

# **Standaard verklarende woordenlijst van testtermen**

## **Vertaling Engels - Nederlands**

**Versie:** 0.2 (dd. 21 mei 2006)  
**Status:** Concept  
**Gemaakt door:** WP-Glossary van Belgium & Dutch Testing Qualifications Board  
**Editor:** Meile Posthuma (Nederland)  
**Verklaring  
betreffende  
auteursrecht:** Dit document kan in al zijn onderdelen, of gedeeltelijk worden  
gekopieerd, indien de bron wordt erkend.  
**Orginele versie bron:** 1.2

**Contributors original glossary:**

Rex Black (USA)  
Sigrid Eldh (Sweden)  
Isabel Evans (UK)  
Dorothy Graham (UK)  
Julian Harty (UK)  
David Hayman (UK)  
Juha Itkonen (Finland)  
Vipul Kocher (India)  
Fernando Lamas de Oliveira (Portugal)  
Tilo Linz (Germany)  
Peter Morgan (UK)  
Thomas Müller (Switseland)  
Avi Ofer (Israel)  
Dale Perry (USA)  
Horst Pohlmann (Germany)  
Meile Posthuma (The Netherlands)  
Erkki Pöyhönen (Finland)  
Maaret Pyhäjärvi (Finland)  
Andy Redwood (UK)  
Stuart Reid (UK)  
Piet de Roo (The Netherlands)  
Shane Sauders (UK)  
Steve Sampson (UK)  
Shane Sanders (UK)  
Hans Schaefer (Norway)  
Jurriën Seubers (The Netherlands)  
Dave Sherratt (UK)  
Mike Smith (UK)  
Andreas Spillner (Germany)  
Richard Taylor (UK)  
Geoff Thompson (UK)  
Stephanie Ulrich (Germany)  
Matti Vuori (Finland)  
Gearrel Welvaart (The Netherlands)  
Pete Williams (UK)

**Vertaald en getoetst door:**

Paul Beekman  
Alain Bultink  
Erwin Engelsma  
Ralph Eyckerman  
Jurian van de Laar  
Anke van der Moer  
Iris Pinkster  
Meile Posthuma  
Ewald Roodenrijs  
Marjolein Steyerberg  
Erik van Veenendaal

## Inhoudsopgave:

Standaard verklarende woordenlijst van testtermen Vertaling Engels - Nederlands .....	1
Voorwoord .....	4
1. Introductie .....	4
2. Reikwijdte .....	4
3. Rangschikking .....	4
4. Normatieve referenties .....	4
5. Definities .....	6
A .....	6
B .....	7
C .....	9
D .....	12
E .....	15
F .....	16
G .....	17
H .....	17
I .....	18
K .....	20
L .....	20
M .....	21
N .....	22
O .....	23
P .....	23
Q .....	25
R .....	25
S .....	27
T .....	30
U .....	36
V .....	36
W .....	36
Bijlage A (Informatief).....	38
Bijlage B (Methode om commentaar op deze woordenlijst aan te leveren) .....	39

## Voorwoord

Bij het maken van deze verklarende woordenlijst heeft de werkgroep naar de meningen en de commentaren van een zo breed mogelijk spectrum in de industrie, handel en overheidsinstellingen en organisaties gestreefd, met het doel een internationale teststandaard neer te zetten die zo breed mogelijk gedragen wordt. Complete overeenstemming zal zelden, indien ooit, bereikt worden bij het maken van een document van deze aard.

Testgemeenschappen die aan deze woordenlijst hebben bijgedragen zijn: Australië, België, Finland, Duitsland, India, Israël, Nederland, Noorwegen, Portugal, Zweden, het Verenigd Koninkrijk, en de V.S.. Vele (software) testers hebben de BS 7925-1 sinds zijn oorspronkelijke publicatie in 1998 gebruikt. Het heeft ook als belangrijke referentie gediend voor de Information Systems Examination Board (ISEB) op zowel Foundation- als Practitionerniveau gediend. De standaard werd aanvankelijk ontwikkeld met de nadruk op componenttesten, maar sinds zijn publicatie, zijn vele commentaren en voorstellen voor nieuwe definities ingediend om de standaard te verbeteren en uit te breiden om een bredere dekking van het software testen te bieden. In deze nieuwe versie van de verklarende testwoordenlijst zijn veel van deze voorgestelde aanvullingen opgenomen. Het zal als referentiedocument door de International Software Testing Qualification Board (ISTQB) worden gebruikt als certificeringsprogramma.

## 1. Introductie

Veel tijd en inspanning worden verspild zowel binnen als tussen de industrie, handel, overheid en professionele en academische instellingen wanneer de ambiguïteiten zich voordoen. Als resultaat van het onvermogen om voldoende onderscheid te maken tussen termen als 'coderegeldkking' en 'besluitdekking'; 'test set', 'testspecificatie' en 'testplan' en soortgelijke termen die een interface tussen diverse sectoren van de maatschappij vormen. Voorts is het professionele- of technische gebruik van deze termen vaak in tegenspraak met de verschillende betekenissen die aan hen worden toegeschreven.

## 2. Reikwijdte

Dit document vertegenwoordigt concepten, termen en definities die dienen als hulp voor communicatie in (software) testen en verwante disciplines.

## 3. Rangschikking

De verklarende woordenlijst is gerangschikt in één enkele sectie op alfabetische volgorde. Sommige termen hebben aan andere synoniem de voorkeur gekregen waarbij, de definitie van de voorkeursterm verschijnt, met een verwijzing naar de synonieme term. Bijvoorbeeld: gestructureerd testen verwijst naar het white box testen. Voor het synoniem, wordt de "Zie" indicator gebruikt. "Zie ook" verwijzingen worden ook gebruikt. Zij helpen de gebruiker om snel naar de juiste indexterm te navigeren. "Zie ook" verwijzingen worden gebruikt voor relaties wanneer een term met een bredere betekenis naar een term met een engere betekenis verwijst en voor overlappende betekenis tussen twee termen.

## 4. Normatieve referenties

De aangegeven editie was geldig ten tijde van publicatie. Alle standaarden kunnen een revisie hebben ondergaan, en partijen bij overeenkomsten die op deze Standaard zijn gebaseerd

worden aangemoedigd om de mogelijkheid te onderzoeken om de meest recente editie van de standaards die beneden staan vermeld toe te passen.

Leden van de IEC en ISO onderhouden registers van de huidig geldige Internationale Standaards.

1. BS 7925-2:1998. Software Component Testing.
2. DO-178B:1992. Software Considerations in Airborne Systems and Equipment
3. Certification, Requirements and Technical Concepts for Aviation (RTCA SC167).
4. IEEE 610.12:1990. Standard Glossary of Software Engineering Terminology.
5. IEEE 829:1998. Standard for Software Test Documentation.
6. IEEE 1008:1993. Standard for Software Unit Testing.
7. IEEE 1012:1986. Standard for Verification and Validation Plans
8. IEEE 1028:1997. Standard for Software Reviews and Audits.
9. IEEE 1044:1993. Standard Classification for Software Anomalies.
10. IEEE 1219:1998. Software Maintenance.
11. ISO/IEC 2382-1:1993. Data processing - Vocabulary - Part 1: Fundamental terms.
12. ISO 9000:2000. Quality Management Systems – Fundamentals and Vocabulary.
13. ISO/IEC 9126-1:2001. Software Engineering – Software Product Quality – Part 1:
14. Quality characteristics and sub-characteristics.
15. ISO/IEC 12207:1995. Information Technology – Software Life Cycle Processes.
16. ISO/IEC 14598-1:1996. Information Technology – Software Product Evaluation - Part 1: General Overview.

## 5. Definities

### A

- |     |                        |  |  |
|-----|------------------------|--|--|
| 1.  | Abstract test case:    | <i>Logisch testgeval:</i>              | <b>Zie:</b> <i>high level test case - (hoog niveau testgeval)</i>  |
| 2.  | Acceptance:            | <i>Acceptatie:</i>                     | <b>Zie:</b> <i>acceptance testing – (acceptatietesten).</i>  |
| 3.  | Acceptance criteria:   | <i>Acceptatie criteria:</i>            | De eind criteria waaraan een component of systeem moet voldoen teneinde geaccepteerd te worden door een gebruiker, klant of andere geautoriseerde entiteit.<br><b>[IEEE 610]</b>   |
| 4.  | Acceptance testing:    | <i>Acceptatietesten:</i>               | Formele testen met betrekking tot gebruikers behoeften, eisen (requirements) en bedrijfsprocessen, die worden uitgevoerd om vast te stellen of een systeem al dan niet aan de acceptatie criteria voldoet, en om de gebruiker, klanten of een andere geautoriseerde entiteit informatie te geven voor het besluit het systeem wel of niet te accepteren.<br><b>[Na IEEE 610]</b> |
| 5.  | Accessibility testing: | <i>Toegankelijkheidstesten:</i>        | Testen om te bepalen hoe makkelijk het is voor gebruikers met een handicap om een component of systeem te gebruiken.<br><b>[Gerrard]</b>   |
| 6.  | Accuracy:              | <i>Nauwkeurigheid:</i>                 | De mate waarin een software product de resultaten of effecten correct en juist kan geven.<br><b>[ISO 9126]</b><br><b>Zie:</b> ook: <i>functionality testing - (functionaliteitstesten)</i>   |
| 7.  | Actual outcome:        | <i>Feitelijke uitkomst:</i>            | <b>Zie:</b> <i>actual result – (feitelijk resultaat).</i>  |
| 8.  | Actual result:         | <i>Feitelijk resultaat:</i>            | Het geproduceerde dan wel geobserveerde gedrag tijdens de test van een component of systeem.   |
| 9.  | Ad hoc review:         | <i>Ad hoc review:</i>                  | <b>Zie:</b> <i>informele review</i>  |
| 10. | Ad hoc testing:        | <i>Ad hoc testen:</i>                  | Informele testen; er vindt geen specificatie van de test plaats, er wordt geen geaccepteerde testontwerptechniek gebruikt, er zijn geen verwachte testresultaten opgesteld en willekeur bepaalt de test uitvoeringsactiviteit.   |
| 11. | Adaptability:          | <i>Aanpasbaarheid:</i>                 | De mogelijkheid van het software product om aangepast te worden aan verschillende gespecificeerde omgevingen zonder andere acties of middelen dan diegenen die geleverd zijn voor dit doel en voor de desbetreffende software [ISO 9126].<br><b>Zie ook:</b> <i>portability – (portabiliteit)</i>  |
| 12. | Agile testing:         | <i>Agile testen (behendig testen):</i> | Test aanpak voor een project dat ‘agile’ (= behendig) methodieken gebruikt, zoals Extreem Programmeren (XP), die ontwikkeling beschouwt als de klant van testen en die de nadruk leggen op het ontwerp paradigma om eerst te testen.<br><b>Zie ook:</b> <i>test driven development - (test gedreven ontwikkeling).</i>   |
| 13. | Algorithm test:        | <i>Algoritme test:</i>                 | <b>[TMap®]</b><br><b>Zie:</b> <i>branch testing – (branch (= programmataak) testen)</i>  |
| 14. | Alpha testing:         | <i>Alfa testen:</i>                    | Gesimuleerd of feitelijk operationeel testen door potentiële gebruikers / klanten of door een onafhankelijk testteam op de locatie van de ontwikkelaars, maar buiten de ontwikkelorganisatie. Alfa testen wordt vaak toegepast voor ‘off-the-shelf’ software als een vorm van interne acceptatietesten.  |
| 15. | Analyzability:         | <i>Analyseerbaarheid:</i>              | De mate waarin het software produkt middelen heeft om te diagnosticeren voor gebreken of foutoorzaken in de software, ofwel om de delen die moeten worden gewijzigd te identificeren. <b>[ISO 9126].</b><br><b>Zie ook:</b> <i>maintainability – (onderhoudbaarheid).</i>  |
| 16. | Analyzer:              | <i>Analyse software:</i>               | <b>Zie:</b> <i>static analyzer – (statische analyse)</i>   |
| 17. | Anomaly:               | <i>Anomalie:</i>                       | Elke conditie die afwijkt van verwachtingen gebaseerd op eisen   |

(requirements) specificaties, ontwerp documenten, gebruikers documenten, standaarden enz. of vanuit iemands perceptie of ervaring. Anomalieën kunnen gevonden worden, tijdens (maar niet alleen tijdens), toetsen, testen, analyse, compilatie, gebruik van software producten of van toepassing zijnde documentatie.

**[IEEE 1044].**

**Zie ook:** *defect, deviation, error, fault, failure, incident, problem - (defect, afwijking, menselijke fout (error), fault, failure, incident, problem).*

- |     |                     |                                   |   |
|-----|---------------------|-----------------------------------|---|
| 18. | Arc testing:        | <i>Arc testen:</i>                | Testen gebaseerd op de verwerkingslogica van het programma of het systeem.  |
| 19. | Attractiveness:     | <i>Aantrekkelijkheid:</i>         | <p><b>Zie:</b> <i>branch testing – (programmatak testen)</i></p> <p>De mogelijkheid van de software om aantrekkelijk te zijn voor een gebruiker</p> <p><b>[ISO 9126].</b></p> <p><b>Zie ook:</b> <i>usability – (bruikbaarheid)</i></p>   |
| 20. | Audit:              | <i>Controle:</i>                  | <p>Een onafhankelijk evaluatie van software producten of processen om te verzekeren dat ze voldoen aan standaarden, richtlijnen, specificaties, en / of procedures, die zijn gebaseerd op objectieve criteria, inclusief documenten die specificeren:</p> <ol style="list-style-type: none"> <li>1. de vorm of inhoud van de producten die moeten worden gemaakt</li> <li>2. het proces dat moet worden gevolgd om de producten te maken</li> <li>3. hoe zal worden vastgesteld dat standaarden of richtlijnen worden nageleefd.</li> </ol> <p><b>[IEEE 1028]</b></p> |
| 21. | Audit trail:        | <i>Audit trail:</i>               | <p>Een pad via welke de originele invoer voor een proces (bijvoorbeeld gegevens) kan worden gevolgd door het proces, met het proces resultaat als beginpunt. Dit ondersteunt foutanalyse en maakt het mogelijk een procescontrole uit te voeren.</p> <p><b>[Volgens TMap®]</b></p>  |
| 22. | Automated testware: | <i>Geautomatiseerde testware:</i> | Testware gebruikt bij geautomatiseerd testen zoals tool scripts.  |
| 23. | Availability:       | <i>Beschikbaarheid:</i>           | <p>De mate waarin een component of systeem operationeel is en toegankelijk wanneer er gebruik van moet worden gemaakt. Veelal uitgedrukt als een percentage.</p> <p><b>[IEEE 610]</b></p>   |

## B

- |     |                       |                                   |   |
|-----|-----------------------|-----------------------------------|---|
| 24. | Back-to-back testing: | <i>Back-to-back testen:</i>       | <p>testen waarbij twee of meer varianten van een component of systeem worden uitgevoerd met dezelfde invoer, waarna de resultaten worden vergeleken en geanalyseerd in het geval van afwijkingen.</p> <p><b>[IEEE 610]</b></p>                      |
| 25. | Baseline:             | <i>Uitgangspunt of startpunt:</i> | <p>Een specificatie of software produkt dat formeel is beoordeeld of overeengekomen en daarna dient als basis voor verdere ontwikkeling, en dat alleen gewijzigd kan worden middels een formeel wijzigingsproces.</p> <p><b>[Naar IEEE 610]</b></p> |
| 26. | Basic block:          | <i>Basisblok:</i>                 | Een reeks van een of meer opeenvolgende uit te voeren programmaregels zonder vertakkingen.  |
| 27. | Basis test set:       | <i>Basis testset:</i>             | Een verzameling van testgevallen, afgeleid van de interne structuur van een component of specificatie om het 100% gespecificeerde dekkingsgraadcriterium te verzekeren.   |
| 28. | Bebugging             | <i>Bebugging:</i>                 | <p><b>[Abbott]</b></p> <p><b>Zie:</b> <i>error seeding – (foutzaaien).</i></p>  |
| 29. | Behaviour:            | <i>Gedrag:</i>                    | De gedragingen van een component of systeem op een verzameling van invoerwaarden en randvoorwaarden   |
| 30. | Benchmark test:       | <i>Referentie test:</i>           | <ol style="list-style-type: none"> <li>1) Een standaard tegen welke metingen of vergelijkingen</li> </ol>   |

- kunnen worden gemaakt
- 2) Een test die gebruikt moet worden om componenten of systemen met elkaar te vergelijken of met een standaard zoals in (1).

**[Naar IEEE 610]**

31.	Bespoke software:	<i>Maatwerk software:</i>	Software specifiek ontwikkeld voor een verzameling gebruikers of klanten. Het tegenovergestelde is 'off-the-shelf' software.
32.	Best practice:	<i>Beste praktijkervaring:</i>	Een superieure methode of vernieuwende ervaring, die bijdraagt aan een organisatie die beter presteert in een bepaald verband, meestal door andere gelijksoortige organisaties als 'beste' erkent.
33.	Beta testing:	<i>Bèta testen:</i>	Operationele testen door een potentiële en / of bestaande gebruiker / klant op een externe locatie, die niet op enig andere manier betrokken is bij de ontwikkelaars, om uit te maken of een component al dan niet de gebruiker / klanten behoeften invult en of het past in het bedrijfsproces. Bèta testen worden vaak gebruikt als een vorm van externe acceptatietesten voor 'off-the-shelf' software om aldus terugkoppeling van de markt te krijgen
34.	Big-bang testing:	<i>Big-bang testen.</i>	Een vorm van integratie testen waarbij de software elementen, hardware elementen of beiden in één keer in een component of volledig systeem worden gecombineerd, in plaats van in stappen. <b>[Na IEEE 610].</b> <b>Zie ook:</b> <i>integration testing – (integratie testen)</i> <b>Zie:</b> <i>black-box design technique – (black-box testontwerptechniek).</i>
35.	Black-box technique:	<i>Black-box techniek:</i>	
36.	Black-box testing:	<i>Black-box testen:</i>	Testen, zowel functioneel als niet-functioneel, zonder referentie naar de interne structuur van een component of systeem.
37.	Black-box test design technique:	<i>Black-box testontwerptechniek:</i>	Procedure om testgevallen af te leiden of te selecteren, gebaseerd op een analyse van de specificatie, functioneel dan wel niet-functioneel, van een component of systeem, zonder referentie naar zijn interne structuur.
38.	Blocked test case:	<i>Geblokkeerd testgeval:</i>	Een testgeval dat niet kan worden uitgevoerd, omdat niet aan de randvoorwaarden van zijn uitvoering is voldaan.
39.	Bottom-up testing:	<i>Bottom-up testen:</i>	Een incrementele benadering van integratietesten waarbij de componenten op het laagste niveau eerst worden getest, en die dan worden gebruikt om het testen van componenten op een hoger niveau mogelijk te maken. Dit proces wordt herhaald tot de component aan de top van de hiërarchie is getest. <b>Zie ook:</b> <i>integration testing – (integratietesten).</i>
40.	Boundary value:	<i>Grenswaarde:</i>	Een invoerwaarde of uitvoerwaarde die op de grens ligt van een equivalentieklasse, of op de kleinste incrementele afstand aan beide kanten van een rand, bijvoorbeeld de minimum of maximum waarde van een bereik.
41.	Boundary value analysis:	<i>Grenswaarde analyse:</i>	een black-box testtechniek waarbij testcases worden ontworpen gebaseerd op grenswaarden.
42.	Boundary value coverage:	<i>Grenswaarde dekkingsgraad:</i>	het percentage van grenswaarden dat is uitgevoerd door een test suite.
43.	Boundary value testing:	<i>Grenswaarde testen:</i>	<b>Zie:</b> <i>boundary value analysis – (grenswaarde analyse)</i>
44.	Branch:	<i>Programmapad:</i>	een basis blok dat geselecteerd kan worden voor uitvoering gebaseerd op een programmaconstructie waarin een van twee of meer alternatieve programmapaden beschikbaar zijn, bijvoorbeeld: case, jump, go to, if-then-else.
45.	Branch condition:	<i>Programmapadconditie:</i>	<b>Zie:</b> <i>condition – (conditie, kan ook als 'voorwaarde' vertalen)</i>
46.	Branch condition combination coverage:	<i>Programmapad conditie-combinatiedekking:</i>	<b>Zie:</b> <i>multiple condition coverage. – (meerdere condities dekking)</i>
47.	Branch condition combination	<i>Programmapad conditie-</i>	<b>Zie:</b> <i>multiple condition testing. – (meerdere condities testen)</i>



48.	testing: Branch condition coverage:	<i>combinatie-testen: Programmapad conditie dekking:</i>	<b>Zie:</b> <i>condition coverage. – (conditie dekking)</i>
49.	Branch coverage:	<i>Programmapaddekking:</i>	het percentage paden dat door een test suite is uitgevoerd. 100% programmapaddekking impliceert zowel 100% beslissingsdekking als 100% programmaregel dekking.
50.	Branch testing:	<i>Programmapadtesten:</i>	een white box testtechniek waarin test cases ontworpen worden om programmapaden uit te voeren.
51.	Bug:	<i>Fout:</i>	<b>Zie:</b> <i>defect – (defect).</i>
52.	Bug:	<i>Fout:</i>	<b>Zie:</b> <i>defect report – (foutrapport)</i>
53.	Business process-based testing:	<i>Bedrijfsproces gebaseerd testen:</i>	Een benadering tot testen waarbij tests gevallen worden ontworpen gebaseerd op beschrijvingen en / of kennis van bedrijfsprocessen.

## C

54.	Capability Maturity Model (CMM):	<i>Capability Maturity Model (CMM):</i>	een raamwerk dat is opgedeeld in 5 opeenvolgende niveaus die de sleutelementen beschrijft van een effectief software proces. Het CMM omvat beste praktijkervaringen voor plannen, ontwerpen en het besturen van software ontwikkeling en -onderhoud.
55.	Capability Maturity Model Integration (CMMI):	<i>Capability Maturity Model Integration (CMMI):</i>	Een raamwerk dat de sleutelementen beschrijft van een effectief produktontwikkelings en -onderhoudsproces. Het CMMI omvat beste praktijkervaringen voor plannen, uitvoeren en het besturen van produktontwikkeling en -onderhoud. CMMI is de aangewezen opvolger van CMM.
56.	Capture / playback tool:	<i>Opname- / afspeelgereedschap:</i>	een testuitvoerings-tool die invoer opneemt gedurende handmatige testen om testscripts te genereren die later automatisch kunnen worden uitgevoerd (dwz: afgespeeld). Deze tools worden vaak gebruikt om automatische regressietesten uit te kunnen voeren.
57.	CASE:	<i>CASE:</i>	acroniem voor Computer Aided Software Engineering. Software engineering met behulp van een computer.
58.	CAST	<i>CAST:</i>	acroniem voor Computer Aided Software Testing. Testen met behulp van een computer. <b>Zie ook:</b> <i>test automation – (testautomatisering)</i>
59.	cause-effect graph:	<i>oorzaak-gevolg graaf:</i>	een grafische representatie van invoer en / of stimuli (oorzaken) met de daarbijbehorende uitvoer (gevolgen), die gebruikt kan worden om testgevallen te ontwerpen.
60.	cause-effect graphing:	<i>oorzaak-gevolg diagrammen:</i>	een black box testontwerptechniek waarbij testgevallen worden ontworpen uitgaande van oorzaak-gevolg diagrammen <b>[BS 7925/2]</b>
61.	cause-effect analysis:	<i>oorzaak-gevolg analyse:</i>	<b>Zie:</b> <i>cause-effect graphing – (oorzaak-gevolg diagrammen).</i>
62.	cause-effect decision table:	<i>beslissingstabel</i>	<b>Zie:</b> <i>decision table – (beslissingstabel).</i>
63.	certification:	<i>certificering:</i>	het proces dat bevestigt dat een component, systeem of persoon voldoet aan de expliciete eisen, bijvoorbeeld door te slagen voor een examen.
64.	changeability:	<i>veranderlijkheid:</i>	De mate waarin de software het toelaat om gespecificeerde aanpassingen door te voeren. <b>[ISO 9126].</b> <b>Zie ook:</b> <i>maintainability – (onderhoudbaarheid).</i>
65.	change control:	<i>wijzigingscontrole:</i>	<b>Zie:</b> <i>configuration control – (configuratiecontrole).</i>
66.	change control board:	<i>wijzigingscontroleforum:</i>	forum verantwoordelijk voor de beslissingen in de wijzigingsprocedure. <b>Zie:</b> <i>configuration control board – (configuratiecontroleforum).</i>
67.	Checker:	<i>Inspecteur:</i>	<b>Zie:</b> <i>reviewer – (inspecteur).</i>
68.	Chow's coverage metrics:	<i>Chow's dekkingsmetriek:</i>	Zie n-switch dekkingsgraad [Chow]
69.	classification tree method:	<i>classificatie boom methode:</i>	een black box testontwerptechniek waarin testgevallen ontworpen worden op basis van combinaties van waarden uit invoer- en / of

70.	<i>Code:</i>	<i>Code:</i>	uitvoerequivalentieklassen. [Grochtmann] <i>Computerinstructies en datadefinities uitgedrukt in een programmeertaal of in een assembleerprogramma, vertaler of ander vertaalprogramma</i> <b>[IEEE 610].</b>
71.	code analyzer:	<i>code analysator:</i>	<b>Zie:</b> <i>static code analyzer – (statische code analysator).</i>
72.	code coverage	<i>code</i> <i>dekkingsgraad:</i>	Een analysemethode die vaststelt welke delen van de software uitgevoerd (of afgedekt) zijn door de testset en welke delen niet, bijvoorbeeld programmaregeldekking, beslissingsdekking of conditiedekking.
73.	code-based testing	<i>programmatesten:</i>	<b>Zie:</b> <i>white box testing – (white box testen)</i>
74.	co-existence:	<i>coëxistentie:</i>	De mate waarin het software product naast andere onafhankelijke software in een gemeenschappelijke omgeving kan bestaan gebruikmakend van gemeenschappelijke bronnen. <b>[ISO 9126]</b> <b>Zie ook:</b> <i>portability – (portabiliteit).</i> <b>Zie:</b> <i>off-the-shelf software – (standaard software).</i>
75.	commercial off-the-shelf software:	<i>commerciële software:</i>	
76.	comparator:	<i>uitvoervergelijker:</i>	apparaat dat uitvoer van producten vergelijkt. <b>Zie:</b> <i>test comparator – (test vergelijkingstool).</i>
77.	compatibility testing:	<i>compatibiliteitstesten:</i>	<b>Zie:</b> <i>interoperability testing – (interoperabiliteitstesten).</i>
78.	compiler:	<i>compiler:</i>	Een software tool dat programma code in een hogere generatie taal omzet naar de overeenkomstige machinetaal.
79.	complete testing:	<i>volledig testen:</i>	<b>Zie:</b> <i>exhaustive testing – (uitputtend testen).</i>
80.	completion criteria:	<i>stopcriteria:</i>	<b>Zie:</b> <i>exit criteria – (stopcriteria).</i>
81.	complexity:	<i>complexiteit:</i>	De mate waarin het ontwerp en / of de interne structuur van een component of systeem moeilijk te begrijpen, onderhouden of verifiëren valt. <i>Zie ook: cyclomatic complexity – (cyclomatische complexiteit).</i>
82.	compliance:	<i>volgzaamheid:</i>	De mate waarin het software product voldoet aan standaarden, conventies, wettelijke regelgeving of soortgelijke voorschriften volgt. <b>[ISO 9126]</b>
83.	compliance testing	<i>volgzaamheidstesten:</i> <i>(navolging)</i>	Het testen om te bepalen of een component of systeem voldoet.
84.	component	<i>component:</i>	Het kleinste deel van de software dat afzonderlijk getest kan worden.
85.	component integration testing	<i>componentintegratietesten:</i>	Testen om defecten in de interfaces en de communicatie tussen gerelateerde componenten aan het licht te brengen.
86.	component specification:	<i>componentspecificatie:</i>	Een beschrijving van de werking van een component in het bijzonder over de uitvoer voor bepaalde invoerwaarden onder bepaalde condities en over het verwachte niet-functionele gedrag (bijvoorbeeld geheugengebruik).
87.	component testing:	<i>component testen:</i>	het testen van afzonderlijke software componenten. <b>[Naar IEEE 610]</b>
88.	compound condition:	<i>samengestelde conditie:</i>	twee of meer enkelvoudige condities, samengesteld door een logische operator (AND, OR, XOR), bijvoorbeeld 'A > B AND C > 1000'.
89.	concrete test case:	<i>welomschreven testgeval:</i>	<b>Zie:</b> <i>low level test case – (laag niveau testgeval).</i>
90.	concurrency testing:	<i>gelijktijdigheidstesten:</i>	testen om te bepalen hoe een component omgaat met het uitvoeren van twee of meer activiteiten binnen een zelfde tijdsinterval, door deze activiteiten afwisselend of gelijktijdig uit te voeren. <b>[Naar IEEE 610]</b>
91.	condition:	<i>conditie:</i>	een logische uitdrukking die Waar of Onwaar oplevert, bijvoorbeeld

			A > B.
			<b>Zie ook:</b> <i>test condition – (testconditie).</i>
92.	condition combination coverage:	<i>conditie combinatie dekkingsgraad:</i>	<b>Zie:</b> <i>multiple condition coverage – (meervoudige conditie dekkingsgraad).</i>
93.	condition combination testing:	<i>conditie combinatietesten:</i>	<b>Zie:</b> <i>multiple condition testing – (meervoudige conditietesten).</i>
94.	condition coverage:	<i>conditiedekking:</i>	Het percentage conditieresultaten dat afgedekt wordt door een testset. 100% conditiedekking vereist dat elke beslissing getest wordt als Waar of Onwaar bij een enkelvoudige conditie.
95.	condition determination coverage:	<i>conditie bepalingstekking:</i>	Het percentage van alle enkelvoudige conditieresultaten die, door het uitvoeren van een testset, elk afzonderlijk een beslissingsresultaat beïnvloed. 100% conditie bepalingstekking betekent 100% beslissingsconditie dekking.
96.	condition determination testing:	<i>conditie bepalingstesten:</i>	Een white box testspecificatietechniek waarbij testgevallen gespecificeerd worden om enkelvoudige conditieresultaten te testen die onafhankelijk van elkaar een beslissingsresultaat beïnvloeden.
97.	<i>Configuration management tool:</i>	<i>Configuratiebeheer gereedschap:</i>	<i>Een gereedschap dat ondersteuning biedt voor, het identificeren en beheren van configuratieonderdelen, de status van wijzigingen en versies en het vrijgeven van de basisversie met configuratieonderdelen.</i>
98.	condition testing:	<i>conditietesten:</i>	Een white box testspecificatietechniek waarbij testgevallen gespecificeerd worden om conditieresultaten te testen.
99.	condition outcome:	<i>conditieresultaat:</i>	het resultaat van een conditie, zijnde Waar of Onwaar.
100.	confidence test:	<i>betrouwbaarheidstest:</i>	<b>Zie:</b> <i>smoke test – (proeftest).</i>
101.	configuration:	<i>configuratie:</i>	de opbouw van een component of systeem bepaald door aantal, aard en de onderlinge verbinding van de samengestelde delen.
102.	configuration auditing:	<i>configuratiecontrole:</i>	de functie om de inhoud van de bibliotheken van configuratieonderdelen te verifiëren, bijvoorbeeld voor het voldoen aan standaarden.
103.	configuration control:	<i>configuratiebeheer:</i>	een onderdeel van configuratiebeheer, bestaand uit de evaluatie, coördinatie, het goed- of afkeuren en het doorvoeren van wijzigingen aan configuratieonderdelen na het formeel vaststellen van hun configuratieidentificatie.
104.	configuration control board (CCB):	<i>configuratiebeheer commissie (CBC):</i>	<b>[IEEE 610]</b> een groep van mensen verantwoordelijk is voor het evalueren en vervolgens goed- of afkeuren van voorgestelde wijzigingen op configuratieonderdelen en toezien op het doorvoeren van de goedgekeurde wijzigingen.
105.	configuration identification:	<i>configuratieidentificatie:</i>	<b>[IEEE 610]</b> een onderdeel van configuratiebeheer bestaand uit het selecteren van configuratieonderdelen van een systeem en het vastleggen van hun functionele en fysieke kenmerken in technische documentatie.
106.	configuration item:	<i>configuratieonderdeel:</i>	<b>[IEEE 610]</b> een samenstelling van hardware, software of beide dat valt onder configuratiebeheer en wordt behandeld als een enkelvoudige eenheid in het configuratiebeheerproces.
107.	configuration management:	<i>configuratiebeheer:</i>	<b>[IEEE 610]</b> Een discipline die technische en administratieve sturing geeft aan en toezicht houdt op: <ul style="list-style-type: none"> <li>• het vaststellen en documenteren van de functionele- en fysieke kenmerken van een configuratieonderdeel,</li> <li>• het beheer van wijzigingen van deze kenmerken,</li> <li>• het vastleggen van en rapporteren over het doorvoeren en invoeren van wijzigen,</li> <li>• het controleren van het voldoen aan de gespecificeerde eisen.</li> </ul>

108.	configuration testing:	<i>configuratie testen:</i>	<b>[IEEE 610]</b> <b>Zie:</b> <i>portability testing – (overdraagbaarheidstesten).</i>
109.	confirmation testing:	<i>bevestigingstesten:</i>	<b>Zie:</b> <i>re-testing – (hertesten).</i>
110.	conformance testing:	<i>conformiteitstesten</i> :	<b>Zie ook:</b> <i>compliance testing – (volgzaamheidstesten).</i>
111.	consistency:	<i>consistentie:</i>	De mate van eenvormigheid, standaardisatie en ontbreken van tegenstrijdigheid tussen de documenten of delen van een component of systeem. <b>[IEEE 610]</b>
112.	control flow:	<i>programmastroom:</i>	Een opeenvolging van gebeurtenissen (programmapaden) tijdens uitvoering van een component of systeem.
113.	control flow graph:	<i>programmastroom diagram:</i>	Grafische voorstelling van een programmastroom.
114.	control flow path:	<i>programmastroom diagrampad:</i>	<b>Zie:</b> <i>path – (programmapad).</i>
115.	conversion testing:	<i>conversietesten:</i>	het testen van software die de data van het bestaande systeem omzet om te worden gebruikt in het vervangende systeem.
116.	COTS:	<i>COTS:</i>	Acroniem voor commerciële standaard software. <b>Zie:</b> <i>off-the-shelf software – (standaard software).</i>
117.	coverage:	<i>dekkingsgraad:</i>	de mate waarin een bepaald dekkingsonderdeel geraakt wordt door een testset, uitgedrukt als percentage van het geheel.
118.	coverage analysis	<i>dekkingsanalyse:</i>	Het toetsen van de bereikte dekking van een dekkingsonderdeel tijdens testuitvoering aan vooropgestelde criteria om te bepalen of extra testen nodig zijn en zo ja, welke testgevallen daarvoor nodig zijn.
119.	coverage item	<i>dekkingsonderdeel:</i>	Een geheel of een eigenschap dat de basis vormt van de testdekkingsgraad, bijvoorbeeld equivalentieclassen of codedeclaraties.
120.	coverage tool	<i>dekkingsgraadtool:</i>	Een tool die objectieve metingen uitvoert over welke structurele elementen, als declaraties en programmapaden, geraakt zijn door een testset.
121.	custom software	<i>maatwerk software:</i>	<b>Zie:</b> <i>bespoke software – (maatwerk software).</i>
122.	cyclomatic complexity	<i>cyclomatische complexiteit:</i>	Het aantal onafhankelijke paden door een programma. De definitie van Cyclomatische complexiteit is: $L - N + 2P$ , waarbij: L = aantal randen, verbindingen in een diagram N = aantal knooppunten in een diagram P = aantal ontkoppelde delen van het diagram (bijvoorbeeld een oproep naar een ander diagram, een deelprogramma) <b>[Naar McCabe]</b>
123.	cyclomatic number	<i>cyclomatic nummer:</i>	<b>Zie:</b> <i>cyclomatic complexity – (cyclomatische complexiteit).</i>
<b>D</b>			
124.	daily build:	<i>dagelijkse oplevering:</i>	een ontwikkelactiviteit waarbij een compleet systeem elke dag (meestal gedurende de nacht) gecompileerd en gelinkt wordt, om op elk moment een consistent systeem beschikbaar te hebben waarin alle laatste wijzigingen zijn opgenomen.
125.	data definition:	<i>datadeclaratie:</i>	een uitvoerbare programmaregel waarbij een waarde wordt toegekend aan een variabele.
126.	data driven testing:	<i>datagericht testen:</i>	een scenariotechniek die zowel invoer als verwachte uitvoer in een tabel of rekenblad opslaat, zodat één enkel controlescenario alle testen in de tabel kan uitvoeren. Datagericht testen wordt vaak toegepast tijdens de test uitvoering met behulp van geautomatiseerde test tools zoals record / playback tools. <b>[Fewster and Graham]</b> <b>Zie ook:</b> <i>keyword driven testing – (actiewoord testen).</i>

127.	data flow:	<i>gegevenscyclus:</i>	een abstracte voorstelling van de volgorde en mogelijke wijzigingen in de toestand van gegevensobjecten, waarbij een object zich in één van de volgende toestanden kan bevinden: creëren, gebruiken, wijzigen of verwijderen. <b>[Beizer]</b>
128.	data flow analysis:	<i>gegevensstroomanalyse:</i>	een vorm van statische analyse gebaseerd op definitie en gebruik van variabelen.
129.	data flow coverage:	<i>gegevensstroomdekking:</i>	gegevensstroomdekking (een dekkingsgraadmetriek voor definitie en gebruik van variabelen (data) tijdens uitvoering van testen).
130.	Data flow testing:	<i>Gegevenscyclustest:</i>	gegevensstroomtest (een structurele (white box) testspecificatietechniek, waarbij testgevallen worden gemaakt, gericht op definitie en gebruik van variabelen binnen de code). <b>Zie:</b> <i>database integrity testing – (database integriteit testen).</i>
131.	data integrity testing:	<i>dataintegriteitstesten:</i>	
132.	database integrity testing:	<i>database integriteitstesten:</i>	het testen van de procedures en processen gebruikt voor toegang en beheer van de data(base), om te waarborgen dat toegangsmethoden, processen en regels functioneren zoals verwacht en dat gedurende de toegang tot de database, gegevens niet corrupt raken of onbedoeld verwijderd, aangepast of aangemaakt worden. <b>Zie:</b> <i>unreachable code – (onbereikbare code).</i> <b>Zie:</b> <i>debuggingtool – (debuggereedschap).</i>
133.	dead code:	<i>dode code:</i>	
134.	debugger:	<i>debugger:</i>	
135.	debugging:	<i>debugging:</i>	het proces van het vinden, analyseren en verwijderen van de oorzaken van fouten (failures) in software.
136.	debugging tool:	<i>fout oplossing gereedschap:</i>	een gereedschap dat door programmeurs wordt gebruikt om operationele fouten (failures) te reproduceren, de toestand van het programma te onderzoeken en de bijbehorende programmafout (defect) te vinden. Een foutverwijderaar maakt het voor programmeurs mogelijk om programma's stap voor stap uit te voeren, de programmauitvoering op elke programmaregel te stoppen en programmavariabelen te onderzoeken of een waarde toe te kennen.
137.	decision:	<i>beslissing:</i>	een beslissingspunt in een programma waarbij het programmaverloop twee of meer alternatieve "paden" heeft. Een knooppunt met twee of meer verbindingen naar aparte programmapaden.
138.	decision condition coverage:	<i>beslissingsconditie dekking:</i>	het percentage van alle conditieuitkomsten en beslissingsuitkomsten welke zijn uitgevoerd bij het testen. 100% beslissingsconditiedekking impliceert zowel 100% conditiedekking en 100% beslissingsdekking.
139.	decision condition testing:	<i>beslissingsconditietesten:</i>	Een structurele (white box) testspecificatietechniek waarmee testgevallen worden gespecificeerd om conditieuitkomsten en beslissingsuitkomsten uit te voeren.
140.	decision coverage:	<i>beslissingsdekking:</i>	Het percentage van beslissingsuitkomsten welke zijn uitgevoerd bij het testen. 100% beslissingsdekking impliceert zowel 100% programmapaddekking als 100% programmaregeldekking.
141.	decision table:	<i>beslissingstabel:</i>	Een tabel met combinaties van invoer en / of stimuli (oorzaken) met de gerelateerde uitkomsten en / of acties (gevolgen), die gebruikt kunnen worden bij het specificeren van testgevallen.
142.	decision table testing	<i>beslissingstabeltesten:</i>	Een functionele (black box) testspecificatietechniek waarmee testgevallen worden ontworpen om de combinaties van invoer en / of stimuli (oorzaken), weergegeven in een tabel, uit te voeren. <b>[Veenendaal]</b>
143.	decision testing:	<i>beslissingstesten:</i>	Een structurele (white box) testspecificatietechniek waarin testgevallen zijn ontworpen om beslissingsuitkomsten uit te voeren.
144.	decision outcome:	<i>beslissingsresultaat:</i>	Het resultaat van een beslissing (waarmee dus wordt bepaald welk programmapad wordt genomen).
145.	defect:	<i>Fout:</i>	Een fout in een component of systeem die er toe kan leiden dat het systeem of component de gewenste functie niet uitvoert, bijvoorbeeld een onjuiste programmaregel of datadefinitie. Wanneer een fout wordt geraakt bij de uitvoering van het programma, kan dit

146.	defect density:	<i>foutdichtheid:</i>	leiden tot een falen (failure) van het systeem of de component. Het aantal defects geïdentificeerd in een component of systeem gedeeld door de grootte van de component of systeem (uitgedrukt in standaard meeteenheden zoals: regels code, aantal klassen of functiepunten).
147.	Defect detection percentage:	<i>Fout detectiepercentage (FDP):</i>	Het aantal fouten gevonden in een testfase, gedeeld door het aantal gevonden fouten in die testfase plus de fouten die daarna worden gevonden op elke mogelijke manier.
148.	Defect management:	<i>Bevindingenbeheer :</i>	Het proces van herkennen, onderzoeken, actie ondernemen en verwijderen van fouten. Hieronder valt het registreren van fouten, classificeren en identificeren van de mogelijke gevolgen (impact). <b>[Naar IEEE 1044]</b> <b>Zie ook:</b> <i>problem management - (probleembeheer).</i>
149.	Defect management tool:	<i>Bevindingenbeheer gereedschap:</i>	Een gereedschap dat het registreren en bijhouden van de status van fouten ondersteunt. Ze hebben vaak werkstroom georiënteerde faciliteiten om de toewijzing, correctie en her-testen van fouten te volgen en beheren en bieden rapportagevoorzieningen. <b>Zie ook:</b> <i>incident management tool – (bevindingenbeheergereedschap).</i>
150.	Defect masking:	<i>Foutmaskeren:</i>	Een gebeurtenis waarbij een fout de detectie van een andere fout voorkomt. <b>[Na IEEE 610]</b>
151.	Defect report:	<i>Foutrapport:</i>	Een rapportage document voor iedere fout in een component of systeem met als gevolg dat een bepaalde uit te voeren functie van een component of systeem faalt.
152.	Defect tracking tool:	<i>Bevindingenbeheer gereedschap</i>	<b>Zie:</b> <i>defect management tool – (bevindingenbeheergereedschap).</i>
153.	definition-use pair:	<i>Definitie-gebruik koppel:</i>	De koppeling van de definitie van een variabele met het gebruik van dezelfde variabele. Het gebruik van variabelen omvat ook berekeningen (bv. vermenigvuldiging) of sturing van het uitvoeren van een programmapad (gebruik van predikaten).
154.	Deliverable:	<i>Op te leveren product:</i>	Elk (werk)product dat opgeleverd moet worden aan iemand anders dan de auteur van het (werk)product.
155.	Design based testing:	<i>Ontwerpgebaseerd testen:</i>	Een testaanpak waarin testgevallen zijn ontworpen gebaseerd op de architectuur en / of detailontwerp van een component of systeem (bv. Tests van interactie tussen componenten of systemen)
156.	Desk checking:	<i>Bureaucontrole:</i>	Het testen van software of specificaties door handmatige simulatie van de uitvoering. <b>Zie ook:</b> <i>static analysis – (statische analyse).</i>
157.	Development testing	<i>Ontwikkeltesten:</i>	Formeel of informeel testen uitgevoerd tijdens het programmeren van een component of systeem, normaliter in de ontwikkelomgeving door ontwikkelaars. <b>[Na IEEE 610]</b>
158.	Deviation:	<i>Bevinding:</i>	<b>Zie:</b> <i>incident – (bevinding).</i>
159.	Deviation report:	<i>Bevindingenrapportage</i>	<b>Zie:</b> <i>incident report – (bevindingenrapport).</i>
160.	Dirty testing	<i>Negatief testen:</i>	<b>Zie:</b> <i>negative testing – (negatief testen).</i>
161.	Documentation testing:	<i>Documentatie testen:</i>	Het testen van de kwaliteit van de documentatie. Bv. gebruikershandleiding of installatiehandleiding.
162.	Domain:	<i>Domein:</i>	De groep waaruit geldige invoer- en / of uitvoerwaarden geselecteerd kunnen worden.
163.	Driver	<i>Stuurprogramma:</i>	Een software component of testgereedschap die een component vervangt welke een andere component of systeem bestuurt en / of aanroept. <b>[Na TMap®]</b>
164.	Dynamic analysis:	<i>Dynamische analyse:</i>	Het proces van gedragsevaluatie, bv. geheugenprestatie, CPU-gebruik, van een systeem of component tijdens uitvoering van het programma. <b>[Na IEEE 610]</b>
165.	Dynamic	<i>Dynamische</i>	Een gereedschap dat informatie verstrekt over de status van de

	analysis tool:	<i>analyse gereedschap:</i>	software tijdens uitvoering. Deze hulpmiddelen worden gewoonlijk gebruikt om niet toegewezen pointers (geheugenaanwijzers) te identificeren, het controleren van berekeningen met pointers, om de toewijzing, gebruik en vrijgave van geheugen te bewaken en om geheugen lekken aan te wijzen.
166.	Dynamic comparison:	<i>Dynamische vergelijking:</i>	Vergelijking van werkelijke resultaat met het verwachte resultaat, uitgevoerd tijdens het uitvoeren van de software, bv. door een test uitvoering gereedschap.
167.	Dynamic testing:	<i>Dynamisch testen:</i>	Tests waarbij de software van een component of systeem wordt uitgevoerd.
<b>E</b>			
168.	Efficiency:	<i>Efficiëntie:</i>	Het vermogen van het software product om de geschikte prestatie te leveren, gerelateerd aan de hoeveelheid gebruikte systeembronnen onder de gestelde condities.
169.	Efficiency testing:	<i>Efficiëntie testen:</i>	<b>[ISO 9126]</b> Het proces van testen om de efficiëntie vast te stellen van een software product.
170.	Elementary comparison testing:	<i>Elementaire vergelijkingentest:</i>	Een black box testspecificatietechniek waarmee testgevallen worden gespecificeerd om combinaties van invoerwaarden uit te voeren, gebruik makend van het conditiebepalingsdekking concept.
171.	Emulator:	<i>Emulator:</i>	<b>[TMap®]</b> Een apparaat, computerprogramma, of systeem dat dezelfde invoer accepteert en uitvoer genereert als het gegeven systeem.
172.	Entry criteria:	<i>Ingangscriteria:</i>	De set van generieke en specifieke condities voor het toelaten van een proces om verder te gaan met een gedefinieerde taak, bv. test fase. Het doel van toegangscriteria is om te voorkomen dat het starten van een taak zou leiden tot (het verloren gaan van) meer inspanning dan het oplossen van het gefaalde toegangscriterium.
173.	Entry point:	<i>Ingangspunt:</i>	De eerste uitvoerbare programmaregel in een component.
174.	Equivalence class:	<i>Equivalentie klasse:</i>	<b>Zie: equivalence partition – (equivalentiepartitie).</b>
175.	Equivalence partition:	<i>Equivalentiepartitie:</i>	Een gedeelte van een invoer- of uitvoerdomein waarvan op basis van de specificatie wordt aangenomen dat het gedrag van het component of systeem hetzelfde is.
176.	Equivalence partition coverage:	<i>Equivalentiepartitie dekking:</i>	Het percentage van equivalentiepartities dat is uitgevoerd door een testsuite.
177.	Equivalence partitioning	<i>Equivalentiepartitioneren</i>	Een black box testspecificatietechniek waarmee testgevallen worden ontworpen om representanten van elke equivalentiepartitie uit te voeren. In principe zijn de testgevallen ontworpen om elke partitie minstens één keer af te dekken.
178.	Error:	<i>Fout:</i>	Een menselijke actie die tot een incorrect resultaat leidt.
179.	Error guessing:	<i>Foutraden:</i>	<b>[Na IEEE 610]</b> Een testspecificatietechniek waarbij de ervaring van de tester wordt gebruikt om te anticiperen op fouten die tengevolge van menselijke fouten (errors) mogelijk aanwezig zijn in het component of systeem dat getest wordt en om de testen zodanig te ontwerpen dat deze fouten worden ontdekt.
180.	Error seeding:	<i>Foutzaaien:</i>	Het proces van opzettelijk toevoegen van bekende fouten aan de reeds aanwezige fouten in het component of systeem met als doel het volgen van de mate van detectie- en verwijdering en het schatten van het resterende aantal fouten.
181.	Error tolerance:	<i>Fouttolerantie:</i>	<b>[IEEE 610]</b> De mogelijkheid van een systeem of component om normaal te blijven werken, ondanks de aanwezigheid van ongeldige invoer.
182.	Evaluation:	<i>Evaluatie</i>	<b>[Na IEEE 610]</b> <b>Zie: testing – (testen).</b>
183.	Exception	<i>Uitzonderingen</i>	Gedrag van een component of systeem ten gevolge van ongeldige

	handling:	<i>afhandeling:</i>	invoer door ofwel een menselijke gebruiker ofwel een ander systeem of component of ten gevolge van een intern fout (failure).
184.	Executable statement:	<i>Uitvoerbare programmeerregel:</i>	Een programmeerregel welke, indien gecompileerd, vertaald is naar programmeercode. De programmeercode zal procedureel worden uitgevoerd als het programma draait en daarbij kan data gemuteerd worden.
185.	Exercised:	<i>Uitgevoerd:</i>	Men spreekt van programmaonderdeeluitvoering door een testgeval, wanneer de invoerwaarde uitvoering van dat programmaonderdeel veroorzaakt, zoals een programmeerregel, beslissing of een ander gestructureerd onderdeel.
186.	Exhaustive testing:	<i>Volledig testen:</i>	Een testaanpak waarbij een testsuite alle invoerwaardencombinaties en pre-condities bevat
187.	Exit criteria	<i>Uitgangscriteria</i>	Een set van generieke en specifieke condities, waarmee de belanghebbenden mee hebben ingestemd om een proces officieel te volbrengen. Het doel van uitgangscriteria is om te voorkomen dat een taak (activiteit) als volledig afgerond wordt gezien, terwijl er nog steeds taakonderdelen zijn die niet zijn afgerond. Uitgangscriteria worden gebruikt voor rapportage en om te bepalen wanneer te stoppen met testen. <b>[Na Gilb and Graham]</b>
188.	Exit point:	<i>Eindpunt:</i>	De laatst uitvoerbare programmeerregel in een component.
189.	Expected outcome:	<i>Verwachte uitkomst:</i>	<b>Zie:</b> <i>expected result – (verwacht resultaat).</i>
190.	Expected result:	<i>Verwacht resultaat:</i>	Het gedrag van een component of systeem, onder specifieke condities voorspeld door een specificatie of andere bron.
191.	<i>Experienced-based test design techniques:</i>	<i>Op ervaringen gebaseerde testspecificatietechniek:</i>	<i>Procedure om testgevallen, gebaseerd op de ervaring van de tester, af te leiden en / of te selecteren.</i>
192.	Exploratory testing:	<i>onderzoekend testen:</i>	Een informele testspecificatietechniek waarbij de tester bewust het ontwerp van de testen controleert op het moment dat de tests worden uitgevoerd en de verzamelde informatie gebruikt om nieuwe en betere tests te ontwerpen. <b>[Naar Bach]</b>
<b>F</b>			
193.	Fail:	<i>Falen:</i>	Een test is mislukt als het werkelijke - en verwachte resultaat niet overeenkomen.
194.	Failure:	<i>Fout:</i>	Afwijking van een component of systeem van zijn verwachte levering, dienst of resultaat. <b>[Naar Fenton]</b>
195.	Failure mode:	<i>Foutmodus:</i>	De fysieke of functionele manifestatie van een fout (failure). Bijvoorbeeld, een systeem in foutmodus kan gekarakteriseerd worden door een trage werking, incorrecte uitvoerwaarden of volledige beëindiging van de uitvoering. <b>[IEEE 610]</b>
196.	Failure Mode and Effect Analysis (FMEA)	<i>FoutModus en GevolgAnalyse (FMGA):</i>	Een systematische aanpak voor risico-identificatie en analyse voor het identificeren van mogelijke foutmodus om hun optreden te voorkomen.
197.	Failure rate:	<i>Foutratio:</i>	De verhouding tussen de hoeveelheid fouten (failures) van een bepaalde categorie en een bepaalde meeteenheid, bijvoorbeeld fouten per tijdseenheid, fouten per aantal transacties, fouten per aantal runs. <b>[IEEE 610]</b>
198.	Fault:	<i>Fout:</i>	<b>Zie:</b> <i>defect – (fout).</i>
199.	Fault density:	<i>Foutdichtheid:</i>	<b>Zie:</b> <i>defect density – (foutdichtheid).</i>
200.	Fault Detection	<i>Foutdetectiepercenten</i>	<b>Zie:</b> <i>Defect Detection Percentage (DDP) - Foutdetectiepercentage</i>



Percentage (FDP):	<i>tage (FDP)</i>	<i>(FDP)</i>
201. Fault masking	<i>Foutverhulling:</i>	Zie: <i>defect masking – (foutmaskeren)</i> .
202. Fault tolerance	<i>Fouttolerantie:</i>	Het vermogen van een software product om een gespecificeerd performance niveau te handhaven in geval van software fouten (defecten) of van een inbreuk op de gespecificeerde interface. <b>[ISO 9126]</b> . <b>Zie ook:</b> <i>reliability – (betrouwbaarheid)</i> .
203. Fault tree analysis:	<i>Analyse d.m.v. foutdiagram:</i>	Een methode gebruikt om de oorzaak van fouten te analyseren (defecten).
204. Feasible path:	<i>Uitvoerbaar pad:</i>	Een pad waarvoor invoerwaarden en précondities bestaan die zorgen voor de uitvoering.
205. Feature:	<i>Kenmerk:</i>	Een attribuut van een component of systeem gespecificeerd of geïmpliceerd door eisdocumentatie (bijvoorbeeld betrouwbaarheid, bruikbaarheid of ontwerprestrictie). <b>[Naar IEEE 1008]</b>
206. Field testing:	<i>Veldtesten:</i>	<b>Zie:</b> <i>beta testen – (bètatesten)</i> .
207. Finite state machine:	<i>Beperkte toestandmachine:</i>	Een rekenmodel dat bestaat uit een beperkte reeks toestanden en overgangen tussen toestanden, mogelijk met bijbehorende acties. <b>[IEEE 610]</b>
208. Finite state testing:	<i>Beperkte toestandtesten:</i>	<b>Zie:</b> <i>state transition testing – (toestandovergangstesten)</i> .
209. Formal review:	<i>Formele review:</i>	Een review die gekarakteriseerd wordt door gedocumenteerde procedures en eisen, bijvoorbeeld inspectie.
210. Frozen test basis:	<i>Bevroren testbasis:</i>	Een testbasis document dat alleen gewijzigd kan worden door een formeel wijzigingsproces. <b>Zie ook:</b> <i>baseline - uitgangspunt</i> .
211. Function Point Analysis (FPA)	<i>Functie Punt Analyse (FPA)</i>	Een methode die erop gericht is de omvang van de functionaliteit van een informatiesysteem te meten. De meting is onafhankelijk van de technologie. Deze meting kan gebruikt worden als basis voor het meten van de productiviteit, het schatten van benodigde middelen en project beheersing.
212. Functional integration:	<i>Functionele integratie:</i>	Een aanpak voor integratie waarin componenten of systemen samengevoegd worden met als doel zo vroeg mogelijk de basisfunctionaliteit werkende te krijgen. <b>Zie ook:</b> <i>integration testing – (integratietesten)</i> .
213. Functional requirement:	<i>Functionele eis:</i>	Een eis die specificeert welke functie een component of systeem moet uitvoeren. <b>[IEEE 610]</b>
214. Functional test design technique:	<i>Functionele testspecificatietechniek:</i>	Procedure om testgevallen af te leiden of selecteren gebaseerd op een analyse van de functionele specificatie van een component of systeem zonder referentie naar de interne structuur. <b>Zie ook:</b> <i>black box test design technique – (black box testspecificatietechniek)</i> .
215. Functional testing:	<i>Functioneel testen:</i>	Testen gebaseerd op een analyse van de functionele specificatie van een component of systeem. <b>Zie ook:</b> <i>black box testing – (black box testen)</i> .
216. Functionality:	<i>Functionaliteit:</i>	Het vermogen van een software product om in functies te voorzien die voldoen aan gedefinieerde en geïmpliceerde behoeften als de software wordt gebruikt onder specifieke omstandigheden. <b>[ISO 9126]</b>
217. Functionality testing:	<i>Functionaliteitstesten:</i>	Het testproces om de functionaliteit van een software product samen te stellen.
<b>G</b>		
218. Glass box testing	<i>Glazen doos testen:</i>	<b>Zie:</b> <i>white box testing- (white box testen)</i> .
<b>H</b>		
219. Heuristic	<i>Heuristische</i>	Een statische testtechniek om de gebruikers-interface te toetsen aan

	evaluation:	<i>evaluatie:</i>	de vastgestelde bruikbaarheidsprincipes (de zogenoemde ‘heuristics’).
220.	High level test case:	<i>Logisch testgeval:</i>	Een testgeval zonder concrete (implementatieniveau) waarden van invoerdata en verwachte resultaten. Logische operatoren zijn; voorkomens van de werkelijke waarden zijn nog niet gedefinieerd en / of beschikbaar. <b>Zie ook:</b> <i>low level test case – fysiek testgeval.</i>
221.	Horizontal traceability:	<i>Horizontale vindbaarheid:</i>	Het volgen van eisen voor een testsoort door de verschillende testdocumentatie heen (bijvoorbeeld testplan, testontwerp specificatie, testgeval specificatie en test procedure specificatie of test script).
<b>I</b>			
222.	Impact analysis	Impact analyse:	De beoordeling van wijzigingen tegen de ontwikkeldocumentatie, testdocumentatie en componenten met als doel een wijziging op een specifieke eis te implementeren.
223.	Incident:	<i>Bevinding:</i>	Elke gebeurtenis die onderzoek vereist. <b>[Naar IEEE 1008]</b>
224.	<i>Incident logging:</i>	<i>Bevindingen vastleggen:</i>	<i>Het vastleggen van de details van een bevinding, b.v. tijdens testen.</i>
225.	Incident management:	<i>Bevindingen beheer:</i>	Het proces van herkennen, onderzoeken, oplossen en sluiten van bevindingen. Het beslaat bevindingen administreren, classificeren en bepalen van de impact. <b>[Naar IEEE 1044]</b>
226.	Incident management tool:	<i>Bevindingenbeheer gereedschap:</i>	Een gereedschap dat faciliteert bij het administreren en volgen van bevindingen. Deze gereedschappen hebben vaak workflow georiënteerde faciliteiten voor het volgen en beheersen van het toewijzen, aanpassen en hertesten van bevindingen en faciliteren het genereren van faciliteiten. <b>Zie ook:</b> <i>defect management tool –(bevindingenbeheergereedschap).</i>
227.	Incident report:	<i>Bevindingenrapport:</i>	Een document dat rapporteert over elke gebeurtenis die optreedt en die onderzoek vereist, bijvoorbeeld tijdens het testen. <b>[Naar IEEE 829].</b>
228.	Incremental development model:	<i>Incrementeel ontwikkelmodel:</i>	Een ontwikkellevenscyclus waarbij een project wordt opgebroken in een serie incrementen, waarbij elk een deel beslaat van de gehele functionaliteit van alle project eisen. De eisen zijn geprioriteerd en worden opgeleverd in volgorde van belangrijkheid binnen een bepaald increment. In sommige (maar niet alle) versies van dit levenscyclusmodel volgt elk subproject een ‘mini V-model’ met zijn eigen ontwerp-, bouw- en testfasen.
229.	Incremental testing:	<i>Incrementeel testen:</i>	Testen waarbij één of meerdere componenten of systemen per keer worden geïntegreerd en getest, totdat alle componenten of systemen zijn geïntegreerd en getest.
230.	Independence:	<i>Onafhankelijkheid:</i>	Scheiding van verantwoordelijkheden wat objectief testen in de hand werkt. <b>[Naar DO-178b]</b>
231.	Infeasible path:	<i>Onmogelijk pad:</i>	Een pad dat niet kan worden uitgevoerd door een set van mogelijke invoerwaarden.
232.	Informal review:	<i>Informele review:</i>	Een review die niet gebaseerd is op een formele (gedocumenteerde) procedure.
233.	Input:	<i>Invoer:</i>	Een variabele (opgeslagen binnen dan wel buiten een component) die gelezen wordt door een component.
234.	Input domain:	<i>Invoerdomein:</i>	De set waaruit mogelijke invoerwaarden geselecteerd kunnen worden. <b>Zie ook:</b> <i>domain –(domein).</i>
235.	Input value:	<i>Invoerwaarde:</i>	Een voorkomen van een invoer. <b>Zie ook:</b> <i>input – (invoer).</i>
236.	Inspection:	<i>Inspectie:</i>	Een soort collegiale review die bestaat uit het visueel onderzoeken van documenten om fouten op te sporen, bijvoorbeeld het afwijken

		van ontwikkelstandaards en het niet aansluiten bij gerelateerde documentatie. Het is de meest formele review techniek en daarom altijd gebaseerd op een gedocumenteerde procedure. <b>[Naar IEEE 610, IEEE 1028].</b> <b>Zie ook:</b> <i>peer review – (collegiaal beoordelen).</i> <b>Zie:</b> <i>moderator - (moderator).</i>
237. Inspection leader:	<i>Inspectieleider:</i>	
238. Inspector:	<i>Inspecteur:</i>	<b>Zie:</b> <i>reviewer - (beoordelaar).</i>
239. Installability:	<i>Installeerbaarheid:</i>	Het vermogen van een software product om te worden geïnstalleerd in een gespecificeerde omgeving. <b>[ISO 9126].</b> <b>Zie ook:</b> <i>portability – (portabiliteit).</i>
240. Installability testing:	<i>Installeerbaarheids testen:</i>	Het proces voor het testen van de installeerbaarheid van een software product. <b>Zie ook:</b> <i>portability testing – (portabiliteitstesten).</i>
241. Installation guide:	<i>Installatie handleiding:</i>	Geleverde instructies op welke media dan ook, die de installateur helpt bij het installatieproces. Denk hierbij aan een gebruikershandleiding, een stap-voor-stap procedure, een installatie wizard of een vergelijkbare procesomschrijving.
242. Installation wizard:	<i>Installatie wizard:</i>	Geleverde software op welke media dan ook, die de installateur door het installatieproces leidt. Het draait het installatieproces, geeft terugkoppeling over installatieresultaten en geeft keuzemogelijkheden aan.
243. Instrumentation:	<i>Instrumentatie:</i>	Het toevoegen van code aan een programma om informatie te verzamelen over het gedrag van het programma tijdens het uitvoeren b.v. voor het meten van codedekking.
244. Instrumenter:	<i>Instrumentator:</i>	software gereedschap om “instrumentatie” te kunnen gebruiken.
245. Intake test:	<i>Innametest:</i>	Een speciaal geval van een proeftest om te beslissen of de component of het systeem voor gedetailleerd en verder testen klaar is. Een innametest wordt typisch uitgevoerd bij het begin van de fase van de testuitvoering. <b>Zie ook:</b> <i>smoke test – (proeftest).</i>
246. Integration:	<i>Integratie:</i>	Het proces om componenten of systemen in een groter geheel te combineren.
247. Integration testing:	<i>Integratietest:</i>	Testen uitgevoerd om fouten in de interfaces en in de interactie tussen geïntegreerde componenten of systemen bloot te leggen. <b>Zie:</b> <i>component-, system integration testing – (component-, systeemintegratietesten).</i>
248. Integration testing in the large:	<i>Integratietesten (voor systemen):</i>	<b>Zie:</b> <i>system integration testing – (systeemintegratietesten).</i>
249. Integration testing in the small:	<i>Integratietesten (voor componenten):</i>	<b>Zie:</b> <i>component integration testing – (componentintegratietest).</i>
250. Interface testing:	<i>Interfacetest:</i>	Een integratietesttype dat zich bezig houdt met de interfaces tussen componenten of systemen.
251. Interoperability:	<i>Interoperabiliteit:</i>	Het vermogen van het softwareproduct om met één of meerdere gespecificeerde componenten of systemen in wisselwerking te treden. [Na ISO 9126] <b>Zie ook:</b> <i>functionality – (functionaliteit).</i>
252. Interoperability testing:	<i>Interoperabiliteitstesten:</i>	Het proces van testen om de interoperabiliteit van een softwareproduct te bepalen. <b>Zie:</b> <i>functionality testing – (functionaliteitstesten).</i>
253. Invalid testing:	<i>ongeldigheidstesten:</i>	Het testen met invoerwaarden die door de component of het systeem zouden moeten worden verworpen. <b>Zie ook:</b> <i>error tolerance – (fouttolerantie).</i>
254. Isolation testing:	<i>Isolatietesten</i>	Het testen van individuele componenten in in afzondering van de omliggende componenten, waarbij omliggende componenten zijn vervangen door stubs en drivers.

255. Item transmittal report: *Overdrachtsrapport:* **Zie:** *release note – (versiebericht).*
256. Iterative development model: *Iteratief ontwikkelmodel:* Een ontwikkellevenscyclus waar een project in grote iteratieve brokstukken is opgedeeld. Een iteratie is een volledige ontwikkelingslijn die in een (intern of externe) oplevering resulteert van een werkend product, als onderdeel van het definitieve te ontwikkelen product. Het product groeit met iedere iteratie tot het uiteindelijke product.

## K

257. Key performance indicator: *Prestatieindicator:* **Zie:** *performance indicator – (prestatieindicator).*
258. Keyword driven testing: *Actiewoordtesten:* Een scriptingtechniek die gegevensbestanden gebruikt die niet alleen testgegevens en verwachte resultaten bevat, maar ook de actiewoorden bevat die op de te testen toepassing betrekking hebben. De actiewoorden worden geïnterpreteerd door speciale ondersteunende scripts die door het controlescript worden aangeroepen voor de test.  
**Zie:** *data driven testing – (data gedreven testen).*

## L

259. LCSAJ: *LCSAJ (Linear Code Sequence And Jump):* Een ‘Linear Code Sequence And Jump-techniek’, bestaat uit de volgende drie elementen (identificatie gebeurt normaal door regelnummers in een uitdraai van broncode van een computerprogramma):
  - het begin van de lineaire volgorde van de uitvoerbare coderegels,
  - het eind van de lineaire volgorde,
  - en de doelcoderegel waarnaar gesprongen wordt, aan het eind van de lineaire volgorde.
260. LCSAJ coverage: *LCSAJ-dekking:* Het percentage LCSAJs van een component die door een testsuite zijn uitgeoefend. 100% LCSAJ-dekking impliceert 100% beslissingsdekking.
261. LCSAJ testing: *LCSAJ-testen:* Een white box testspecificatietechniek, waarin testgevallen worden ontwikkeld voor de uitvoering o.b.v. LCSAJ
262. Learnability: *Leerbaarheid:* Het vermogen van het softwareproduct om de gebruiker de toepassing te laten leren.  
**[ ISO 9126 ]**  
**Zie ook:** *usability – (bruikbaarheid).*
263. Level test plan: *Detailtestplan:* Een testplan dat zich typisch op één testsoort richt.  
**Zie ook:** *test plan – (testplan).*
264. Link testing: *Verbindingstest:* **Zie:** *component integration test – (componentintegratietest).*
265. Load testing: *Belastbaarheidstest:* Een testtype die het gedrag meet van een component of systeem met toenemende belasting, b.v. aantal van parallelle gebruikers en/of aantallen transacties om te bepalen welke belasting door de component of het systeem kan worden verwerkt.  
**Zie:** *stress testing – (duurtest).*
266. Logic-coverage testing: *Logica-dekkingstest:* **[Myers]**  
**Zie:** *white box testing – (white box testen).*
267. Logic-driven testing: *Logica gedreven testen:* **Zie:** *white box testing – (white box testen).*
268. Logical test case: *Logisch testgeval:* **Zie:** *high level test case – (high level testgeval).*
269. Low level test case: *Fysiek testgeval:* Een testgeval met concrete (implementatieniveau) waarden voor invoergegevens en verwachte resultaten. De logische operatoren van logische testgevallen worden vervangen door daadwerkelijke waarden die aan de doelstellingen van de logische operatoren beantwoorden.  
**Zie ook:** *high level test case – (high level testgeval).*

## M

270. Maintenance:	<i>Onderhoud:</i>	Wijziging van een softwareproduct na levering om een fout te corrigeren, prestaties of andere eigenschappen te verbeteren, of het product aan te passen aan een gewijzigde omgeving. <b>[IEEE 1219]</b>
271. Maintenance testing:	<i>Onderhoudstest:</i>	Het testen van de wijzigingen in een operationeel systeem of het effect van een veranderde omgeving op een operationeel systeem.
272. Maintainability:	<i>Onderhoudbaarheid:</i>	Het gemak waarmee een softwareproduct kan worden gewijzigd om fouten te corrigeren, kan worden gewijzigd om aan nieuwe eisen te voldoen, kan worden gewijzigd om toekomstig onderhoud gemakkelijker te maken, of kan worden aangepast aan een veranderende omgeving. <b>[ISO 9126]</b>
273. Maintainability testing:	<i>Onderhoudbaarheidstest:</i>	Het testproces om de onderhoudbaarheid van een softwareproduct vast te stellen.
274. Management review:	<i>Management review:</i>	Een systematische evaluatie van softwareaanschaf, levering, ontwikkeling, werking, of onderhoudsproces. Dit wordt door of namens het management uitgevoerd om de voortgang te bewaken, de status van plannen en programma's te bepalen, eisen en hun systeemtoepassing te bevestigen, of de doeltreffendheid van de managementaanpak, m.b.t.de geschiktheid om het doel te bereiken, te evalueren. <b>[Na IEEE 610, IEEE 1028]</b>
275. Master test plan:	<i>Master testplan:</i>	Een testplan dat zich op meerdere testsoorten richt. <b>Zie ook:</b> <i>test plan – (testplan).</i>
276. Maturity:	<i>Volwassenheid:</i>	(1) Het vermogen van een organisatie op het gebied van doeltreffendheid en de efficiency van zijn processen en werkmethoden. <b>Zie ook:</b> <i>maturity model, test maturity model – (volwassenheidsmodel, testvolwassenheidsmodel).</i> (2) Het vermogen van het softwareproduct om een verstoring als resultaat van fouten (defects) in de software te vermijden. <b>[ISO 9126]</b> <b>Zie ook:</b> <i>reliability – (betrouwbaarheid).</i>
277. Measure:	<i>Meetwaarde:</i>	Het nummer of de categorie die aan een attribuut van een entiteit wordt toegewezen door een meting te doen. <b>[ISO 14598]</b>
278. Measurement:	<i>Meting:</i>	Het proces om een nummer of categorie aan een entiteit toe te wijzen om een attribuut van deze entiteit te beschrijven. <b>[ISO14589]</b>
279. Measurement scale:	<i>Meetschaal:</i>	Een schaal die het type gegevensanalyse beperkt dat op die schaal kan worden uitgevoerd. <b>[ISO 14598]</b>
280. Memory leak:	<i>Geheugenlek:</i>	Een fout (defect) in dynamische opslagallocatie van een programma die veroorzaakt dat deze het geheugen niet schoont (of vrijgeeft) nadat deze klaar is met het gebruik van het geheugen die uiteindelijk veroorzaakt dat het programma faalt wegens gebrek aan geheugen.
281. Metric:	<i>Metriek:</i>	Een meetschaal en de methode die voor de meting gebruikt wordt <b>[ISO 14598]</b>
282. Migration testing:	<i>Migratietest:</i>	<b>Zie:</b> <i>conversion test – (conversietest).</i>
283. Milestone:	<i>Mijlpaal:</i>	Een punt in de tijd binnen een project waarbij (tussentijdse) producten en resultaten klaar moeten zijn.
284. Mistake:	<i>Vergissing:</i>	<b>Zie:</b> <i>error – (fout).</i>
285. Moderator:	<i>Moderator:</i>	De leider en de belangrijkste persoon verantwoordelijk voor een inspectie of ander review-proces.
286. Modified condition	<i>Gewijzigde conditie</i>	<b>Zie:</b> <i>condition determination coverage – (conditie bepalingdekking).</i>

	decision coverage:	<i>beslissingsdekkingt est:</i>	
287.	Modified condition decision testing:	<i>Gewijzigde conditie beslissingsdekkingt est:</i>	<b>Zie:</b> <i>condition determination coverage testing - (conditie bepalingstest).</i>
288.	Modified multiple condition coverage:	<i>Gewijzigde meervoudige conditiedekking:</i>	<b>Zie:</b> <i>condition determination coverage - (conditie bepalingstest).</i>
289.	Modified multiple condition testing:	<i>Gewijzigde meervoudige conditietest:</i>	<b>Zie:</b> <i>condition determination coverage testing -(conditie bepalingstest).</i>
290.	Module:	<i>Module:</i>	<b>Zie:</b> <i>component – (component).</i>
291.	Module testing:	<i>Moduletest:</i>	<b>Zie:</b> <i>component – (component).</i>
292.	Monitor:	<i>Monitor:</i>	Een softwarehulpmiddel of hardware apparaat dat gelijktijdig met de component of het te testen systeem draait en controleert, registreert en / of analyseert wat het gedrag van de component of het systeem is. <b>[Na IEEE 610]</b>
293.	Monitoring tool:	<i>Monitorgereedschap:</i>	<b>Zie:</b> <i>monitor – (monitor).</i>
294.	Multiple condition:	<i>Meervoudige conditie:</i>	<b>Zie:</b> <i>compound condition – (samengestelde conditie).</i>
295.	Multiple condition coverage:	<i>Meervoudige conditiedekking:</i>	Het percentage combinaties van alle enkelvoudige conditieresultaten binnen één coderegel die door een testset zijn uitgevoerd. 100% meervoudige conditiedekking impliceert 100% conditie determinantiedekking.
296.	Multiple condition testing:	<i>Meervoudige conditietest:</i>	Een white box testspecificatietechniek waarmee testgevallen worden ontworpen om combinaties enkelvoudige conditieresultaten (binnen één coderegel) uit te voeren.
297.	Mutation analysis:	<i>Mutatieanalyse:</i>	Een methode om de diepgang van de testsuite te bepalen door te meten in welke mate een testsuite het programma van lichte varianten (mutanten) kan onderscheiden.
298.	Mutation testing:	<i>Mutatietest:</i>	<b>Zie:</b> <i>back-to-back testing – ( back-to-back test).</i>
<b>N</b>			
299.	N-switch coverage:	<i>N-schakelingdekking:</i>	Het percentage opeenvolgingen van N+1 overgangen die door een testsuite zijn uitgeoefend. <b>[Chow]</b>
300.	N-switch testing:	<i>N-schakelingstest:</i>	Een vorm van statusovergangstest waar testgevallen worden ontworpen om alle geldige opeenvolgingen van N+1 overgangen uit te voeren. <b>[Chow]</b> <b>Zie:</b> <i>state transition testing – (statusovergangstest).</i>
301.	Negative testing:	<i>Negatieftesten:</i>	Testen om aan te tonen dat de component of het systeem niet werkt. Het negatief testen is eerder verwant met de houding van de testers dan met een specifieke testbenadering of met een testspecificatietechniek, b.v. testen met ongeldige invoerwaarden of uitzonderingen. <b>[Na Beizer]</b>
302.	Non-conformity:	<i>Non-conformiteit:</i>	Het niet invullen van een gespecificeerde eis. <b>[ISO 9000]</b>
303.	Non-functional requirement:	<i>Niet functionele eis.</i>  <i>Hoedanigheidseis: (extra toegevoegd)</i>	Een eis die niet te maken heeft met functionaliteit, maar met kwaliteitsattributen zoals betrouwbaarheid, efficiëntie, bruikbaarheid, onderhoudbaarheid en overdraagbaarheid. <i>Daarbij staat functionaliteit voor het geheel aan functies waarmee een informatiesysteem een bedrijfsproces ondersteunt. De hoedanigheid staat voor de manier waarop deze functionaliteit</i>

304. Non-functional testing: *Niet functioneel testen:* wordt aangeboden: het totaal van eigenschappen van het product. Het testen van een component of systeem op niet functionele kwaliteitsattributen, zoals betrouwbaarheid, efficiëntie, bruikbaarheid, onderhoudbaarheid en overdraagbaarheid.
305. Non-functional test design techniques: *Niet functionele testspecificatietechnieken:* Procedure om testgevallen af te leiden en / of te selecteren voor niet functionele testen, gebaseerd op een analyse van de specificatie van een component of systeem zonder referentie naar de interne structuur.  
**Zie ook:** *black box test design technique – (black box test ontwerp techniek).*

## O

306. Off-the-shelf software: *Standard software* Een softwareproduct dat is ontwikkeld voor de algemene markt, met andere woorden voor een groot aantal klanten, en wordt aan veel klanten in identiek formaat wordt geleverd.
307. Operability: *Bedienbaarheid:* De mogelijkheid van een softwareproduct een gebruiker in staat te stellen het te gebruiken en te controleren  
**[ISO 9126]**  
**Zie ook:** *usability – (bruikbaarheid).*
308. Operational environment: *Operationele omgeving:* Hardware en softwareproducten die bij de gebruiker of klant zijn geïnstalleerd waar de te testen component of systeem wordt gebruikt. De software kan bestaan uit de operatingsystemen, evenals database management systemen en andere applicaties.
309. Operational profile testing: *Operationeel profiel testen:* Statistisch testen gebruik makend van een model voor systeemgebruik (met kortdurende taken) en hun mogelijkheid voor typisch gebruik  
**[Musa].**
310. Operational testing: *Operationeel testen:* Testen, uitgevoerd om een component of systeem in zijn operationele omgeving te testen en te evalueren  
**[IEEE 610].**
311. Oracle: *Vraagbaak:* **Zie:** *test oracle – (testorakel).*
312. Outcome: *Resultaat:* **Zie:** *result – (resultaat).*
313. Output: *Uitvoer:* Een variabele (opgeslagen in een component of daarbuiten), die door een component wordt beschreven
314. Output domain: *Uitvoerdomein:* De verzameling waaruit valide uitvoerwaarden kunnen worden geselecteerd.  
**Zie ook:** *domain – (domein).*
315. Output value: *Uitvoerwaarde:* Een voorkomen van een uitvoerwaarde.  
**Zie ook:** *output – (uitvoer).*

## P

316. Pair programming: *Paargewijs programmeren:* Een softwareontwikkelaanpak waarbij regels code (zowel voor het product als voor de test) worden geschreven door twee programmeurs die aan een enkele computer zitten. Dit betekent impliciet dat voortdurend en tegelijk code reviews worden uitgevoerd.
317. Pair testing: *Paargewijs testen:* Twee personen, bijvoorbeeld twee testers, een ontwikkelaar en een tester of een eindgebruiker en een tester, werken samen om fouten te vinden. Normaal gesproken delen ze één computer en wisselen ze de controle erover af terwijl ze testen.
318. Partition testing: *Partitietesten:* **[Beizer]**  
**Zie:** *equivalence partitioning – (equivalentie partitionering).*
319. Pass: *Geslaagd:* Een test wordt als geslaagd beschouwd als de feitelijke resultaten en de verwachte resultaten overeenkomen.
320. Pass / fail criteria *Geslaagd / gefaald criteria:* Beslissingsregels om te bepalen of een testonderdeel (functie) of een test is geslaagd of is gefaald.  
**[IEEE 829]**
321. Path: *Pad:* Een opeenvolging van gebeurtenissen zoals uitvoerbare coderegels van een component of systeem vanaf een beginpunt tot een eindpunt.

322. Path coverage:	<i>Paddekkingsgraad:</i>	Het percentage van de paden die door een test suite zijn uitgevoerd. 100% paddekking impliceert 100% LCSAJ-dekking.
323. Path sensitizing:	<i>Pad ontvankelijk maken:</i>	Een verzameling invoerwaarden kiezen, die de uitvoering van een specifiek programmapad forceert.
324. Path testing:	<i>Padtesten:</i>	Een white box testspecificatietechniek waarmee testgevallen worden ontworpen om programmapaden uit te voeren.
325. Peer review:	<i>Collegiale toetsing:</i>	Een collegiale toets van een software werk product van de maker van het product met als doel om fouten (defecten) en verbeteringen te vinden. Voorbeelden zijn inspectie, technische toets en het gezamenlijk doorlopen.
326. Performance:	<i>Performance:</i>	De mate waarin een systeem of component de toegewezen functies volbrengt binnen gestelde verwerkingstijd en doorlooptijd. <b>[Naar IEEE 610].</b> <b>Zie ook:</b> <i>efficiency – (efficiëntie).</i>
327. Performance indicator:	<i>Performance indicator:</i>	Een hoog niveau metric voor effectiviteit en / of efficiëntie gebruikt om te sturen en te controleren op voortgang van de ontwikkeling. Bijvoorbeeld uitloop van de productietijd voor softwareontwikkeling. <b>[CCMi]</b>
328. Performance testing:	<i>Performance testen:</i>	Het testproces om de performance van een softwareproduct te bepalen. <b>Zie ook:</b> <i>efficiency testing – (efficiëntie testen).</i>
329. Performance testing tool:	<i>Performance test tool:</i>	Een tool dat performancetesten ondersteunt en dat meestal twee hoofdvoorzieningen heeft: het genereren van belasting en het meten van de duur van een testtransactie. Het genereren van belasting kan d.m.v. meerdere gebruikers of hoge volumes invoerdata genereren. Gedurende de uitvoering, worden reactietijden gemeten van geselecteerde transacties, en deze worden vastgelegd. Performance testgereedschap verschaft gewoonlijk rapportages van testjournaal, en grafieken van de belasting versus de response tijd.
330. Phase test plan:	<i>Fase testplan:</i>	Een testplan dat zich typisch richt op één testfase. <b>Zie ook:</b> <i>test plan – (testplan).</i>
331. Portability:	<i>Portabiliteit:</i>	Het gemak waarmee een softwareproduct van de ene hardware- of software omgeving naar de andere kan worden overgezet. <b>[ISO 9126]</b>
332. Portability testing:	<i>Portabiliteitstesten:</i>	Het testproces om te bepalen hoe goed een softwareproduct overdraagbaar is naar een andere omgeving.
333. Postcondition:	<i>Postconditie:</i>	Omgevings- en statuscondities waaraan voldaan moet zijn na uitvoering van een test of testprocedure.
334. Post-execution comparison	<i>Post-uitvoeringsvergelijking:</i>	Vergelijking van feitelijke- en verwachte resultaten, uitgevoerd nadat de software gestopt is.
335. Precondition:	<i>Preconditie:</i>	Omgevings- en status condities waaraan voldaan moet zijn voordat een component of systeem kan worden onderworpen aan een bepaalde test of test procedure. <b>Zie:</b> <i>expected result – (verwacht resultaat).</i>
336. Predicted outcome:	<i>Voorspeld resultaat:</i>	
337. Pretest:	<i>Pretest:</i>	<b>Zie:</b> <i>intake test – (ingang test).</i>
338. Priority:	<i>Prioriteit:</i>	De mate van (bedrijfs)belang die aan een onderdeel bijvoorbeeld een fout (defect) wordt toegekend.
339. Probe effect:	<i>Onderzoekseffect:</i>	Het effect op een component of systeem door het meetinstrument wanneer de component of het systeem wordt gemeten, bijvoorbeeld door een performancetestgereedschap of monitor. Bijvoorbeeld kan de performance iets slechter worden wanneer een performancemeetgereedschap wordt gebruikt.
340. Problem:	<i>Probleem:</i>	<b>Zie:</b> <i>defect – (fout).</i>
341. Problem management	<i>Probleembeheer:</i>	<b>Zie:</b> <i>defect management – (bevindingenbeheer).</i>
342. Problem report:	<i>Probleemrapport:</i>	<b>Zie:</b> <i>defect report – (foutrapport).</i>
343. Process:	<i>Process:</i>	Een verzameling van onderling gerelateerde activiteiten, die



		invoerwaarden in uitvoerwaarden omzetten. <b>[ISO 12207]</b>
344. Process cycle test:	<i>Procescyclustest:</i>	Een black box testspecificatietechniek waarmee testgevallen worden ontworpen om bedrijfsprocedures en processen uit te voeren. <b>[TMap®]</b>
345. Product risk:	<i>Productrisico:</i>	Een risico dat direct aan het te testen object is gerelateerd. <b>Zie ook:</b> risk risico.
346. Project:	<i>Project:</i>	Een project is een unieke verzamelingen van gecoördineerde en gecontroleerde activiteiten, met een begin- en een einddatum om een doel te bereiken in overeenkomst met specifieke eisen, inclusief randvoorwaarden betreffende tijd, geld en middelen <b>[ISO 9000]</b>
347. Project risk:	<i>Projectrisico:</i>	Een risico gerelateerd aan het bestuur en controle van een (test)project. <b>Zie ook:</b> risk – (risico).
348. Program instrumenter:	<i>Programma-instrumentator:</i>	<b>Zie:</b> instrumentor – (instrumentator).
349. Program testing:	<i>Programmatesten:</i>	<b>Zie:</b> component testing – (componenttesten).
350. Project test plan:	<i>Project testplan:</i>	<b>Zie:</b> master test plan – (master testplan).
351. Pseudo-random:	<i>Pseudo-random:</i>	Een reeks die willekeurig lijkt te zijn maar die in feite wordt gegenereerd volgens een van te voren bepaalde volgorde.

## Q

352. Quality:	<i>Kwaliteit</i>	De mate waarin een component, systeem of proces voldoet aan gespecificeerde eisen en / of gebruikers / klant behoeften en verwachtingen <b>[naar IEEE 610]</b>
353. Quality assurance:	<i>Kwaliteitsborging:</i>	Onderdeel van het kwaliteitsbeheer, met nadruk op het zorgdragen voor het vertrouwen, dat aan de kwaliteitseisen wordt voldaan. <b>[ISO 9000]</b>
354. Quality attribute:	<i>Kwaliteitsattribuut:</i>	Een eigenschap of karakteristiek die de kwaliteit van een onderdeel beïnvloedt. [IEEE 610]
355. Quality characteristic:	<i>Kwaliteitskenmerk:</i>	<b>Zie:</b> quality attribute – (kwaliteits attribuut).
356. Quality management:	<i>Kwaliteitsbeheer:</i>	Gecoördineerde activiteiten die een organisatie richting geeft en controleert m.b.t. kwaliteit. Richting en controle m.b.t. kwaliteit leidt in het algemeen tot het instellen van kwaliteitbeleid en kwaliteitsdoelen, kwaliteitsplanning, kwaliteitscontrole, kwaliteitsborging en kwaliteitsverbetering. <b>[ISO 9000]</b>

## R

357. Random testing:	<i>Toevals testen</i>	Een black box testspecificatietechniek waarbij testgevallen worden geselecteerd, mogelijk met behulp van een pseudo-toevalsgeneratie algoritme, om te voldoen aan een operationeel profiel. Deze techniek kan gebruikt worden voor het testen van niet-functionele kwaliteitsattributen, zoals betrouwbaarheid en performance.
358. Recorder:	<i>Schrijver:</i>	<b>Zie:</b> scribe – (notulist).
359. Record / playback tool:	<i>Opname- / afspeelgereedschap:</i>	<b>Zie:</b> capture / playback tool – (opname- / afspeelgereedschap).
360. Recoverability:	<i>Herstelbaarheid:</i>	De mogelijkheid van een softwareproduct om opnieuw een bepaald niveau van performance te halen en om data terug te halen in het geval van een opgetreden fout. <b>[ISO 9126]</b>
361. Recoverability testing:	<i>Herstelbaarheidstesten:</i>	<b>Zie ook:</b> reliability – (betrouwbaarheid). Het proces van testen om uit te maken hoe goed de herstelbaarheid van een softwareproduct is.

362. Recovery testing:	<i>Hersteltesten:</i>	<b>Zie ook:</b> <i>reliability testing – (betrouwbaarheidstesten).</i> <b>Zie:</b> <i>recoverability testing – (herstelbaarheidstesten).</i>
363. Regression testing:	<i>Regressie testen:</i>	Het testen van een eerder getest programma na een wijziging, om aan te tonen dat er geen fouten (defecten) zijn geïntroduceerd of ontdekt in ongewijzigde gebieden van de software als gevolg van die wijzigingen. Het vindt plaats wanneer de software of de omgeving is gewijzigd.
364. Regulation testing:	<i>Reglementtesten:</i>	<b>Zie:</b> <i>compliance testing – (volgzaamheidstesten).</i>
365. Release note:	<i>Opleveringsdocument:</i>	Een document dat testeenheden, hun configuratie, huidige status en andere opleveringsdetails bevat. Dit wordt overhandigd door het ontwikkelteam aan het testteam en mogelijk andere betrokkenen bij aanvang van de testuitvoeringsfase.
366. Reliability:	<i>Betrouwbaarheid:</i>	[Naar IEEE 829] Het vermogen van de software om zijn vereiste functies uit te voeren onder gestelde voorwaarden gedurende een bepaalde tijdspanne of gedurende een bepaald aantal bewerkingen.
367. Reliability testing:	<i>Betrouwbaarheidstesten:</i>	[ISO 9126] Het testproces om de betrouwbaarheid van de software te bepalen.
368. Replaceability:	<i>Vervangbaarheid:</i>	Het vermogen van een softwareproduct om een ander software product met hetzelfde doel en in dezelfde omgeving te vervangen. [ISO 9126]
369. Requirement:	<i>Eis:</i>	<b>Zie ook:</b> <i>portability – (overdraagbaarheid).</i> Een voorwaarde of mogelijkheid van een gebruiker om een probleem op te lossen of een doel te bereiken waaraan een systeem of subsysteem moet voldoen, waarbij aan een contract, standaard, specificatie of een ander formeel opgelegd document wordt voldaan. [Naar IEEE 610]
370. Requirements based testing:	<i>Op eisen gebaseerde testen:</i>	Een testaanpak waarmee testgevallen ontworpen worden op basis van testdoelen en testcondities die afgeleid zijn van eisen. B.v. testen die specifieke functies uitvoeren of het onderzoeken van niet-functionele kwaliteitsattributen als betrouwbaarheid of bruikbaarheid.
371. Requirements management tool:	<i>Eisenbeheergereedschap:</i>	Een gereedschap dat ondersteunt bij het vastleggen van kenmerken van eisen (belang, kennis, verantwoordelijkheid) en aantekeningen en ondersteunt bij de traceerbaarheid op verschillende niveaus en het bijhouden van wijzigingen. Sommige van dergelijke gereedschappen bieden ook functies als statische analyse, zoals consistentiecontrole en controle op het overtreden van voorgedefinieerde regels betreffende eisen.
372. Requirements phase:	<i>Fase van een eis:</i>	De tijdspanne in de software levenscyclus waarin de eisen voor een softwareproduct worden gedefinieerd en vastgelegd. [IEEE 610]
373. Resource utilization:	<i>Brongebruik:</i>	Het vermogen van het softwareproduct om de juiste hoeveelheid en type van de hulpbronnen te gebruiken, bijvoorbeeld de hoeveelheid te gebruiken primair- en secundair geheugen door het programma en de grootte van de benodigde tijdelijke- of overloopbestanden, wanneer de software zijn functies uitvoert onder bepaalde omstandigheden. [Naar ISO 9126]
374. Resource utilization testing:	<i>Brongebruiktesten:</i>	<b>Zie ook:</b> <i>efficiency – (efficiëntie).</i> Het testproces om het gebruik van hulpbronnen van een softwareproduct vast te stellen. <b>Zie ook:</b> <i>efficiency testing – (efficiëntie testen).</i>
375. Result:	<i>Resultaat:</i>	Het gevolg / uitkomst van het uitvoeren van een test. Dat houdt schermuitvoer, gegevenswijziging, rapporten en het versturen van communicatieboodschappen in. <b>Zie ook:</b> <i>actual result, expected result – (feitelijk resultaat, verwacht</i>

		<i>resultaat).</i>
376. Resumption criteria:	<i>Hervattingscriteria :</i>	De testactiviteiten die herhaald moeten worden als het testen wordt hervat na een onderbreking. <b>[Naar IEEE 829]</b>
377. Re-testing:	<i>Hertesten:</i>	Testgevallen die de laatste keer niet geslaagd waren en herhaald worden om te verifiëren of de wijzigingen goed zijn.
378. Review:	<i>Toetsing:</i>	De evaluatie van een product of projectstatus om tegenstrijdigheden t.o.v. geplande resultaten of doelstellingen vast te stellen en verbeteringen voor te stellen. Bijvoorbeeld een informele toetsing, technische toetsing, inspectie en walkthrough <b>[Naar IEEE 1028]</b>
379. Reviewer:	<i>Toetsers:</i>	De persoon betrokken bij een toetsing, die tegenstellingen identificeert en beschrijft die voorkomen in het product of het project dat getoetst wordt. Toetsers kunnen gekozen worden om een verschillende invalshoeken en rollen in het toetsingsproces te vervullen.
380. Review tool:	<i>Toetsingsgereedschap:</i>	Een gereedschap dat het toetsingsproces ondersteunt. Typisch voorziet dit gereedschap in functies om de toetsplanning, opvolging en de communicatie te ondersteunen, om gezamenlijke toetsen uit te voeren en bevat dit gereedschap een opslagmogelijkheid om metrieken te verzamelen en daarover te rapporteren.
381. Risk:	<i>Risico:</i>	Een factor die kan uitmonden in toekomstige negatieve gevolgen; gewoonlijk uitgedrukt in effect en waarschijnlijkheid.
382. Risk analysis:	<i>Risicoanalyse:</i>	Het proces om vastgestelde risico's te beoordelen om hun effect en waarschijnlijkheid van optreden in te schatten.
383. Risk-based testing:	<i>Risico gebaseerd testen:</i>	Testen gericht op het aftasten en het verschaffen van informatie over productrisico's. <b>[Naar Gerrard]</b>
384. Risk control:	<i>Risico beheersing:</i>	Het proces waarbij besluitvorming plaats vindt en beperkende maatregelen implementeerd om risico's te verminderen of om risico's binnen bepaalde grenzen te houden.
385. Risk identification:	<i>Risicobepaling:</i>	Het proces om risico's vast te stellen, gebruik makend van technieken zoals brainstormen, checklists en fouthistorie.
386. Risk management:	<i>Risicomangement :</i>	Systematisch toepassen van procedures en ervaringen om het vaststellen, analyseren, prioriteren en het beheersen van risico's.
387. Risk mitigation:	<i>Risicovermindering:</i>	<b>Zie: risk control – (risicobeheersing).</b>
388. Robustness:	<i>Robuustheid:</i>	De mate waarin een component of systeem correct kan functioneren tijdens ongeldige invoer of onder zware omgevingsfactoren. <b>[IEEE 610]</b> <b>Zie ook: fault tolerance – (fouttolerantie).</b>
389. Robustness testing:	<i>Robuustheidstest</i>	Testen om de robuustheid van een systeem te bepalen.
390. Root cause:	<i>Hoofdoorzaak:</i>	Een onderliggende factor die afwijkend is en die mogelijk permanent uitgeschakeld moet worden door procesverbetering.

## S

391. Safety:	<i>Veiligheid:</i>	Het vermogen van een softwareproduct om een acceptabel risiconiveau te bereiken waarbij de schade aan mensen, bedrijf, software, eigendom of het milieu aanvaardbaar is binnen een bepaalde gebruikscontext. [ISO 9126]
392. Safety testing:	<i>Veiligheidstesten:</i>	Testen om de veiligheid van een softwareproduct te bepalen.
393. Sanity test:	<i>Gezondheidstest:</i>	<b>Zie: smoke test – (proeftest).</b>
394. Scalability:	<i>Schaalbaarheid:</i>	De mogelijkheid van een softwareproduct om te worden opgewaardeerd om verhoogde werklast aan te kunnen. <b>[Naar Gerrard]</b>
395. Scalability testing:	<i>Schaalbaarheidstesten:</i>	Testen om de schaalbaarheid van een softwareproduct te bepalen.
396. Scenario testing	<i>Scenariotesten:</i>	<b>Zie: use case testing – (use case testen).</b>

397. Scribe:	<i>Notulist:</i>	De persoon die iedere fout (defect) en elke verbeteringsuggestie tijdens een loggingbijeenkomst noteert op een aantekenformulier. De notulist moet erop toezien dat het formulier leesbaar en begrijpelijk is.
398. Scripting language:	<i>Scriptingtaal:</i>	Een programmeertaal waarin uitvoerbare testscripts worden geschreven die gebruikt worden door testautomatiseringsgereedschap (bijvoorbeeld een capture/playback tool).
399. Security:	<i>Beveiliging:</i>	Kenmerken van softwareproducten die invloed hebben op de mogelijkheid om op opzettelijke of toevallige wijze ongeautoriseerde toegang tot programma's of gegevens te voorkomen. <b>[ISO 9126]</b> <b>Zie ook:</b> <i>functionality – (functionaliteit).</i>
400. Security testing:	Beveiligingstesten:	Testen om de beveiliging van een softwareproduct te bepalen. <b>Zie ook:</b> <i>functionality testing – (functionaliteit testen).</i>
401. <i>Security testing tool:</i>	<i>Beveiligingstestgereedschap:</i>	<i>Een gereedschap dat ondersteuning biedt voor het testen van veiligheidskwaliteitsattributen en kwetsbaarheden.</i>
402. <i>Security tool:</i>	<i>Beveiligingsgereedschap:</i>	<i>Een gereedschap dat ondersteuning biedt voor operationele beveiliging.</i>
403. Serviceability testing:	<b><i>Bruikbaarheids of duurzaamheids testen????</i></b>	<b><i>Zie: maintainability testing – (onderhoudbaarheidstesten).</i></b>
404. Severity:	<i>Ernst:</i>	De mate van impact die een fout (defect) heeft op de ontwikkeling of op het functioneren van een component of systeem. <b>[Naar IEEE 610]</b>
405. Simulation:	<i>Simulatie:</i>	Het nabootsen van bepaalde gedragingen van een fysiek of abstract systeem door een ander systeem. <b>[ISO 2382/1]</b>
406. Simulator:	<i>Simulator:</i>	Een toestel, computer programma of systeem dat tijdens het testen gebruikt wordt, dat zich gedraagt of dat werkt als een bepaald systeem wanneer het voorzien wordt van een bepaalde verzameling gegevens. <b>[Naar IEEE 610, DO178b]</b> <b>Zie ook:</b> <i>emulator – (emulator).</i>
407. Site acceptance testing:	<i>Locatie acceptatietest</i>	Acceptatietesten op de locatie van en door gebruikers / klanten om te bepalen of een component al dan niet voldoet aan de verwachtingen van die gebruiker / klant en om te bepalen of het aansluit bij de bedrijfsprocessen. Meestal betreft het zowel hard- als software.
408. smoke test	<i>proeftest</i>	Een deel van alle beschreven / geplande testgevallen die de belangrijkste functionaliteiten van een component of systeem afdekken, om te verzekeren dat de kritische functies van een programma werken, zonder verdere details in beschouwing te nemen. Een dagelijkse oplevering en een proeftest behoren tot de beste ervaringen uit de industrie. <b>Zie ook:</b> <i>intake test – (innametest).</i>
409. Software:	<i>Software:</i>	Computerprogramma's, procedures en mogelijk bijhorende documentatie en gegevens die relevant zijn voor de uitvoering van een computersysteem. <b>[IEEE 610]</b>
410. Software feature:	<i>Software-kenmerk:</i>	<b>Zie:</b> <i>feature – (kenmerk).</i>
411. Software quality:	<i>Softwarekwaliteit:</i>	Het geheel van functionaliteit en eigenschappen van een softwareproduct dat ondersteuning biedt om te voldoen aan expliciete of impliciete behoeften. <b>[Naar ISO 9126]</b>
412. Software quality characteristic:	<i>Software kwaliteitskenmerk:</i>	<b>Zie:</b> <i>quality attribute – (kwaliteitsattribuut).</i>
413. Software test incident:	<i>Software testbevinding:</i>	<b>Zie:</b> <i>incident – (bevinding).</i>
414. software test	<i>Software</i>	<b>Zie:</b> <i>incident report – (bevindingenrapport).</i>

incident report:	<i>testbevindingenrap port:</i>	
415. Software Usability Measurement Inventory (SUMI):	<i>Software Usability Measurement Inventory (SUMI):</i>	Een testtechniek gebaseerd op een vragenlijst die de bruikbaarheid, b.v. gebruikerstevredenheid, van een component of systeem beoordeelt. <b>[Veenendaal]</b>
416. Source statement:	<i>Bronregel:</i>	<b>Zie:</b> <i>statement – (coderegel).</i>
417. Specification:	<i>Specificatie:</i>	Een document dat de eisen, het gedrag en andere kenmerken van een component of systeem beschrijft, idealiter op een volledige, nauwkeurige en te controleren manier. Vaak worden de procedures om te bepalen of aan deze voorwaarden is voldaan ook beschreven. <b>[Naar IEEE 610]</b>
418. Specification based testing:	<i>Specificatie gebaseerd testen:</i>	<b>Zie:</b> <i>black box testing – (black box testen).</i>
419. Specification based test design technique:	<i>Specificatie gebaseerde testspecificatietechniek:</i>	<b>Zie:</b> <i>black box test design technique – (black box testspecificatietechniek)</i>
420. Specified input:	<i>Beschreven invoer:</i>	Een invoerwaarde waarvoor de specificatie een resultaat voorspelt.
421. Stability:	<i>Stabiliteit:</i>	De mogelijkheid van een softwareproduct om onverwachte effecten van wijzigingen in de software te vermijden. <b>[ISO 9126]</b> <b>Zie ook:</b> onderhoudbaarheid.
422. standard software:	<i>Standard software</i>	<b>Zie:</b> off-the-shelf software – (standaard software).
423. standards testing:	<i>Testen van standaards</i>	<b>Zie:</b> <i>compliance testing – (conformiteitstesten).</i>
424. State diagram:	<i>Toestandsdiagram:</i>	Een diagram dat weergeeft welke toestanden een component of systeem kan aannemen en toont welke gebeurtenissen of omstandigheden kunnen leiden tot of resulteren in een toestandswijziging. <b>[IEEE 610].</b>
425. State table:	<i>Toestandstabel:</i>	Een tabel die de resulterende toestandsovergangen toont voor alle toestanden gecombineerd met elke mogelijke gebeurtenis, zowel geldige als ongeldige overgangen.
426. State transition:	<i>Toestandsovergang :</i>	Een overgang tussen twee toestanden van een component of systeem.
427. State transition testing:	<i>Toestandsovergang testen:</i>	Een functionele (black box) test specificatie techniek waarmee de testgevallen ontworpen worden om de geldige en ongeldige toestandsovergangen uit te voeren. <b>Zie ook:</b> <i>N-switch testing – (N-switch testen).</i>
428. Statement:	<i>Programmaregel:</i>	Een entiteit in een programmeertaal, die typisch bestaat uit de kleinste mogelijke ondeelbare eenheid van programmauitvoering.
429. Statement coverage:	<i>Programmaregeldekking:</i>	Het percentage van uitvoerbare programmaregels die zijn uitgevoerd door een testset.
430. Statement testing:	<i>Programmaregelingen:</i>	Een structurele (white-box) testspecificatietechniek waarmee testgevallen ontworpen worden om programmaregels uit te voeren.
431. Static analysis:	<i>Statische analyse:</i>	Analyse van softwareproducten, bijv. eisen of code, gedaan zonder deze softwareproducten uit te voeren.
432. Static analysis tool:	<i>Statische analyse gereedschap:</i>	<b>Zie:</b> <i>static code analyzer – (statische code analyse).</i>
433. Static analyzer:	<i>Statische analyse gereedschap:</i>	Een gereedschap dat statische analyse uitvoert.
434. Static code analysis :	<i>Statische code-analyse:</i>	Analyse van broncode zonder dat de software wordt uitgevoerd.
435. Static code analyzer:	<i>Statische analyse gereedschap:</i>	Een gereedschap dat statische code analyse uitvoert. Het gereedschap controleert bron code op bepaalde eigenschappen zoals

436. Static testing:	<i>Statisch testen:</i>	het volgen van codeerstandaarden, kwaliteitsmetrieken of afwijkingen in de gegevensstroom. Testen van een component of systeem op specificatie- of implementatie-niveau, zonder die software uit te voeren, bijv. reviews of statische code analyse.
437. Statistical testing:	<i>Statistisch testen:</i>	Een testspecificatietechniek waarin een model van de statistische verdeling van de invoer gebruikt wordt om representatieve testgevallen te construeren. <b>Zie ook:</b> <i>operational profile testing – (operationeel profieltesten)</i> .
438. Status accounting:	<i>Status bijhouden:</i>	Een onderdeel van configuratiebeheer, dat bestaat uit het vastleggen en rapporteren van informatie die nodig is om een configuratie effectief te beheren <b>[IEEE610].</b>
439. Storage:	<i>Opslag:</i>	<b>Zie:</b> <i>resource utilization – (bronnengebruik)</i> .
440. Storage testing:	<i>Opslag testen:</i>	<b>Zie:</b> <i>resource utilization testing – (bronnengebruiktesten)</i> .
441. Stress testing:	<i>Stress testen:</i>	Testen die erop gericht zijn om een systeem of component te evalueren op of over de grenzen van de daarvoor gespecificeerde eisen <b>[IEEE610].</b>
442. Structure-based techniques:	<i>structuur gebaseerde technieken:</i>	<b>Zie ook:</b> <i>load testing – (belastbaarheidstesten)</i> . <b>Zie:</b> <i>structural white box test design technique – (structurele (whitebox) test specificatietechniek)</i> .
443. Structural coverage:	<i>structurele dekkingsgraad:</i>	Mate van dekkingsgraad metingen gebaseerd op de interne structuur van een component of systeem.
444. Structural test design technique:	<i>Structurele testspecificatietechniek:</i>	<b>Zie:</b> <i>white box test design technique – (whitebox testspecificatietechniek)</i> .
445. Structural testing:	<i>Structureel testen:</i>	<b>Zie:</b> <i>white box testing – (white box testen)</i> .
446. Structured walkthrough:	<i>Gestructureerd doorlopen:</i>	<b>Zie:</b> <i>walkthrough – (doorlopen)</i> .
447. Stub:	<i>Stub:</i>	Een raamwerk of specifieke implementatie van een software component, gebruikt om een component te ontwikkelen of te testen die aanroept of er op een andere manier van afhankelijk is. Het vervangt een aan te roepen component. <b>[Naar IEEE610].</b>
448. Subpath:	<i>Subpad:</i>	Een reeks van uitvoerbare programmaregels binnen een component.
449. Suitability:	<i>Toepasbaarheid:</i>	Het vermogen van een softwareproduct om in een geschikte verzameling van functies voor specifieke taken en gebruikersdoelen te voorzien. <b>[ISO9126]</b>
450. Suspension criteria:	<i>Opschortingscriteria:</i>	<b>Zie ook:</b> <i>functionality – (functionaliteit)</i> . De criteria die gebruikt worden om (tijdelijk) alle of een deel van de testactiviteiten op testonderdelen te stoppen. <b>[Naar IEEE 829]</b>
451. Syntax testing:	<i>Syntactisch testen:</i>	Een functionele (black box) testspecificatietechniek waarmee testgevallen ontworpen worden gebaseerd op de definitie van het invoerdomein en / of uitvoerdomein.
452. System:	<i>System:</i>	Een groep componenten opgezet om een specifieke functie of groep van functies tot stand te brengen.
453. System integration testing:	<i>Systeemintegratie testen:</i>	Testen van de integratie van systemen en onderdelen, testen van koppelingen naar externe organisaties (bijv. elektronische data uitwisseling, Internet).
454. System testing:	<i>Systeemtesten:</i>	Het proces van het testen van een geïntegreerd systeem om te verifiëren dat het aan de gespecificeerde eisen voldoet.

## T

455. Technical	<i>Technische</i>	Een collegiale groepsdiscussie die gericht is op het bereiken van
----------------	-------------------	---

review:	<i>toetsing:</i>	consensus over de te nemen inhoudelijke aanpak. <b>[Gibb and Graham, IEEE 1028]</b> <b>Zie ook:</b> <i>peer review – (collegiale toetsing).</i>
456. Test:	<i>Test:</i>	Een verzameling van één of meer testgevallen.
457. Test approach:	<i>Testaanpak:</i>	De implementatie van een teststrategie voor een specifiek project. Typisch omvat het de genomen beslissingen die volgen op basis van het doel van het (test-)project en de uitgevoerde risicoanalyse, uitgangspunten van het testproces, de toe te passen testspecificatietechnieken, uitgangscriteria, en uit te voeren testtypen.
458. Test automation:	<i>Testautomatisering:</i>	Het gebruik van software om test activiteiten uit te voeren of te ondersteunen, bijv. testbeheer, testontwerp, testuitvoering en het controleren van de resultaten.
459. Testbasis:	<i>Testbasis:</i>	Alle documenten waarvan de eisen voor een component of systeem kunnen worden afgeleid. De documentatie waarop de testgevallen zijn gebaseerd. Als een document alleen gewijzigd kan worden volgens een formele wijzigingsprocedure, dan wordt de test basis een ‘bevroren test basis’ genoemd. <b>[Naar TMap®]</b>
460. Test bed:	<i>Testbedding:</i>	<b>Zie:</b> <i>test environment – (test omgeving).</i>
461. Test case:	<i>Testgeval:</i>	Een verzameling van invoerwaarden, voorwaarden voor uitvoering, verwachte resultaten en voorwaarden na uitvoering, ontwikkeld voor een bepaald doel of testconditie, zoals een bepaald programmapad uit te voeren of het verifiëren of is voldaan aan een specifieke eis. <b>[Naar IEEE610].</b>
462. Test case design technique:	<i>Testgeval specificatietechniek:</i>	<b>Zie:</b> <i>test design technique – (test specificatie techniek).</i>
463. Test case specification:	<i>Testgevalspecificatie:</i>	Een document dat een verzameling testgevallen (doelstelling, invoer, acties, verwachte resultaten, en voorwaarden voor uitvoer) specificeert voor een testonderdeel. <b>[Naar IEEE829].</b>
464. Test case suite:	<i>Testgevallenreeks:</i>	<b>Zie:</b> <i>test suite – (testset).</i>
465. Test charter:	<i>Testhandvest:</i>	Een vastlegging van testdoelstellingen en mogelijk ideeën over hoe te testen. Een testopzet wordt bijvoorbeeld vaak gebruikt bij onderzoekend testen. <b>Zie ook:</b> <i>exploratory testen – (onderzoekend testen).</i>
466. Test closure:	<i>Testafronding:</i>	Tijdens de afrondingsfase van een test proces worden gegevens verzameld van de voltooide activiteiten om ervaringen, testware, feiten en cijfers te consolideren. De testaf rondingsfase bestaat uit het beëindigen en archiveren van de testware en uit het evalueren van het testproces, inclusief de voorbereiding van het testevaluatierapport. <b>Zie ook:</b> <i>test process – (testproces).</i>
467. Test comparator:	<i>Testvergelijkingsprogramma:</i>	Een testgereedschap om een geautomatiseerde testvergelijking uit te voeren.
468. Test comparison:	<i>Testvergelijking:</i>	Het proces van identificeren van verschillen tussen de werkelijke resultaten geproduceerd door de geteste component of het systeem en de verwachte resultaten voor een test. Testvergelijking kan tijdens testuitvoering (dynamische vergelijking) of na testuitvoering gedaan worden. <b>Zie:</b> <i>exit criteria – (uitgangscriteria).</i>
469. Test completion criteria:	<i>Testafrondingscriteria:</i>	
470. Test condition:	<i>Testconditie:</i>	Een onderdeel of gebeurtenis van een component of systeem dat geverifieerd zou kunnen worden door één of meer testgevallen, bijv. een functie, transactie, eigenschap, kwaliteitsattribuut of structureel element.
471. Test control:	<i>Testbeheer:</i>	Een testbeheerstaak die er voor zorgt dat een aantal maatregelen ontwikkeld en toegepast worden om het testproject op koers te krijgen wanneer de projectbewaking een afwijking aantoont t.o.v.

			wat was gepland.
			<b>Zie ook:</b> <i>test management – (testbeheer).</i>
472. Test coverage:	<i>Testdekkingsgraad:</i>		<b>Zie:</b> <i>coverage – (dekkingsgraad).</i>
473. test cycle:	<i>Testcyclus:</i>		Uitvoering van het testproces op een afzonderlijk te identificeren vrijgave van een testobject.
474. test data:	<i>Testgegevens:</i>		Gegevens die beschikbaar zijn (bijvoorbeeld in een database) voordat een test wordt uitgevoerd, en die de geteste component of het systeem beïnvloeden of er door beïnvloed worden.
475. Test data preparation tool:	<i>Testgegevens voorbereidingsgereedschap:</i>		Een type testgereedschap dat het mogelijk maakt dat gegevens geselecteerd worden uit een bestaande database of gecreëerd, gegenereerd, gemanipuleerd of gewijzigd worden voor gebruik in testen.
476. Test design:	<i>Test ontwerp:</i>		Zie test ontwerp specificatie.
477. Test design specificatie:	<i>Testontwerpspecificatie:</i>		Een document dat de testcondities (dekkingsgraadonderdelen) voor een testonderdeel en de gedetailleerde testaanpak specificeert en dat de gerelateerde logische testgevallen identificeert.
			<b>[Naar IEEE829].</b>
478. Test design technique:	<i>Testspecificatietechniek:</i>		Een procedure die gebruikt wordt om testgevallen af te leiden en / of te selecteren.
479. Test design tool:	<i>Testontwerpgereedschap:</i>		Een gereedschap dat ondersteuning biedt bij de testontwerpactiviteit door testinput te genereren op basis van een specificatie die mogelijk in een CASE-gereedschap is opgeslagen, bijv. een eisenbeheergereedschap, op basis van de gespecificeerde testcondities opgeslagen in het gereedschap zelf, of van de code.
480. Test driver:	<i>Testdriver:</i>		<b>Zie:</b> <i>driver – (driver).</i>
481. Test driven development:	<i>test gedreven ontwikkeling:</i>		Een manier om software te ontwikkelen waarbij de testgevallen worden ontwikkeld, en vaak geautomatiseerd, voordat de software is ontwikkeld om de testgevallen uit te voeren.
482. Test environment:	<i>Testomgeving:</i>		Een omgeving die hardware, instrumentatie, simulatoren, softwareprogramma's en andere ondersteunende elementen bevat die nodig zijn om een test uit te voeren.
			<b>[Naar IEEE 610].</b>
483. Test evaluation report:	<i>Testevaluatie rapport:</i>		Een document dat aan het eind van het testproces geproduceerd wordt en waarin alle testactiviteiten en -resultaten zijn samengevat. Het bevat ook een evaluatie van het testproces en de geleerde lessen.
484. Test execution:	<i>Testuitvoering:</i>		Het proces van het uitvoeren van een test op de geteste component of het systeem, waarbij werkelijke resultaten worden geproduceerd.
485. Test execution automation:	<i>Testuitvoering automatisering:</i>		Het gebruik van software, bijv. opname / weergave gereedschappen, om de uitvoering van testen, de vergelijking van de werkelijke uitkomsten en de verwachte resultaten, de opzet van precondities en voor andere testbeheer en rapportagefuncties te beheren.
486. Test execution phase:	<i>Testuitvoeringsfase:</i>		De periode binnen een softwareontwikkellevenscyclus waarin de componenten van een softwareproduct worden uitgevoerd en waarin het softwareproduct wordt geëvalueerd om vast te stellen of het voldoet aan de eisen.
			<b>[IEEE 610].</b>
487. Test execution schedule:	<i>Testdraaiboek:</i>		Een draaiboek voor de uitvoering van testprocedures. De testprocedures worden opgenomen in het testdraaiboek in de context en volgorde waarin ze uitgevoerd moeten worden.
488. Test execution technique:	<i>Testuitvoeringstechniek:</i>		De methode die gebruikt wordt om de test uit te voeren, handmatig of geautomatiseerd.
489. Test execution tool:	<i>Testuitvoeringsgereedschap:</i>		Een test gereedschap dat gebruikt kan worden om andere software uit te voeren, gebruikmakend van een geautomatiseerd testscript, bijvoorbeeld capture / playback.
			<b>[Fewster en Graham].</b>
490. Test fail:	<i>Testfout:</i>		<b>Zie:</b> <i>fail – (falen).</i>
491. Test generator:	<i>Testgenerator:</i>		<b>Zie:</b> <i>test data preparation tool – (testgegevens voorbereidingsgereedschap).</i>
492. Test leader:	<i>Testleider:</i>		<b>Zie:</b> <i>test manager – (testmanager).</i>
493. Test harness:	<i>Testraamwerk:</i>		Een testomgeving, bestaand uit stubs en drivers, die nodig is om een



		test uit te voeren.
494. Test incident:	<i>Testbevinding:</i>	<b>Zie:</b> <i>incident – (bevinding).</i>
495. Test incident report:	<i>Testbevindingenrapport:</i>	<b>Zie:</b> <i>incident report – (bevindingenrapport).</i>
496. Test infrastructure:	<i>Testinfrastructuur:</i>	De organisatorische benodigdheden voor het uitvoeren van een test, bestaand uit testomgevingen, testtools, werkplekken en procedures.
497. Test input:	<i>Testinvoer:</i>	De gegevens die tijdens de testuitvoering door het testobject ontvangen worden van een externe bron. De externe bron kan hardware, software of een persoon zijn.
498. Test item:	<i>Testonderdeel:</i>	Een individueel onderdeel dat getest moet worden. Normaal gesproken is er één testobject en zijn er meerdere testonderdelen. <b>Zie ook:</b> <i>test object – (testobject).</i>
499. Test item transmittal report:	<i>Testonderdeel overdrachtsrapportage:</i>	<b>Zie:</b> <i>release note – (opleveringsdocument).</i>
500. <i>Test leader:</i>	<i>Testleider:</i>	<i>Zie: test manager – (testmanager).</i>
501. Test level:	<i>Testsoort:</i>	Een groep testactiviteiten die gezamenlijk georganiseerd en beheerd worden. Een testsoort wordt gekoppeld aan de verantwoordelijkheden in een project. Voorbeelden van testsoorten zijn de componenttest, integratietest, systeemtest en acceptatietest. <b>[Naar TMap®].</b>
502. Test log:	<i>Testlogboek:</i>	Een chronologische vastlegging van relevante details over de testuitvoering. <b>[IEEE 829].</b>
503. Test logging:	<i>Testverslaglegging:</i>	Het proces van het vastleggen van details over de testuitvoering in een testlogboek.
504. Test manager:	<i>Testmanager:</i>	De persoon die verantwoordelijk is voor het projectbeheer van testactiviteiten en hulpbronnen, en het evalueren van een testobject. De persoon die de evaluatie van een testobject stuurt, beheerst, plant en bijstelt.
505. Test management:	<i>Testmanagement:</i>	Het plannen, begroten, monitoren en beheersen van testactiviteiten; dit zijn typisch werkzaamheden van een testmanager.
506. Test management tool:	<i>Testmanagementgereedschap:</i>	Een gereedschap dat hulp biedt bij het testmanagement en beheersen van een testproces. Het heeft vaak meerdere mogelijkheden, denk aan testware beheer, het plannen van tests, het vastleggen van resultaten, voortgangsbeheer, bevindingenbeheer en testrapportage.
507. Test Maturity Model (TMM):	<i>Test Maturity Model (TMM):</i>	Een model voor testprocesverbetering bestaand uit vijf getrapte niveaus, gerelateerd aan het Capability Maturity Model (CMM). Het model beschrijft de belangrijkste elementen van een effectief testproces.
508. Test monitoring:	<i>Testmonitoring:</i>	Een testbeheertaak die bestaat uit het periodiek bekijken van de status van een testproject. Rapporten worden samengesteld die bestaan uit het vergelijken van de werkelijke stand van zaken met de geplande. <b>Zie ook:</b> <i>test management – (testbeheer).</i>
509. Test object:	<i>Testobject:</i>	Het onderdeel of systeem dat getest moet worden. <b>Zie ook:</b> <i>test item – (testonderdeel).</i>
510. Test objective:	<i>Testdoel:</i>	De reden of doel voor het ontwerpen en uitvoeren van een test.
511. Test oracle:	<i>Testvraagbaak:</i>	De bron om de te verwachten resultaten te bepalen waarmee de actuele testresultaten vergeleken kunnen worden. De vraagbaak kan een bestaand systeem zijn (voor een benchmark), een gebruikershandboek, of de gespecialiseerde kennis van iemand, maar in geen geval de code. <b>[Naar Adrion].</b>
512. Test outcome:	<i>Testresultaat:</i>	<b>Zie:</b> <i>result – (resultaat).</i>
513. Test pass:	<i>Geslaagd:</i>	<b>Zie:</b> <i>pass – (geslaagd).</i>
514. Test performance indicator:	<i>Testvoortgangsindicatie:</i>	Een maat op hoog niveau met betrekking tot de effectiviteit en / of efficiëntie die wordt gebruikt om de verdere testontwikkeling te sturen. B.v. Defect Detection Percentage (DDP).
515. Test phase:	<i>Testfase:</i>	Een afgebakende verzameling van bij elkaar horende testactiviteiten

		samengevoegd tot een beheersbare fase in een project bijvoorbeeld de uitvoering van activiteiten binnen een testsoort. <b>[Naar Gerrard].</b>
516. Test plan:	<i>Testplan:</i>	Een document dat de afbakening, de aanpak, de hulpmiddelen en de planning van de testactiviteiten beschrijft. Het beschrijft o.a. de testelementen, de te testen aspecten, de testtaken, wie welke taak uit zal voeren, niveau van onafhankelijkheid van de tester, de testomgeving, de testspecificatietechnieken en de ingangs- en uitgangscriteria en de beweegredenen voor die keuze, en de risico's die noodscenario's behoeven. Het is het eindresultaat van het testplanningproces. <b>[Naar IEEE 829].</b>
517. Test planning:	Testplanning:	De activiteit waarin een testplan wordt opgesteld of bijgewerkt.
518. Test policy:	Testbeleid:	Een hoog niveau document waarin is beschreven welke principes worden gehanteerd, voor welke aanpak is gekozen, en wat de belangrijkste afspraken zijn binnen de organisatie met betrekking tot het testen.
519. Test Point Analysis (TPA):	TestPunt Analyse (TPA):	Een op rekenkundige wijze gebaseerde methode gebaseerd op functiepunanalyse. <b>[TMap®].</b>
520. Test procedure:	Testprocedure:	<b>Zie:</b> <i>test procedure specification – (testprocedurespecificatie).</i>
521. Test procedure specification:	Testprocedurespecificatie:	Een document waarin opgenomen de volgorde waarin activiteiten van een test uitgevoerd dienen te worden. Ook bekend onder de naam testscript of handmatig testscript. <b>[Naar IEEE829].</b>
522. Test process:	Testproces:	Een fundamenteel testproces omvat alle activiteiten voor uitvoering, vastlegging, voorgangscntrole en afronding. <b>[Naar BS 7925/2].</b>
523. Test Process Improvement (TPI):	TestProcesVerbetering (TPI):	Een continu raamwerk voor het verbeteren van het testproces waarin de belangrijkste elementen van een effectief testproces worden beschreven speciaal toegespitst op systeem en acceptatie testen.
524. Test record:	<i>Testlogboek:</i>	<b>Zie:</b> <i>testlog – (testlogboek).</i>
525. Test recording:	<i>Testverslaglegging</i> :	<b>Zie:</b> <i>test logging – (testvastlegging).</i>
526. Test reproduceability:	<i>Testreproduceerbaarheid:</i>	Een eigenschap van een test die aangeeft of iedere keer dat de test wordt uitgevoerd hetzelfde resultaat wordt geproduceerd.
527. Test report:	<i>Testrapport:</i>	<b>Zie:</b> <i>test summary report – (testsamenvattingsrapport).</i>
528. Test requirement:	<i>Testeis:</i>	<b>Zie:</b> <i>test condition – (tesconditie).</i>
529. Test run:	<i>Testuitvoering:</i>	Uitvoering van een test op een specifieke versie van een testobject.
530. Test run log:	<i>Testuitvoeringslogboek:</i>	<b>Zie:</b> <i>test log – (testlogboek).</i>
531. Test result:	<i>Testresultaat:</i>	<b>Zie:</b> <i>result – (resultaat).</i>
532. Test scenario:	<i>Testscenario:</i>	<b>Zie:</b> <i>test procedure specification – (testprocedurespecificatie).</i>
533. Test script:	<i>Testscript:</i>	Wordt gewoonlijk gebruikt om te refereren naar een testprocedure specificatie, meestal een geautomatiseerde test procedure. <b>Zie:</b> <i>test suite – (testsuite).</i>
534. Testset:	<i>Testset:</i>	
535. Test situation:	<i>Testsituatie</i>	Zie Test condition
536. Test specification:	<i>Testspecificatie of testontwerp:</i>	Een document waarin de testspecificaties, de testgevallen en / of de testprocedures beschreven worden.
537. Test specification technique:	<i>Testspecificatietechniek:</i>	<b>Zie:</b> <i>test design technique – (testspecificatietechniek).</i>
538. Test stage:	<i>Testfase:</i>	<b>Zie:</b> <i>test level – (testsoort).</i>
539. Test strategy:	<i>Teststrategie:</i>	Een beschrijving op hoog niveau van de testsoorten die uitgevoerd moeten worden alsmede hoe de tests binnen de testsoort moet worden uitgevoerd voor een organisatie of een programma (wanneer sprake is van één of meerdere projecten).
540. Test suite:	<i>Testset:</i>	Een verzameling testgevallen die op een component of te testen systeem worden losgelaten, waarbij de eindstatus van de ene test

541. Test summary report:	<i>Samenvatting test rapport:</i>	<p>vaak gebruikt wordt als startconditie voor een volgende test.</p> <p>Een document waarin de testactiviteiten en resultaten zijn samengevat. Het bevat ook de evaluatie van de testgevallen ten opzichte van de exitcriteria.</p> <p><b>[Naar IEEE 829].</b></p>
542. Test target:	<i>Testdoelstelling:</i>	Een set uitgangscriteria.
543. <i>Test technique:</i>	<i>Testtechniek:</i>	<b>Zie:</b> <i>test design technique – (testspecificatietechniek).</i>
544. Test tool:	<i>Testgereedschap:</i>	<p>Een computerprogramma dat een of meer testactiviteiten ondersteunt zoals planning en beheer, het specificeren, het opbouwen van initiële bestanden en gegevens, het uitvoeren van de test en testanalyse.</p> <p><b>[TMap®]</b></p> <p><b>Zie ook:</b> <i>CAST – (CAST).</i></p>
545. Test type:	<i>Testtype:</i>	<p>Een verzameling testactiviteiten met als doelstelling het testen van een component of systeem op een of meer gerelateerde kwaliteitsattributen. Een testtype wordt meestal toegespitst op een specifiek testdoel zoals betrouwbaarheid, bruikbaarheid, regressie enz., en kan plaatshebben binnen meerdere testsoorten of testfasen.</p> <p><b>[Naar TMap®].</b></p>
546. Testability:	<i>Testbaarheid:</i>	<p>Het vermogen van het softwareproduct om gewijzigde delen uit te laten testen.</p> <p><b>[ISO9126].</b></p> <p><b>Zie ook:</b> <i>maintainability – (onderhoudbaarheid).</i></p>
547. Testability review:	<i>Testbaarheidsreview:</i>	<p>Een uitgebreide controle op de testbasis om te bepalen of de testbasis van een adequaat kwaliteitsniveau is om als een invoerdocument voor het testproces te dienen.</p> <p><b>[Naar TMap®].</b></p>
548. Testable requirements:	<i>Testbare eisen:</i>	<p>De mate waarin een eis is opgesteld in termen die het mogelijk maken een testontwerp (en de daarop volgende testgevallen) op te stellen en tests uit te voeren, om vast te stellen dat inderdaad aan de eis is voldaan. <b>[Naar IEEE610].</b></p>
549. Tester:	<i>Tester:</i>	Een toegeruste professional die zich bezighoudt met het testen van een component of systeem.
550. Testing:	<i>Testen:</i>	Het proces bestaande uit alle levenscyclusactiviteiten, zowel statisch als dynamisch, die te maken hebben met planning, voorbereiding en evaluatie van softwareproducten en aanverwante zaken om aan te geven dat ze aan de gespecificeerde eisen voldoen, om aan te tonen dat wordt voldaan aan de doelstelling en om omissies op te sporen.
551. Testware	<i>Testware</i>	<p>Producten die gedurende het testen worden vervaardigd benodigd voor het plannen, ontwerpen, en het uitvoeren van tests zoals documentatie, scripts, invoer, verwachte resultaten, opzet en afbouw procedures, bestanden, databases, omgeving en extra software of hulpprogramma's gebruikt tijdens het testen.</p> <p><b>[Naar Fewster en Graham].</b></p>
552. Thread testing:	<i>Procestesten:</i>	<p>Een variant van een component integratie test waarbij de gestaag vorderende integratie van componenten volgt op de implementatie van onderdelen van de eisen in tegenstelling tot de integratie van componenten op hiërarchische basis.</p> <p><b>Zie:</b> <i>performance – (performance).</i></p>
553. Time behaviour:	<i>Tijdgedrag:</i>	
554. top-down testing:	<i>Top-down testen:</i>	<p>Een incrementele benadering van de integratietest, waar de component aan de bovenkant van de componentenhiërarchie als eerste wordt getest, terwijl componenten op een lager niveau door een stub worden gesimuleerd. De geteste componenten worden dan gebruikt om componenten op een lager niveau te testen. Het proces wordt herhaald tot de componenten op het laagste niveau zijn getest.</p> <p><b>Zie ook:</b> <i>integration testing – (integratietesten).</i></p>
555. Traceability:	<i>Traceerbaarheid:</i>	<p>Het vermogen om gerelateerde onderdelen in documentatie en software, zoals eisen met bijbehorende tests te kunnen identificeren.</p> <p><b>Zie ook:</b> <i>horizontal traceability, vertical tracibility – (horizontale traceerbaarheid, verticale traceerbaarheid).</i></p>

## U

556. Understandability:	<i>Begrijpelijkheid:</i>	Het vermogen van het softwareproduct om de gebruiker in staat te stellen om te begrijpen of de software geschikt is, en hoe het gebruikt kan worden voor de uitvoering van specifieke taken en condities <b>[ISO 9126]</b> <b>Zie ook:</b> <i>usability – (bruikbaarheid).</i>
557. Unit:	<i>Programma:</i>	<b>Zie:</b> <i>component – (component).</i>
558. Unit testing:	<i>Programmatest:</i>	<b>Zie:</b> <i>component testing – (componenttest).</i>
559. Unreachable code:	<i>Onbereikbare code:</i>	Code die niet aangeroepen wordt en dus ook niet kan worden uitgevoerd.
560. Usability:	<i>Bruikbaarheid:</i>	Het vermogen van de software om door de gebruiker begrepen, eenvoudig te leren, te gebruiken en aantrekkelijk te zijn binnen de gespecificeerde omstandigheden. <b>[ISO 9126].</b>
561. Usability testing:	<i>Bruikbaarheidstest:</i>	Testen om de mate te bepalen waarin de gebruikers het softwareproduct begrijpen, gemakkelijk leren, gemakkelijk mee te werken en aantrekkelijk vinden in de gespecificeerde omstandigheden. [Na ISO 9126]
562. Use case:	<i>Use case:</i>	Een opeenvolging van transacties in een dialoog tussen een gebruiker en het systeem met een tastbaar resultaat.
563. Use case testing:	<i>Use case testen:</i>	Een black box testspecificatietechniek waarin de testgevallen worden ontworpen om gebruikersscenario's uit te voeren. <b>Zie:</b> <i>acceptance testing – (acceptatie testen).</i>
564. User acceptance testing:	<i>Gebruikersacceptatietest:</i>	<b>Zie:</b> <i>use case testing – (use case testen).</i>
565. user scenario testing:	<i>Gebruikersscenario testen:</i>	
566. User test:	<i>Gebruikerstest:</i>	Een test waarbij echte gebruikers betrokken zijn om de bruikbaarheid van een component of een systeem te beoordelen.

## V

567. V-model:	<i>V-model:</i>	Een kader om de activiteiten van de softwareontwikkellevenscyclus van specificatie van eisen tot en met onderhoud te beschrijven. Het V-Model illustreert hoe de testactiviteiten in elke fase van de softwareontwikkellevenscyclus kunnen worden geïntegreerd.
568. Validation:	<i>Validatie:</i>	Bewijs door onderzoek en door aanleveren van objectief bewijsmateriaal dat aan de eisen ten aanzien van een specifiek voorgenomen gebruik of van een toepassing is voldaan. <b>[ISO 9000].</b>
569. Variable:	<i>Variabele:</i>	Een opgeslagen element in een computer die door een softwareprogramma toegankelijk is door naar een naam te verwijzen.
570. Verification:	<i>Verificatie:</i>	Bewijs door onderzoek en door aanleveren van objectief bewijsmateriaal dat aan de gespecificeerde eisen is voldaan. <b>[ISO 9000].</b>
571. Vertical traceability:	<i>Verticale traceerbaarheid:</i>	Het traceren van eisen door de niveaus van ontwikkelingsdocumentatie heen tot aan de componenten.
572. Version control:	<i>Versiebeheer:</i>	<b>Zie:</b> <i>configuration control – (configuratiebeheer).</i>
573. Volume testing:	<i>Volumetest:</i>	Testen waarbij het systeem met grote volumes gegevens wordt belast. <b>Zie ook:</b> <i>resource utilization testing – (zuinigheidstesten).</i>

## W

574. Walkthrough:	<i>Doorlopen:</i>	Een stapsgewijze presentatie van een document door de auteur om informatie te verzamelen en een gemeenschappelijk begrip over de inhoud van het document te krijgen. <b>[Freedman en Weinberg, IEEE 1028 ]</b>
-------------------	-------------------	---

575. White-box test design technique:	<i>White box testspecificatietechniek:</i>	<b>Zie ook:</b> <i>peer review – (collegiale toetsing).</i> Procedure om testgevallen af te leiden en / of te selecteren gebaseerd op een analyse van de interne structuur van een component of een systeem.
576. White-box testing:	<i>White box testen:</i>	Testen gebaseerd op de analyse van de interne structuur van de component of het systeem.
577. Wide Band Delphi:	<i>Wide Band Delphi:</i>	Een testbegrotingstechniek die tot doel heeft om met behulp van de collectieve ervaring van de teamleden een nauwkeurige begroting te maken.

## Bijlage A (Informatief)

Index van bronnen; de volgende non-normatieve bronnen zijn gebruikt bij het samenstellen van deze verklarende woordenlijst:

[Abbott] J. Abbot (1986), *Software Testing Techniques*, NCC Publications.

[Adrion] W. Adrion, M. Branstad and J. Cherniabsky (1982), Validation, Verification and Testing of Computer Software, in: *Computing Surveys*, Vol. 14, No 2, June 1982.

[Bach] J. Bach (2004), Exploratory Testing, in: E. van Veenendaal, *The Testing Practitioner – 2 nd edition*, UTN Publishing, ISBN 90-72194-65-9.

[Beizer] B. Beizer (1990), *Software Testing Techniques*, van Nostrand Reinhold, ISBN 0-442-20672-0

[Chow] T. Chow (1978), *Testing Software Design Modelled by Finite-Sate Machines*, in: *IEEE Transactions on Software Engineering*, Vol. 4, No 3, May 1978.

[CMM] M. Paulk, C. Weber, B. Curtis and M.B. Chrissis (1995), *The Capability Maturity Model, Guidelines for Improving the Software Process*, Addison-Wesley, ISBN 0-201- 54664-7

[CMMI] M.B. Chrissis, M. Konrad and S. Shrum (2004), *CMMI, Guidelines for Process Integration and Product Improvement*, Addison Wesley, ISBN 0-321-15496-7

[Fenton] N. Fenton (1991), *Software Metrics: a Rigorous Approach*, Chapman & Hall, ISBN 0-53249-425-1

[Fewster and Graham] M. Fewster and D. Graham (1999), *Software Test Automation, Effective use of test execution tools*, Addison-Wesley, ISBN 0-201-33140-3.

[Freedman and Weinberg] D. Freedman and G. Weinberg (1990), *Walkthroughs, Inspections and Technical Reviews*, Dorset House Publishing, ISBN 0-932633-19-6.

[Gerrard] P. Gerrard and N. Thompson (2002), *Risk-Based E-Business Testing*, Artech House Publishers, ISBN 1-58053-314-0.

[Gilb and Graham] T. Gilb and D. Graham (1993), *Software Inspection*, Addison-Wesley, ISBN 0-201-63181-4.

[Grochtmann] M. Grochtmann (1994), *Test Case Design Using Classification Trees*, in: Conference Proceedings STAR 1994.

[Hetzel] W. Hetzel (1988), *The complete guide to software testing – 2 nd edition*, QED Information Sciences, ISBN 0-89435-242-3.

[McCabe] T. McCabe (1976), *A complexity measure*, in: *IEEE Transactions on Software Engineering*, Vol. 2, pp. 308-320.

[Musa] J. Musa (1998), *Software Reliability Engineering Testing*, McGraw-Hill Education, ISBN 0-07913-271-5.

[Myers] G. Myers (1979), *The Art of Software Testing*, Wiley, ISBN 0-471-04328-1.

[TMap®] M. Pol, R. Teunissen, E. van Veenendaal (2002), *Software Testing, A guide to the TMap Approach*, Addison Wesley, ISBN 0-201-745712. [Veenendaal] E. van Veenendaal (2004), *The Testing Practitioner – 2 nd edition*, UTN Publishing, ISBN 90-72194-65-9.

## **Bijlage B (Methode om commentaar op deze woordenlijst aan te leveren)**

Geef vooral commentaar op dit document, zodat deze woordenlijst verder verbeterd kan worden om te voldoen aan de testgemeenschap.

Wanneer er commentaar wordt aangeleverd, zorg ervoor dat de volgende informatie aanwezig is:

- U naam en contactgegevens;
- Het versienummer van de woordenlijst (huidige versie 1.2);
- Exacte gedeelte van de woordenlijst;
- Ondersteunende informatie, zoals de reden waarom een wijziging gewenst is, of de referentie naar het gebruik van een definitie.

U kunt commentaar op verschillende wijzes aanleveren, welke in gewenste volgorde zijn:

1. Email: [eve@improveqs.nl](mailto:eve@improveqs.nl);
2. Post: Improve Quality Services BV, t.a.v. Mr. E. van Veenendaal, Waalreweg 39, 5554 HA, Valkenswaard, Nederland;
3. FAX: +31 40 20 21450, t.a.v. Mr. E. van Veenendaal.