strategie naar een persoonlijke roadmaps voor medewerkers te vertalen. Tevens geeft dit boek aan hoe Continuous Learning in te richten in uw organisatie op basis van het paradigma van de verander-manager en architectuurprincipes en -modellen. Met deze agile benadering van HRM heeft u een krachtig gereedschap in handen om de competenties op het gewenste niveau van uw organisatie te krijgen..

Author : Bart de Best
Publisher : Leonon Media, 2021
ISBN (NL) : 978 94 92618 528
ISBN (UK) : -



Continuous Assessment

Een uitgave in de Continuous Everything reeks.

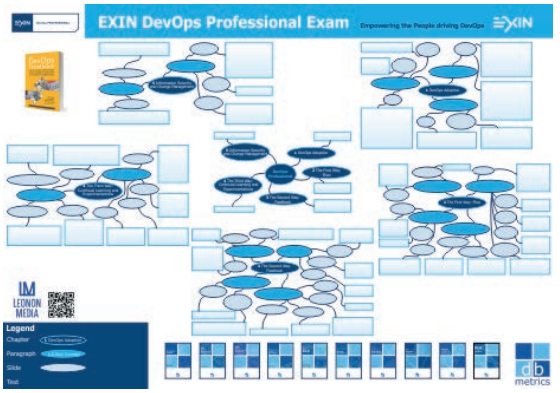
Continuous Assessment is een aanpak die beoogt om DevOps teams zich op een continue wijze zich te laten ontwikkelen qua kennis en kunde op het gebied van business, development, operations en security.

Dit boek geeft een hulpmiddel om de DevOps teams bewust te maken waar zij staan qua ontwikkeling en welke eerstvolgende stappen zij kunnen zetten om zich te ontwikkelen. Dit boek is een publicatie in de Continuous Everything reeks. De content bestaat uit de business case voor Continuous Assessment, de architectuur van de twee assessment modellen en de assessment vragenlijsten.

Het DevOps Cube model is gebaseerd op het idee dat DevOps vanuit zes verschillende perspectieven van een kubus kunnen worden bekeken te weten: ‘Flow’, ‘Feedback’, ‘Continuous Learning’, ‘Governance’, ‘Pipeline’ en ‘QA’.

Het DevOps CE model is gebaseerd op de Continuous Everything perspectieven te weten: ‘Continuous Integration’, ‘Continuous Delivery’, ‘Continuous Testing’, ‘Continuous Monitoring’, Continuous Documentation’ en ‘Continuous Learning’. Dit boek is een uitstekende spiegel voor ieder DevOps team dat snel een compleet beeld wil vormen van op te pakken DevOps best practices.

Author : Bart de Best
Publisher : Leonon Media, 2021
ISBN (NL) : 978 94 92618 474
ISBN (UK) : -



Author : Bart de Best
Publisher : Leonon Media, 2018
Ordering : info@leonon.nl

DevOps Poster

DevOps Professional Exam Poster

This poster lists all the DevOps terms that a student must learn in order to pass the exam of DevOps Professional of Exin. This poster can be ordered at info@leonon.nl.

The subjects on the poster are based on the basic training material of Exin. Since there are many terms to be learned, this poster will help to learn them by reviewing them all at once daily.