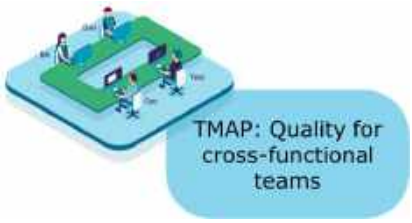


TMAP®:
Quality for cross-functional teams

Syllabus

Version 1.0

Released 16-07-2020

Copyright notice

Revision history

Version	Date	Author	Remarks
0.1	22 April 2020	Ralph Klomp	Initial version
0.2	1 May 2020	Bert Linker	Adding content
0.3	8 May 2020	Bert Linker	Further elaboration after consultation
0.4	13 May 2020	Bert Linker/Rik Marselis	Review remarks and updates
0.5	19 May 2020	Review team	Review remarks and updates
0.6	25 May 2020	Rik Marselis	Aligned order of subjects
0.7	27 May 2020	Rik Marselis	Management review
0.8	2 June 2020	Rik Marselis	Sent for review by iSQI
0.9	9 June 2020	Rik Marselis	Distributed for review by members of TMAP Special Interest Group
0.95	9 July 2020	Rik Marselis	Incorporated all feedback from SIG members and from pilot training.
1.0	16 July 2020	Rik Marselis	Final version

Table of Contents

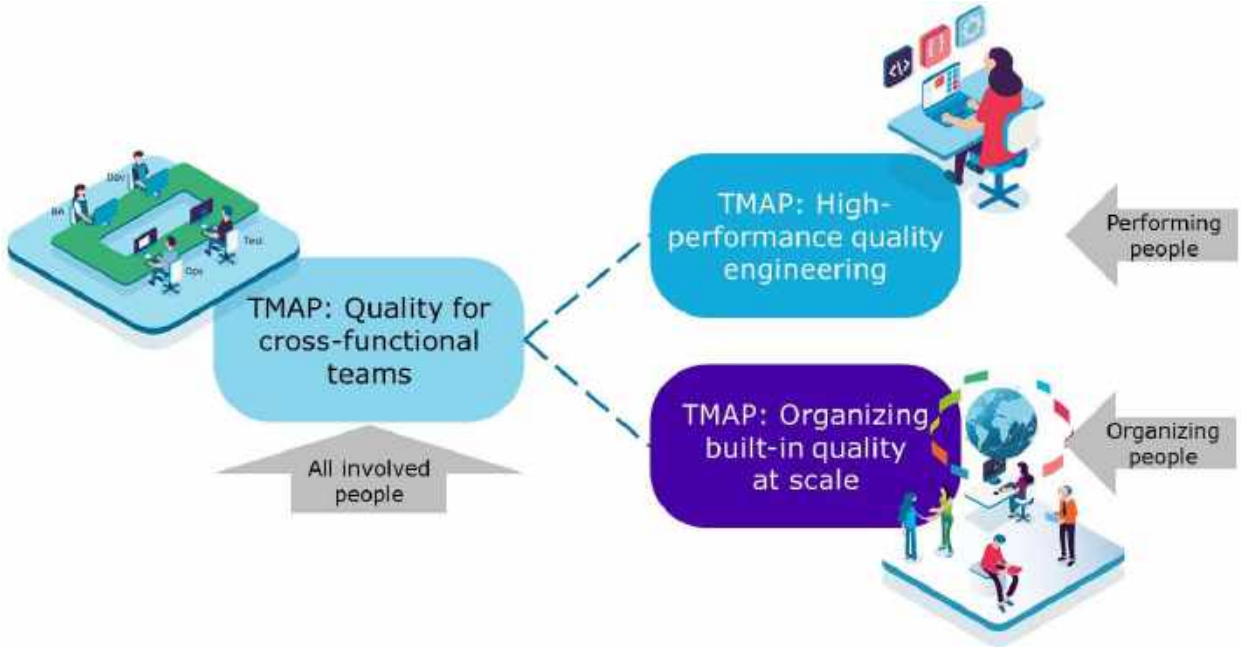
0. Introduction to this syllabus

0.1. TMAP®: Quality engineering certification scheme

In today's IT world cross-functional teams are expected to deliver business value with the right quality at speed. This requires high-performance IT delivery models such as DevOps and Scrum, which may be extended to a hybrid IT delivery model such as the SAFe® framework.

The TMAP® body of knowledge for quality engineering supports working towards built-in quality and takes the need for quality in products, process and people far beyond just testing.

The TMAP® certification scheme tailors to the needs of three target audiences. The figure below shows the certifications and indicates that the first certification "TMAP®: Quality for cross-functional teams" provides knowledge necessary for the other two certifications.



0.2. Purpose of this syllabus

The training course "**TMAP®: Quality for cross-functional teams**" is focused on all people that are working in, or are related to, a high-performance IT delivery team, such as in DevOps or Scrum. For example (but not restricted to) Business analysts, Systems architects, Developers, Programmers, Quality architects, Quality engineers, Test managers, Testers, Operations managers, Operations persons, Key-users, Business managers, Product owners, Scrum masters, Agile coaches and Agile quality coaches.

These people will acquire the required knowledge and skills that are important for building quality in their IT system and gathering information necessary to establish confidence that the pursued business value can be achieved.

This syllabus is the basis for the training course **"TMAP®: Quality for cross-functional teams"** and provides directions for the associated examination and certification. This training course is a 3-day training course, consisting of five sessions plus exam preparation and a 1-hour exam in the sixth session. Every session takes 3 hours (excluding breaks).

0.3. Brief introduction to the other TMAP certifications

There are two other certifications in the TMAP certification scheme:

Performing QA & testing activities in an organization requires a wide variety of knowledge and skills. The training course "**TMAP®: High-performance quality engineering**" enables professionals to perform these operational activities. It is a 3-day training course with a separate exam of 1.5 hours.

Organizing QA & testing requires orchestrating, arranging, planning, preparing and controlling the activities. The training course "**TMAP®: Organizing built-in quality at scale**" enables professionals that are responsible for organizing QA & testing to acquire necessary knowledge and skills to enable teams to achieve this. It is a 3-day training course with a separate exam of 1.5 hours.

0.4. Format of this training course and the syllabus

The 3-day training course consists of 5 sessions with a minimum of 3 hours (that is 15 contact hours in total), and session 6 with exam preparation and the actual exam (so the exam is included in the training course).

The number of hours mentioned above is excluding logistical preparation of the exam and breaks. Time for homework (such as self-study) is also excluded but for the average candidate homework should not take significant time.

The order of chapters and sections in this syllabus is according to the sequence of the training course, which gives a mix of theoretical and practical subjects. Every training session is a separate chapter in this syllabus and the sections each cover a learning objective.

0.5. Learning objectives and K-levels explained

Learning objectives (LOs) are brief statements that describe what a candidate is expected to know after studying each subject. The book "Quality for DevOps teams" contains all relevant information for the learning objectives, with each learning objective there is a reference to the relevant chapter(s) or section(s). Each learning objective has a corresponding cognitive level of knowledge (K-level). These K-levels, based on Bloom's modified taxonomy, are as follows:

- K1: Remember (knowledge). The candidate should remember or recognize a term or a concept.
- K2: Understand (comprehension). The candidate should select an explanation for a statement related to the question topic.
- K3: Apply (application). The candidate should select the correct application of a concept or technique and apply it to a given context.

An overview of the subjects of the learning objectives for this certification and their corresponding K-levels is given in section 0.7 and the details of the LOs are in chapters 1 through 6.

0.6. The TMAP®: Quality for cross-functional teams - exam

The format of the exam is multiple choice. There are 30 questions, 20 relate to K2 LOs, 10 relate to K3 LOs (K1 LOs are not explicitly examined). Each correctly answered question gives 1 point. To pass the exam, at least 66% of the points (that is 20 points) must be achieved.

The exams and certificates are provided by the independent exam provider iSQI.

More information and a sample exam can be found at: www.isqi.org.

0.7. Learning objectives and K-levels for this certification

Learning objectives in the order in which the subjects appear in the book Quality for DevOps teams.		K-level	Section	
			in this syllabus	in the book
The VOICE model				
LO01	The VOICE model of business delivery and IT delivery	K2	§ 1.2	§ 1.2.2, Ch 3, § 9.2
LO02	Indicators	K3	§ 1.3	§ 3.2; Ch 4; § 5.2.2; § 9.2.1; § 17.1; § 25.2.1.
IT delivery models				
LO03	IT delivery models - general	K2	§ 1.7	Ch 7; Ch 8; § 9.3; Ch 10 introduction; § 10.1
LO04	Scrum	K1 *	§ 6.1	§ 9.1
LO05	DevOps	K2	§ 1.8	§ 1.1, § 9.2 intro, § 9.2.1, § 9.2.2
LO06	SAFe®	K1 *	§ 6.2	§ 10.2
Continuous quality engineering				
LO07	Continuous quality engineering	K2	§ 2.3	§ 1.2, § 2.3, § 2.4, § 6.1, § 6.2, § 9.2.4, Ch 43 intro
LO08	Cross-functional teams	K3	§ 2.1	Ch 2 introduction; § 2.2 introduction, § 2.4, § 16.1
CI/CD pipelines and tooling				
LO09	CI/CD pipeline	K2	§ 3.1	§ 6.1; § 6.2; § 9.2.4
LO10	Capabilities	K3	§ 3.2	§ 6.1; § 6.2; § 6.3
QA & testing topics				
LO11	Introduction QA & testing topics	K2	§ 1.4	Ch 11; Ch 12; Ch 13
LO12	Total cost of quality	K1 *	§ 6.3	§ 15.2
LO13	Responsibilities and roles	K2	§ 2.2	Ch 16
LO14	Monitoring & control	K2	§ 5.1	Ch 17

Learning objectives in the order in which the subjects appear in the book Quality for DevOps teams.		K-level	Section	
			in this syllabus	in the book
LO15	Anomaly management	K2	§ 5.2	Ch 18
LO16	Reporting and alerting	K3	§ 5.3	Section 5.4, Ch 19
LO17	Continuous improvement	K3	§ 2.4	Ch 25
LO18	Quality risk analysis & test strategy (and link this to the voice model)	K2	§ 1.5	§ 5.2.1, § 5.2.2, Ch 26; Ch 35 introduction
LO19	Acceptance criteria	K2	§ 1.6	§ 5.6; Ch 27
LO20	Reviewing	K2	§ 4.4	Ch 29; § 35.6
LO21	Pull requests	K2	§ 4.5	§ 29.1.1.1
LO22	Test automation	K1	§ 3.5	Ch 32 introduction, § 32.1, § 32.2
LO23	Test execution	K2	§ 3.3	Ch 33
LO24	Investigate & assess outcome	K2	§ 3.4	Ch 34
Quality measures and skills				
LO25	Quality measures	K1	§ 4.1	Ch 28
LO26	Specification and Example	K2	§ 4.3	§ 35.2
LO27	Personal, interpersonal and team skills	K3	§ 4.2	Ch 36, § 36.1–36.6, §36.8, § 36.9
Test varieties				
LO28	Test varieties	K2	§ 2.6	Ch 37
LO29	Mutation testing tests the tests	K2	§ 4.9	Ch 42
Test design				
LO30	Test Design - Introduction (including basics of coverage-based and experience-based test design)	K2	§ 2.7	Ch 43; § 45.1
LO31	Test design entities relationships	K2	§ 3.6	Ch 44
Coverage-based testing				
LO32	Process-oriented test design overview	K1	§ 3.7	§ 45.2
LO33	Path testing (Process Cycle Test / Algorithm Test)	K3	§ 3.8	§ 46.3

Learning objectives in the order in which the subjects appear in the book Quality for DevOps teams.		K-level	Section	
			in this syllabus	in the book
LO34	Code coverage	K1	§ 4.8	§ 46.8
LO35	Condition-oriented test design overview	K1	§ 5.4	§ 45.3, § 46.4 introduction
LO36	Condition Coverage (CC), Decision Coverage (DC) & Condition Decision Coverage (CDC)	K2	§ 5.5	§ 46.4.2, § 46.4.3
LO37	Modified Condition Decision Coverage (MCDC)	K1	§ 5.6	§ 46.4.2 § 46.4.4
LO38	Multiple Condition Coverage (MCC)	K3	§ 5.7	§ 46.4.2 § 46.4.5
LO39	Decision Table Testing	K3	§ 5.8	§ 46.4.5, template
LO40	Data-oriented test design overview	K1	§ 2.8	§ 45.4
LO41	Equivalence partitioning	K3	§ 2.9	§ 46.5
LO42	Boundary Value Analysis	K3	§ 2.10	§ 46.5
LO43	Data Combination Test (including EP, BVA and Pairwise)	K1 *	§ 6.4	§ 46.6
LO44	Appearance-oriented test design overview	K1	§ 4.6	§ 45.5
LO45	Syntactic Test	K2	§ 4.7	§ 46.7
Experience-based testing				
LO46	Experience-based testing overview	K1	§ 4.10	§ 43.4, § 47.1
LO47	Exploratory testing and mob testing	K3	§ 4.11	§ 36.1, § 47.4, template charter
LO48	Value of unstructured testing	K1 *	§ 6.5	Ch 48
Quality characteristics				
LO49	Quality characteristics	K1	§ 2.5	Appendix
Terminology				
LO50	Quality	K1	§ 1.1	Ch5 introduction
LO51	Testing and the terms contained in its definition	K1	§ 1.9	Ch5 introduction; § 5.2, § 5.3
LO52	Terms relevant to quality and testing	K1	§ 1.10	§ 5.5, §18.3
LO53	End-to-end testing within a team or across teams	K1 *	§ 6.6	§ 14.3.2; §, 32.4.3; § 33.2; § 37.3; § 37.4

Note: Learning objectives marked with K1 * are relevant but will not explicitly be part of the exam.

0.8. Prerequisites for candidates

The candidates are expected to have basic IT knowledge and experience. Also they must be familiar with the Agile manifesto. There is no required previous certification.

0.9. Accreditation of training providers

Training providers and trainers that want to prepare candidates for the exam will need to acquire accreditation from iSQI. For more information please contact TMAP2020@iSQI.org

0.10. Literature

Exam literature:
- The book "Quality for DevOps teams" (ISBN 978-90-75414-89-9) is for sale at www.ict-books.com and other bookstores.
- TMAP glossary: https://www.tmap.net/page/tmap-glossary-online.
- Exploratory testing charter explanation and template on www.tmap.net
- Decision Table template on www.tmap.net
- Path testing template on www.tmap.net

Additional literature:
- The TMAP body of knowledge website – www.tmap.net

Additional literature (specifically for trainers to acquire more in-depth knowledge):
- The Agile Manifesto – www.agilemanifesto.org
- The Scrum Guide – www.scrumguides.org
- The SAFe website – www.scaledagileframework.com
- ISO25010 - www.iso.org/standard/35733.html
- Also please refer to the references in the book "Quality for DevOps teams".

0.11. Acknowledgements

This syllabus was created by a diverse team. We would like to thank the following people (in no particular order) for their contributions in writing and reviewing this document:

Sogeti people: Rik Marselis, Berend van Veenendaal, Dennis Geurts, Wouter Ruigrok, Ralph Klomp, Bert Linker, Eveline Moolenaars, Annemiek van den Heuvel, Marc Roekens, Joost Coenen, Rob Vijverberg, Tinus Vellekoop, Stefan Gerstner, Charlotte Janus, Irma Hagemans, Robin Klein, Serife Ciftci, Bruno Lepretre, Jürgen Benierman.

iSQI people: Stephan Goericke, Erika Paasche, Corinna Flemming - Vogt, Annaleida van de Meent - Schepers.

TMAP Special Interest Group members: Leo van der Aalst, Guido Dulos, Gilbert Smulders, Freddy de Weerd, Erik Runhaar, Benjamin Timmermans, Nicolaï Roos.

1. Session 1

Learning objectives

LO01, LO02, LO03, LO05, LO11, LO18, LO19, LO50, LO51, LO52.

Key words (please note: keywords are only mentioned in the first session where they appear)

Acceptance criteria; agile mindset; anomaly; business value; capabilities; completion criteria; confidence; defect; definition of done (DoD); definition of ready (DoR); DevOps; DevOps activities; dynamic testing; end-to-end test; error; exploration; failure; fault; high-performance IT delivery; hybrid IT delivery; incident; indicator; IT delivery model; objectives; pre-test; problem; quality; quality at speed; quality engineering; quality risk; SAFe® (Scaled Agile Framework); Scrum; sequential IT delivery; static testing; test automation; test strategy; test variety; testing; validation; value; verification; VOICE model.

1.1. Quality (LO50; K1)

Quality is the totality of features and characteristics of a product or service that bear on its ability to satisfy stated or implied needs.

A quality risk is a specific chance that the product fails in relation to the expected impact if this occurs. The chance of failure is determined by the chance of faults and the frequency of use. The impact is related to the operational use of the product.

The candidate has knowledge of these terms and their meaning.

Book: chapter 5 introduction; section 5.2.1.

1.2. The VOICE model of business delivery and IT delivery (LO01; K2)

High-performance IT delivery teams (such as in Scrum and DevOps) use the VOICE model as a foundation to structure and organize their work.

The candidate can plot the elements of the VOICE model on the DevOps activities.

The candidate can give a description of the VOICE model and knows that it's an acronym of Value, Objectives, Indicators, Confidence and Experience.

Book: section 1.2, chapter 3 and section 9.2.

1.3. Indicators (LO02; K3)

To measure whether the objectives (from the VOICE model) are achieved, one or more indicators per objective are defined. These indicators are measured by means of data collection and data analysis. Measuring is generally done by testing but other quality measuring activities are also used. These indicators can be divided in four groups, Business value related indicators, IT delivery related indicators, Team related indicators and Problem related indicators. For monitoring & control also functional and non-functional indicators are used as well as quality and team performance indicators.

The candidate is able to apply the indicators in the VOICE model as the starting point for determining the needed quality engineering activities, and other quality measuring activities.

Book: section 3.2; chapter 4; section 5.2.2; section 9.2.1; section 17.1; section 25.2.1.

1.4. Introduction QA & testing topics (LO11; K2)

Every IT delivery model, framework, mindset, organization etc. has its own development approach, workflow, phases, roles, work products and/or activities. We defined a set of generic QA & testing activities – the so-called "topics" –, which are applicable – in one way or another – to all these different development approaches.

The candidate understands that there are two overarching groups: Organizing topics and Performing topics.

The candidate is also able to describe both groups of topics and recognize which topic belongs to which group.

Book: chapter 11, chapter 12, chapter 13.

1.5. Quality Risk Analysis & Test Strategy (LO18; K2)

A test strategy is the allocation of quality measures to balance the investment in testing and to make an optimal distribution of effort over test varieties and test approaches to give insight in test coverage and test intensity. Often this is based on the quality risk levels and the pursued business value.

The candidate recalls that for teams to determine where to focus their QA & testing activities they need to investigate the quality risks involved with the IT system they are creating or changing. Therefore, the candidate knows what a quality risk is as well as what a test strategy is and what the relation between the two is. The candidate also understands that based on the quality risks the test intensity is determined which is reflected in the test strategy. The candidate understands that quality risks are part of the indicators that will be measured by testing.

Book: section 5.2.1, 5.2.2; chapter 26, chapter 35 introduction.

1.6. Acceptance criteria (LO19; K2)

A cross-functional team, which is common in DevOps, will agree to deliver an IT product with a specific quality level. This quality level is defined by the acceptance criteria. The team, the product owner and other stakeholders discuss and collaborate closely so that the acceptance criteria are supported by everyone involved.

The candidate demonstrates an understanding of what acceptance criteria are and how acceptance criteria can be obtained and defined. The candidate is also able to compare acceptance criteria with other relevant criteria such as definition of ready (DoR), definition of done (DoD), exit criteria and completion criteria.

Book: section 5.6; chapter 27.

1.7. IT delivery models (LO03; K2)

An IT delivery model is a conceptual framework which supports a software development process and describes all assets and competencies.

The candidate is able to compare the three groups of IT delivery models: Sequential IT delivery, High-performance IT delivery and Hybrid IT delivery.

The candidate also understands the main parts of the IT delivery models in which quality engineering activities are performed (e.g. Design, Code, Test and Deploy).

Book: chapter 7; chapter 8; section 9.3; chapter 10 introduction; section 10.1.

1.8. DevOps (LO05; K2)

DevOps is a cross-functional systems engineering culture that aims at unifying systems development (Dev) and systems operations (Ops) with the ability to create and deliver fast, cheap, flexible and with adequate quality, whereby the team as a whole is responsible for the quality.

The candidate can state the main ideas behind DevOps. Also, the candidate can give descriptions of the six DevOps activities: Monitor, Plan, Code, Integrate, Deploy and Operate. These activities provide support to explain the relation of DevOps activities with the QA and testing topics.

The prerequisites that are often mentioned together with (implementing) DevOps are understood by the candidate.

Book: sections 1.1, 9.2 introduction, 9.2.1 and 9.2.2.

1.9. Testing and the terms contained in its definition (LO51, K1)

Testing consists of verification, validation and exploration activities that provide information about the quality and the related risks, to establish the level of confidence that a test object will be able to deliver the pursued business value.

The candidate has knowledge of these terms, their underlying terms, and their meaning.

Book: chapter 5 introduction; section 5.2; section 5.3.

1.10. Terms relevant to quality and testing (LO52; K1)

Static testing, Dynamic testing, Error, Fault, Failure, Incident, Problem, Anomaly and Defect.

The candidate has knowledge of these terms and their meaning.

Book: section 5.5, section 18.3.

2. Session 2

Learning objectives

LO07, LO08, LO13, LO17, LO28, LO30, LO40, LO41, LO42, LO49.

Key words

Boundary value analysis; Continuous everything; continuous improvement; continuous quality engineering; coverage group; coverage-based test design; cross-functional team; data-oriented test design; end-to-end testing; equivalence partitioning; experience-based testing; functional testing; non-functional testing; organizing topics; performing topics; progression testing; QA & testing topics; quality characteristic; regression testing; responsibilities; roles; spheres of testing; test case; test design entities relationship diagram; test design technique; test plan; test scenario; test script; test situation; test varieties; testing pyramid; testing quadrants.

2.1. Cross-functional teams (LO08; K3)

Working in a cross-functional team means that the team as a whole is responsible for delivering value. The team has all competencies and skills to perform the necessary tasks and no team member has the monopoly on performing any task. This way the team can always go forward, even when a team member is temporarily not available. And of course, a team can work together with specialists from other teams or support groups for specific tasks. A person can have multiple roles sequentially or even in parallel. It is not common for people to have a specific function, since that would easily lead to monopolies on certain tasks.

The candidate can explain how a cross-functional team operates and can state in which way a cross-functional team operates more effectively than a multi-disciplinary team or when working in silos.

Book: chapter 2 introduction; section 2.2 introduction, section 2.4, section 16.1.

2.2. Responsibilities and roles (LO13; K2)

In DevOps, people work together closely, and the team has the required people to make the project successful. Working in cross-functional DevOps teams also means that all team members are prepared to take on any of the roles if necessary.

The candidate is able to link common responsibilities and QA & Testing responsibilities with roles.

Book: chapter 16.

2.3. Continuous quality engineering (LO07; K2)

In the DevOps IT delivery model, there is continuous focus on quality engineering. Commonly DevOps teams try to implement "continuous everything", which means that they strive to automate as many tasks and activities as possible. This needs Continuous Integration, Continuous Delivery, Continuous Deployment, Continuous Monitoring and Continuous Quality and Testing.

The candidate understands why continuous quality engineering is important and what the “continuous” terms mean.

Book: section 1.2, section 2.3, section 2.4, section 6.1; section 6.2, section 9.2.4, chapter 43 intro.

2.4. Continuous improvement (LO17; K3)

DevOps teams work in an everchanging world where the common expectation is that quality and speed improve. They constantly need to improve their way of working and adapt to changed circumstances.

The candidate can describe how to establish a continuous improvement culture and can select good examples of how to continuously improve the product, process and people.

Book: chapter 25.

2.5. Quality characteristics and non-functional testing (LO49; K1)

When deciding on their test varieties many testers start with distinguishing between functional testing and non-functional testing. This refers to the quality characteristics. These are a very useful tool to identify various characteristics of quality that are important for the stakeholders of an IT-system.

The candidate recognizes the eight main quality characteristics for product quality and the five main quality characteristics for quality in use. The candidate also remembers that other characteristics may be needed for specific products such as systems based on Artificial Intelligence.

Book: Appendix.

2.6. Test varieties (LO28; K2)

IT products are different. People are different. Projects are different. Environments are different. So, it would be an illusion to think that one-size-fits-all exists for testing. You need variety in your testing.

The candidate understands how the spheres of testing, the testing pyramid and the testing quadrants support in determining what varieties in testing are needed to address all necessary aspects and scopes of testing. The candidate also understands the ideas behind regression testing and progression testing, and the importance of agreeing on a test strategy.

Book: chapter 37.

2.7. Test design – Introduction (LO30; K2)

Creating tests and executing them may sound easy. But structured testing requires careful consideration. We use the term "test design" for the complex whole of these activities, even though in some approaches to testing there is no actual up-front design involved.

The candidate can distinguish the two ways of creating and executing tests: coverage-based and experience-based test design. The candidate understands why these should always be combined. The candidate understands the basics of test design and the four coverage groups of coverage-based test design techniques.

Book: chapter 43; section 45.1.

2.8. Data-oriented test design overview (LO40; K1)

The data-oriented coverage group contains test design techniques that use the structure or behavior of the data that is used in the IT system.

The candidate recognizes test design techniques that belong to data-oriented test design.

Book: section 45.4.

2.9. Equivalence partitioning (LO41; K3)

In the application of equivalence classes, the entire value range of a parameter is partitioned into classes. In a specific class the system behavior is similar (equivalent) for every value of the parameter.

The candidate can apply Equivalence Partitioning (EP) to a given test basis.

Book: section 46.5.

2.10. Boundary Value Analysis (LO42; K3)

Boundary Value Analysis is a test design technique based on the fact that around a boundary in the value range of a variable there's a higher risk of faults in a system.

The candidate understands the difference between two-value -, three-value – and four-value Boundary Value Analysis. The candidate can apply Boundary Value Analysis (BVA) to a given test basis.

Book: section 46.5.

3. Session 3

Learning objectives

LO09, LO10, LO22, LO23, LO24, LO31, LO32, LO33.

Key words

Algorithm test; assess outcome; capabilities; CI/CD pipeline; continuous delivery; continuous deployment; continuous integration; explicit testing; implicit testing; pairwise testing; path testing; process cycle test; process-oriented test design; test execution; total cost of quality.

3.1. CI/CD pipeline (LO09; K2)

In DevOps, a CI/CD pipeline needs to be implemented. Continuous Integration & Continuous Deployment (CI/CD) is seen as the backbone to enable DevOps. It bridges, maybe even closes, the gap between development and operations by automating the building, packaging, testing, provisioning of infrastructure and deployment of applications.

The candidate understands the stages in a pipeline and the different scopes for team test and business test in the pipeline.

Book: section 6.1; section 6.2; section 9.2.4.

3.2. Capabilities (LO10; K3)

With a CI/CD pipeline, steps in the software delivery process are automated. When creating such a – fully – automated CI/CD pipeline, tools with specific capabilities are needed. Tools can frequently change, therefore the capabilities need to be well defined to have a stable pipeline.

The candidate is able to connect capabilities with the continuous activities and pipeline stages.

Book: section 6.1, section 6.2; section 6.3.

3.3. Test execution (LO23; K2)

Test execution is the execution of tests by running the system under test and thus obtaining the actual results that can be compared with the expected results to determine whether the tests have passed or failed.

The candidate understands the difference between explicit and implicit testing and that the different test varieties have a different focus. Furthermore, the candidate can describe what a pre-test is.

Book: chapter 33.

3.4. Investigate & assess outcome (LO24; K2)

When the team members execute the test scenarios and test scripts, they compare the actual outcomes with the expected outcomes and assess the results.

The candidate can state the main ideas behind investigating & assessing the outcome of tests.

Book: chapter 34.

3.5. Test automation (LO22; K1)

The demand for continuous testing has created a renewed focus on test automation. Test automation is one of the main opportunities to meet the need for quality at speed, but also requires a structured approach in order to effectively realize such a vision.

The candidate recalls that the testing quadrants and the testing pyramid can be used to determine what to test manually and what to test with automated testing tools.

The candidate also knows that DevOps usually coincides with continuous delivery and that therefore most of the testing should be performed automatically during the process.

Book: chapter 32 introduction, section 32.1, section 32.2.

3.6. Test design entities relationships (LO31; K2)

In coverage-based test design we use a number of different terms for specific entities in the test design: test situation, logical test case, physical test case and test scenario.

The candidate understands these entities and can explain the relationships between these entities.

Book: chapter 44.

3.7. Process-oriented test design overview (LO32; K1)

The process-oriented coverage group contains test design techniques that are based on processes, for example a business process.

The candidate recognizes test design techniques that belong to process-oriented test design.

Book: section 45.2.

3.8. Path testing (Process Cycle Test / Algorithm Test) (LO33; K3)

Path testing aims to demonstrate that all combinations of N consecutive paths in a process or program flow are covered. A path in this context consists of all steps between a decision point and the next decision point, or between the start and the first decision point, or between the last decision point and the end.

The candidate can apply the coverage type “path coverage” and the test design techniques “process cycle test” and “algorithm test” to a given test basis.

Book: section 46.3, template path testing on www.tmap.net .

4. Session 4

Learning objectives

LO20, LO21, LO25, LO26, LO27, LO29, LO34, LO44, LO45, LO46, LO47.

Key words

Appearance-oriented test design; checklist; code coverage; collaboration techniques; corrective; detective; error guessing; experience-based testing; exploratory testing; exploring; formal reviewing; four amigos; informal reviewing; INVEST; mob testing; mutation testing; pairing; preventive; pull request; quality measures; Specification and Example; static analysis; syntactic testing; team values; test coverage; three amigos; T-shaped, pi-shaped, comb-shaped.

4.1. Quality measures (LO25; K1)

Quality was, is and remains a challenge within the IT industry. Quality engineering consists of a great number of possible activities, the so-called quality measures.

The candidate remembers that all quality measures may relate to all DevOps activities and that there are three groups of quality measures: preventive, detective and corrective.

Book: chapter 28.

4.2. Personal, interpersonal and team skills (LO27; K3)

People have a wide variety of skills. To be effective in a high-performance team, people need to be cross-functional, which means that the people in the team need to understand and perform all the tasks of the team. This doesn't mean that each person in the team needs to be an expert on all topics. It does mean that the team should not fail when one team member is temporarily unavailable.

The candidate understands the importance of psychological safety to be effective in a cross-functional team. The candidate is able to apply collaboration techniques and can explain how team values and unfavorable team behavior impact the team's performance. The candidate understands the concepts learn fast, exploring, support from staff organization and T-shaped-and-beyond.

Book: chapter 36.

4.3. Specification and Example (LO26; K2)

In order to achieve a shared common understanding of what "it" is that should be built and try to build "it" right the first time, you can use Specification and Example mapping approaches.

The candidate can describe the ideas behind Specification and Example and understands that it supports common understanding of stories/features and exploring ideas. The candidate understands that the four amigos approach can very well contribute to reaching this common understanding.

Book: section 35.2.

4.4. Reviewing (LO20; K2)

Static testing consists of informal reviewing, formal reviewing and static analysis.

The candidate understands which of these groups is generally applied by high-performance teams and with what purpose. Furthermore, the candidate understands INVEST.

Book: chapter 29, section 35.6.

4.5. Pull requests (LO21; K2)

When using a check-out / check-in mechanism for code, as is common in continuous integration pipelines, a pull request is part of the check-in process.

The candidate understands the concept of pull requests as an informal review technique as well as a method for collaboration within the team.

Book: section 29.1.1.1.

4.6. Appearance-oriented test design overview (LO44; K1)

The appearance-oriented coverage group contains test design techniques that relate to the appearance of an IT system, that is how the system presents itself to the users or to other systems.

The candidate recognizes test design techniques that belong to appearance-oriented test design.

Book: section 45.5.

4.7. Syntactic Test (LO45; K2)

The candidate understands that syntactic testing is used to test the validity of input and output data and also to test other attributes of the user interface. The candidate also understands where the relevant rules may be found.

Book: section 46.7.

4.8. Code coverage (LO34; K1)

Code coverage can be measured by specific tools during the execution of tests.

The candidate can recollect the different code coverage types and whether these should be preferred or not.

Book: section 46.8.

4.9. Mutation testing tests the tests (LO29; K2)

Can the products of testing also be tested? Certainly! And they should be tested!

The candidate understands how mutation testing verifies the quality of the test set.

Book: chapter 42.

4.10. Experience-based testing overview (LO46; K1)

Experience-based testing is a group of test approaches that are based on the skills, intuition and experience of the tester. These approaches leave the tester free to design test cases in advance or to create them on the spot during the test execution, mostly testers will do both.

The candidate recognizes approaches that belong to experience-based testing and knows that some level of combination of experience-based and coverage-based testing should be in the test strategy.

Book: section 43.4, section 47.1.

4.11. Exploratory testing and mob testing (LO47; K3)

Exploratory testing is the most versatile of the described approaches of experience-based testing.

The candidate understands the characteristics of exploratory testing and is able to prepare an exploratory testing charter. The candidate understands the importance of a test log and debriefing.

When exploratory testing is done in larger groups this is generally referred to as "mob testing".

The candidate is able to create and execute a charter for a mob testing session and report results.

Book: section 36.1; section 47.4, template Exploratory testing charter on www.TMAP.net.

5. Session 5

Learning objectives

LO14, LO15, LO16, LO35, LO36, LO37, LO38, LO39.

Key words

Alerting; condition-oriented testing; condition coverage (CC); condition decision coverage (CDC); control; dashboarding; decision coverage (DC); decision table test; modified condition decision coverage (MCDC); monitoring; multiple condition coverage (MCC); reporting.

5.1. Monitoring & control (LO14; K2)

Monitoring and control are intended to promptly identify, report and forecast (gaps in) expected and actual quality, related to the pursued business value.

The candidate understands what monitoring & control involves.

Book: chapter 17. Note: The subject 'Indicators' is part of section 1.3 of this syllabus.

5.2. Anomaly management (LO15; K2)

An anomaly is a difference between the expected behavior and the actual outcome of a test. This is registered so that the cause can be analyzed and resolved.

The candidate knows the terms error, fault, failure, incident and problem, and understands how these relate to anomalies and how the term defect can cause confusion thus should be avoided. Furthermore, the candidate understands the light-weight process for handling anomalies.

Book: chapter 18.

5.3. Reporting & alerting (LO16; K3)

Testing is about providing different levels of information. Usually there are multiple audiences for the information that the team generates based on their quality engineering activities.

DevOps teams and their stakeholders want to, and need to, have constant and direct insight into the status of the IT system. And if something (either in product or process) deviates from the expectations, they must be alerted as soon as possible. Therefore, DevOps teams will use state-of-the art tools for reporting and alerting, where on-line real-time dashboards are today perceived as need-to-haves.

The candidate can select relevant information for dashboards & reports.
The candidate is able to analyze and draw conclusions from overview reports.
The candidate can select a proper way of alerting stakeholders.

Book: section 5.4; chapter 19.

5.4. Condition-oriented overview (LO35; K1)

The condition-oriented coverage group contains test design techniques that are based on the behavior of decision points and the conditions that determine the result of a decision.

The candidate recognizes test design techniques and coverage types that belong to condition-oriented test design.

Book: section 45.3, section 46.4 introduction.

5.5. Condition Coverage, Decision Coverage & Condition Decision Coverage (LO36; K2)

CDC is a coverage type, from the coverage group Condition, that ensures the possible outcomes of each condition and of the decision are tested at least once. This implies both "condition coverage" and "decision coverage".

The candidate understands Condition Coverage (CC), Decision Coverage (DC) and Condition Decision Coverage (CDC) and why CDC is preferred.

Book: sections 46.4.2 and 46.4.3.

5.6. Modified Condition Decision Coverage (LO37; K1)

MCDC is a coverage type, from the coverage group Condition, that ensures that every possible outcome of a condition is the determinant of the outcome of the decision, at least once. MCDC implies also "condition/decision coverage".

The candidate recognizes the concepts behind Modified Condition Decision Coverage (MCDC).

Book: sections 46.4.2 and 46.4.4.

5.7. Multiple Condition Coverage (LO38; K3)

MCC is a coverage type ensures that all possible combinations of outcomes of conditions in a decision are tested at least once.

The candidate can apply Multiple Condition Coverage (MCC) to a given test basis in combination with Decision Table Testing.

Book: sections 46.4.2 and 46.4.5.

5.8. Decision Table Testing (LO39; K3)

A decision table defines all possible combinations of the individual conditions.

The candidate can apply the Decision Table test design technique to a given test basis in combination with Multiple Condition Coverage.

Book: section 46.4.5, template Decision Table Testing on www.TMAP.net.

6. Appendix

This appendix contains learning objectives that are relevant for members of cross-functional teams. These subjects are not part of the exam but will contribute to the relevant knowledge of the candidate. Training providers are advised to present information about these learning objectives in an additional way, which may be by using extra available time or by supplying information to be read by the candidate individually.

Learning objectives
LO04, LO06, LO12, LO43, LO48, LO53.

6.1. Scrum (LO04; K1)

The candidate knows that scrum is a framework that people use to address and solve complex problems in an adaptive manner, while delivering the highest value products in a rewarding and creative way.

The candidate recalls the elements which scrum consists of.

Book: section 9.1.

6.2. SAFe® (LO06; K1)

The Scaled Agile Framework (SAFe®) is a structured hybrid IT delivery approach that helps large enterprises implement Agile at large scale.

The candidate has knowledge of the (full) SAFe model characteristics, especially the four layers (Team, Program, Large solution and Portfolio), and shared services and system teams.

Book: section 10.2.

6.3. Total cost of quality (LO12; K1)

When there is little attention to quality, there will be many failures, which cause a huge cost for fixing and damage caused. When there is too much effort in quality assurance, there will be hardly any failures, but the costs are so high that the businesspeople will protest.

The candidate realizes that it is necessary to find the right balance for QA & testing efforts to reach the optimum total cost of quality.

Book: section 15.2.

6.4. Data Combination Test (LO43; K1)

The Data Combination Test tests combinations of values of data items.

The candidate recognizes the basic concepts of the Data Combination Test (DCoT) test design technique and knows that different coverage levels can be achieved.

Book: section 46.6.

6.5. Value of unstructured testing (LO48; K1)

Any testing lacking a plan containing what to do and what to expect of a system or lacking preparation of the test is unstructured.

The candidate recognizes the basic reasons for unstructured testing and why in most situations unstructured testing should be avoided. The candidate knows that experience-based testing can be organized in a structured way, so experience-based testing is not the same as unstructured testing.

Book: chapter 48.

6.6. End-to-end testing within a team or across teams (LO53; K1)

DevOps teams should be properly equipped to perform end-to-end tests from a business process perspective or be supported by specialized people for tasks that the team does not have the knowledge and/or capacity for or which are at another organizational level (an example is end-to-end-regression-tests-on-demand by a system team).

The candidate knows that the concept of end-to-end (regression) testing is important.

Book: section 14.3.2, section, 32.4.3, section 33.2, section 37.3, section37.4.

You can contact the Sogeti Academy in the Netherlands at academy.nl@sogeti.nl.

About Sogeti

Part of the Capgemini Group, Sogeti operates in more than 100 locations globally. Working closely with clients and partners to take full advantage of the opportunities of technology, Sogeti combines agility and speed of implementation to tailor innovative future-focused solutions in Digital Assurance and Testing, Cloud and Cybersecurity, all fueled by AI and automation. With its hands-on 'value in the making' approach and passion for technology, Sogeti helps organizations implement their digital journeys at speed.

A global leader in consulting, technology services and digital transformation, Capgemini is at the forefront of innovation to address the entire breadth of clients' opportunities in the evolving world of cloud, digital and platforms. Building on its strong 50-year heritage and deep industry-specific expertise, Capgemini enables organizations to realize their business ambitions through an array of services from strategy to operations. Capgemini is driven by the conviction that the business value of technology comes from and through people. It is a multicultural company of almost 220,000 team members in more than 40 countries. The Group reported 2019 global revenues of EUR 14.1 billion.

Visit us at www.sogeti.com