Mastertestplan

**{Productnaam, versie}**

Rubriceringsniveau {Rubriceringsniveau}

Versie {Versienummer}, {Datum}

Inhoudsopgave

##### Colofon

###### Rubricering

Rubricering conform [VIRBI 2013, art. 4](https://wetten.overheid.nl/BWBR0033507/2013-06-01#Artikel4).

{Verwijder deze paragraaf en de rubricering op de titelpagina als rubricering niet van toepassing is}

|  |  |  |
| --- | --- | --- |
| Rubriceringsniveau | Rubriceringsduur | Vaststeller |
| {Rubriceringsniveau} | {Rubriceringsduur} | {Vaststeller van de rubricering: minister, staatssecretaris, secretaris-generaal of een door de secretaris-generaal aangewezen rubriceringsambtenaar} |

###### Goedkeuring

|  |  |  |
| --- | --- | --- |
| Versie | Datum goedkeuring | Goedgekeurd door |
| {versie} | {datum} | {naam} |

###### Revisiehistorie

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Versie | Datum | Status | Auteur | Reviewers | Opmerkingen |
| {versie} | {datum} | {status} | {naam} | {namen} | {opmerkingen} |

###### Betrokkenen bij dit document

{Neem in onderstaande tabel de auteurs, reviewers en goedkeurders van dit document op}

|  |  |  |
| --- | --- | --- |
| Organisatie | Functie/rol | Naam |
| {opdrachtgevende organisatie} | Opdrachtgever | {naam} |
| {opdrachtgevende organisatie} | Projectleider | {naam} |
| {opdrachtgevende organisatie} | Product owner | {naam} |
| {opdrachtgevende organisatie} | Kwaliteitsmanager | {naam} |
| {beheerorganisatie} | Projectleider | {naam} |
| {beheerorganisatie} | Kwaliteitsmanager | {naam} |
| ICTU | Projectleider | {naam} |
| ICTU | Software delivery manager | {naam} |
| ICTU | Kwaliteitsmanager | {naam} |

###### Template versie

Versie wip, 02-04-2025

Verbeterpunten t.a.v. deze template graag melden via [GitHub](https://github.com/ICTU/Kwaliteitsaanpak/issues).

# Managementsamenvatting

{De hoofdboodschap van dit document.}

# Inleiding

## Over dit document

Het doel van het mastertestplan is om betrokkenen bij het testproces te informeren over de strategie, aanpak, activiteiten, inclusief de onderlinge relaties en afhankelijkheden, en de op te leveren producten met betrekking tot het testtraject.

Er {zijn/worden} ook detailtestplannen opgesteld voor de testsoorten die tijdens de realisatiefase worden uitgevoerd door het project of onder verantwoordelijkheid van het project door een derde partij. Deze detailtestplannen {zijn/worden} afgeleid van dit mastertestplan.

Het template dat ten grondslag ligt aan dit document is mede gebaseerd op de Nederlandse testmethodiek TMap Next en de internationale testmethodiek ISTQB.

## Doelgroep

Dit document is bedoeld voor leden van het project en voor betrokken medewerkers buiten het project van opdrachtgever, beheerorganisatie en ontwikkelpartij.

## Kaders

De volgende kaders zijn van toepassing op het projectresultaat:

|  |  |
| --- | --- |
| Volgnummer | Kader |
| K01 | NEN-ISO/IEC 27001:2017 en NEN-ISO/IEC 27002:2017, VIR 2007, VIRBI 2013 en BIO voor het inrichten en beheren van informatiebeveiliging in brede zin. |
| K02 | NCSC ICT-beveiligingsrichtlijnen voor webapplicaties |
| K03 | OWASP Top-10 |
| K04 | ISO 9241-210:2019 Ergonomics of human-system interaction - Part 210: Human-centred design for interactive systems |
| K05 | WCAG2.2 (Web Content Accessibility Guidelines) voor eisen met betrekking tot toegankelijkheid |
| K06 | NEN-ISO/IEC 25010:2023 voor het specificeren van productkwaliteit |
| K07 | NORA - Referentiearchitectuur voor de Nederlandse Overheid |

## Uitgangspunten

De volgende uitgangspunten zijn van toepassing op dit document:

|  |  |
| --- | --- |
| Volgnummer | Uitgangspunt |
| U01 | De realisatie van de software wordt door {ontwikkelorganisatie} uitgevoerd. Het functioneel beheer wordt door {functioneel-beheerorganisatie} uitgevoerd. Het technisch beheer wordt door {technisch-beheerorganisatie} uitgevoerd. |
| U02 | De als testbasis geïdentificeerde documenten zijn door alle acceptanten, inclusief het testteam, geaccordeerd, alvorens met de testspecificatie is begonnen. |
| U03 | Per release of per sprint kan de opdrachtgever en/of product owner besluiten om bepaalde functionaliteit, bijvoorbeeld geleverd door externe partijen, niet te testen. Indien dit voorkomt, dan zal dit expliciet worden opgenomen in de managementsamenvatting van het vrijgaveadvies. |
| U04 | De multidisciplinaire samenstelling van de Scrumteams - testers en ontwikkelaars zitten in hetzelfde team - geeft mogelijkheid tot snelle interactie. Issues gevonden binnen een sprint kunnen hierdoor vaak nog binnen dezelfde sprint worden opgelost en worden in dat geval niet apart geadministreerd. |
| U05 | Issues gevonden tijdens de acceptatietests of in productie worden op de product backlog verwerkt samen met de user stories. |
| U06 | Testomgevingen worden conform planning tijdig en correct werkend opgeleverd. |
| U07 | Er is voldoende en gepaste testdata ter beschikking. Zie de paragraaf 'Testdata' in het hoofdstuk 'Infrastructuur' voor een beschrijving van de testdata. |
| U08 | De testspecificatie voor een testsoort start pas als aan de entry-criteria (Definition of Ready), die hiervoor gelden, is voldaan. Hieraan is voldaan voor het deel (van de testbasis) dat in de betreffende sprint gerealiseerd en getest wordt (bijvoorbeeld een aantal bij elkaar horende user stories). |
| U09 | De gevonden en openstaande bevindingen worden structureel besproken in het bevindingenoverleg. Als er veel bevindingen zijn of meer ernstige bevindingen dan verwacht, dan wordt bijgestuurd op de kwaliteit van het ontwikkeltraject. Dat heeft ook impact op de planning en begroting. |
| U10 | Nieuwe functionaliteit wordt getest en de tests worden opgenomen in de geautomatiseerde regressietest. |
| U11 | {Er vindt een ketentest plaats in samenwerking met de bij de gekoppelde systemen betrokkenen organisaties.} |
| {volgnummer} | {uitgangspunt} |

## Relatie met andere documenten

Het mastertestplan is gebaseerd op de volgende documenten, die beschrijven welke eisen en wensen aan de oplossing zijn gesteld en hoe de oplossing werkt {Selecteer de van toepassing zijnde documenten}:

* Projectstartarchitectuur (PSA), {documentreferentie},
* Business impact analyse (BIA), {documentreferentie},
* Data protection impact assessment (DPIA), {documentreferentie},
* Softwarearchitectuurdocument (SAD), {documentreferentie},
* Infrastructuurarchitectuur (IA), {documentreferentie},
* Informatiebeveiligingsplan (IB-plan), {documentreferentie},
* Kwaliteitsplan, {documentreferentie},
* Niet-functionele eisen (NFE), {documentreferentie},
* Globaal functioneel ontwerp (GFO), {documentreferentie},
* Interactie-ontwerp (UX), {documentreferentie},
* Product backlog, {documentreferentie},
* Vastgesteld minimal viable product, {documentreferentie},
* Wireframe, mockup, prototype, animatie, {documentreferentie}.

Daarnaast is ten behoeve van het mastertestplan een productrisicoanalyse uitgevoerd. Zie de paragraaf 'Resultaat productrisicoanalyse'.

{Beschrijf ook een eventuele relatie met andere documenten}

## Leeswijzer

Hoofdstuk 3 beschrijft de testopdracht. Hoofdstuk 4 beschrijft de resultaten van de productrisicoanalyse en de resulterende teststrategie. Hoofdstuk 5 bespreekt de aanpak van de uit te voeren testen. Hoofdstuk beschrijft de testorganisatie. Hoofdstuk 7 bevat een overzicht van de testinfrastructuur. Hoofdstuk 8 beschrijft het beheer van de testactiviteiten en testproducten. Hoofdstuk 9 geeft een overzicht van de belangrijkste risico's voor het testproces. Hoofdstuk 10, tenslotte, geeft een globale planning van de testactiviteiten.

Bijlage A bevat afkortingen en termen die voorkomen in de ICTU Kwaliteitsaanpak Softwareontwikkeling en bijbehorende templates. Bijlage B verwijst naar regelmatig gebruikte bronnen. Bijlage C bevat een beknopte samenvatting van de ICTU Kwaliteitsaanpak Softwareontwikkeling.

# Opdrachtformulering voor de uit te voeren tests

## Doel van het testen

Het doel van de uit te voeren tests is dat {het product} wordt ontwikkeld en in gebruik genomen zonder onacceptabele risico's voor de gebruikers en voor de betrokken organisaties en hun medewerkers. Hiertoe worden tests gepland, uitgevoerd, gedocumenteerd en geaccepteerd. De testaanpak moet duidelijk maken wat de verschillende verantwoordelijkheden zijn van de verschillende partijen en disciplines die betrokken zijn bij het integrale testtraject.

De resultaten van de tests zijn de basis voor het vrijgaveadvies, op basis waarvan een besluit over ingebruikname van de betreffende release kan worden genomen.

## Context

{Beschrijf kort de context waarin de applicatie zal opereren, voor zover deze van belang is voor de uit te voeren tests. Denk aan de speciale eisen die worden gesteld aan de verschillende onderdelen van het informatiesysteem: het “samenhangend geheel van gegevensverzamelingen en de daarbij behorende personen, procedures, processen en programmatuur alsmede de voor het informatiesysteem getroffen voorzieningen voor opslag, verwerking en communicatie” (zie bijlage A).}

{Maak gebruik van beschrijvingen in bestaande documenten (zie paragraaf 2.5).}

## Scope van de tests

### Binnen scope

De volgende onderdelen van {het product} zijn binnen de scope van de tests: {Deel de onderdelen eventueel in per deelsysteem}

* software
* gegevensverzamelingen
* dataconversies
* interfaces (API’s) met: {de systemen waarmee informatie wordt uitgewisseld}
* infrastructurele voorzieningen
* ondersteunde processen
* {vul aan indien nodig}

{het product} wordt op de volgende kwaliteitskenmerken getest: {Maak gebruik van de kwaliteitskarakteristieken van de norm ISO/IEC-25010:2023. Of vat de acceptatiecriteria samen die eventueel al zijn beschreven in het document Niet-functionele eisen.}

* functionele geschiktheid
* prestatie-efficiëntie (performance)
* uitwisselbaarheid (compatibility)
* bruikbaarheid (usability, inclusief toegankelijkheid)
* betrouwbaarheid
* beveiligbaarheid
* onderhoudbaarheid
* overdraagbaarheid
* {vul aan indien nodig}

Voor de combinaties van de onderdelen en de kwaliteitskenmerken worden in de productrisicoanalyse de risico’s bepaald (zie paragraaf 4.1).

### Buiten scope

Buiten de scope van de tests zijn:

* wijzigingen die geen onderdeel zijn van het project, zoals die ten aanzien van de infrastructuur, de systeemsoftware, de systemen waarmee informatie wordt uitgewisseld
* mogelijk toekomstige projecten die op het huidige project van invloed zijn
* {vul aan indien nodig}

{Noem eventueel ook deelsystemen als daarover onduidelijkheid kan bestaan.}

## Acceptanten

In onderstaande tabel is aangegeven wie de verantwoordelijken zijn voor de acceptatie van {het product}. Zij zijn de acceptanten, die akkoord moeten gaan met de resultaten van de tests voor de verschillende kwaliteitskenmerken en onderdelen van het product.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Naam | Organisatie | Functie | Accepteert deze kwaliteitskenmerken | Accepteert deze onderdelen |
| {naam} | {opdrachtgevende organisatie} | {projectleider} | {kenmerk} | {onderdeel} |
| {naam} | {opdrachtgevende organisatie} | {product owner} | {kenmerk} | {onderdeel} |
| {naam} | {organisatie} | {functie} | {kenmerk} | {onderdeel} |
| {naam} | {organisatie} | {functie} | {kenmerk} | {onderdeel} |
| {naam} | {organisatie} | {functie} | {kenmerk} | {onderdeel} |

# Productrisicoanalyse en teststrategie

De beschikbare tijd om te testen is beperkt; niet alles kan even zwaar worden getest. Dus moesten er keuzes worden gemaakt. Daarbij is ernaar gestreefd om de testcapaciteit zo effectief en efficiënt mogelijk over het totale testtraject te verdelen. Dit is vastgelegd in de teststrategie.

De teststrategie is gebaseerd op een risicoanalyse: {het product} moet zodanig voldoen in de praktijk dat er geen onacceptabele risico's voor de organisatie uit voortvloeien. Daar waar de oplevering van het systeem veel risico's met zich meebrengt, is uitgebreid testen op zijn plaats; het omgekeerde is aan de andere kant van het spectrum ook waar: 'no risk, no test'.

De eerste stap bij het vaststellen van de teststrategie was daarom het uitvoeren van een productrisicoanalyse. Hierin is het te testen product geanalyseerd, met als doel dat de testmanager en de verschillende andere belanghebbenden tot een gezamenlijk beeld kwamen van wat de meer of minder risicovolle kwaliteitskenmerken en delen van het te testen product zijn, zodat de grondigheid van testen hieraan gerelateerd kan worden. De uitvoering en resultaten van de productrisicoanalyse zijn in paragraaf 4.1 beschreven.

De teststrategie bouwt voort op de resultaten van de risicoanalyse: hier worden keuzes gemaakt over welke onderdelen van {het product} getest gaan worden, op welke manieren en op welke momenten. Paragraaf 4.2 beschrijft de teststrategie.

## Resultaat productrisicoanalyse

De acceptanten {optioneel: en andere bij het project betrokkenen} hebben gezamenlijk de productrisico’s vastgesteld. Deze zijn bepalend voor de zwaarte van de uit te voeren tests. Deze productrisicoanalyse (PRA) bestond uit twee stappen:

1. Inventariseren van de van belang zijnde risico’s.

2. Classificeren van de risico’s. De risicoklasse is op de volgende manier afgeleid van de faalkans (hoe groot is de kans dat het fout gaat?) en de schade (voor de organisatie als het inderdaad fout gaat): {beschrijving}

{Tijdens/na} de PRA zijn de omschrijvingen van de risico’s vertaald in kwaliteitskenmerken. Op basis van een kenmerk kunnen in de teststrategie de toe te passen testsoorten worden bepaald. De volledige productrisicoanalyse is beschreven in {documentreferentie}.

Onderstaande tabel bevat voor de verschillende onderdelen en kwaliteitskenmerken van {het product} de in de PRA vastgestelde risico’s en risicoklassen (A = hoog, B = middelhoog, C = laag).

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Volgnummer | Onderdeel product | Kwaliteitskenmerk van het product | Omschrijving risico | Risicoklasse |
| R01 | {onderdeel} | {kwaliteitskenmerk} | {omschrijving} | {A, B of C} |
| R02 | {onderdeel} | {kwaliteitskenmerk} | {omschrijving} | {A, B of C} |
| R03 | {onderdeel} | {kwaliteitskenmerk} | {omschrijving} | {A, B of C} |

## Teststrategie

In de teststrategie is per onderdeel van {het product} bepaald welke testsoorten met welke zwaarte worden toegepast. Hierbij is de risicoklasse bepalend voor de zwaarte van de test. De teststrategie is er bovendien op gericht om de risico’s met de hoogste risicoklasse zo vroeg mogelijk in het testtraject af te dekken.

### Testsoorten en testvormen

Onderstaande tabel bevat de te plannen testsoorten met hun definities. Ook zijn de onderkende testvormen genoemd; deze worden in het volgende hoofdstuk toegelicht.

Een testsoort is een concreet uit te voeren test waarmee een of meer testdoelen worden gerealiseerd. Binnen een testsoort kunnen verschillende testvormen voorkomen; een testvorm is bedoeld voor het testen op een specifiek kenmerk van {het product}.

|  |  |  |
| --- | --- | --- |
| Testsoort | Definitie | Testvormen |
| Unittest (UT) | Een door de ontwikkelaar tijdens de realisatie uitgevoerde test, die moet aantonen dat het ontwikkelde onderdeel aan de functionele en niet-functionele specificaties en het technisch ontwerp voldoet. | Functionele test |
| Unit-integratietest (UIT) | Een door de ontwikkelaar tijdens de realisatie uitgevoerde test, die moet aantonen dat meerdere onderdelen samen aan de functionele en niet-functionele specificaties en het technisch ontwerp voldoen. | Functionele test |
| Systeemtest (ST) | Een door het ontwikkelteam in een (goed beheersbare) laboratoriumomgeving uitgevoerde test, die moet aantonen dat het ontwikkelde systeem of delen daarvan aan de functionele- en niet-functionele specificaties voldoen. | Functionele test, securitytest, volledigheidstest, regressietest |
| Functionele acceptatietest (FAT) | Een door een vertegenwoordiger van de opdrachtgever in een acceptatieomgeving uitgevoerde test, die moet aantonen dat het ontwikkelde systeem aan de functionele acceptatiecriteria voldoet. | Functionele test, volledigheidstest |
| Gebruikers-acceptatietest (GAT) | Een door de beoogde gebruikers in een acceptatieomgeving uitgevoerde test, die moet aantonen dat zij met de geleverde applicatie hun werk correct en volledig kunnen doen. | Functionele test |
| Penetratietest (PEN) | Een door testers in een acceptatieomgeving uitgevoerde test, die moet aantonen dat het ontwikkelde systeem aan de beveiligingscriteria voldoet. | Black box penetratietest, grey box penetratietest, white box penetratietest |
| Performancetest (PERF) | Een door testers in een productie-like omgeving uitgevoerde test die worden uitgevoerd om te bepalen hoe een systeem presteert in termen van reactievermogen en stabiliteit onder een bepaalde werklast. | Performance loadtest, performance stresstest, performance duurtest |
| Gebruikerstevreden-heidstest (GEBR) | Een gebruikerstevredenheidstest is een test om de functionaliteit van het systeem te testen door echte gebruikers taken te laten uitvoeren en te observeren hóe ze dat doen. | Usabilitytest,enquête |
| Toegankelijkheidstest (TOEG) | Een test waarbij de functionaliteit en toegankelijkheid van het systeem getest wordt, aan de hand van de webrichtlijnen WCAG 2.1 |  |
| Productie-acceptatietest (PAT) | Een in een productie-like omgeving uitgevoerde test, die moet aantonen dat het ontwikkelde systeem aan de voor de (toekomstige) beheerorganisatie relevante acceptatiecriteria voldoet. | Functionele test, installatie- en rollbacktest, hersteltest, backup- en restoretest |
| Conformancetest (CONF) | Een test waarmee de beheerder van een stelsel kan toetsen of de oplossing van een aansluitende partij (authenticatiedienst, machtigingsdienst of dienstverlener) technisch en functioneel werkt volgens de gestelde eisen) | Afhankelijk van het stelsel |

Naast deze testsoorten voorziet het project de volgende toetsen (toetsingen/reviews van documenten uit de voorfase en realisatiefase):

* {review architectuur}
* {review ontwikkelomgeving}
* {vul aan indien relevant}

### Strategie voor de verschillende testsoorten

Afhankelijk van de risicoklasse (RK) is de beoogde teststrategie bepaald, dat wil zeggen: welke testsoorten met welke zwaarte per onderdeel van {het product} voor ieder kenmerk moeten worden uitgevoerd. Dit is samengevat in onderstaande tabellen.

De mogelijke testzwaarten zijn:

* Beperkte dynamische test: ●
* Gemiddelde dynamische test: ●●
* Zware dynamische test: ●●●

In een dynamische test wordt de software daadwerkelijk uitgevoerd. Daartegenover staan statische tests, waaronder het toetsen van documenten.

Waar de kolom Onderdeel niet is ingevuld geldt er één teststrategie voor alle onderdelen van {het product}. In deze kolom kan ook worden aangegeven dat de strategie de keten betreft.

De risiconummers verwijzen naar de in de PRA vastgestelde risico’s.

{Zorg dat de kolommen in de tabellen overeenkomen met de toe te passen testsoorten genoemd in de vorige paragraaf.}

##### Functionele geschiktheid

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Kenmerk | Onderdeel | RK | Risico-nummers | UT | UIT | ST | FAT | {...} |
| Functionele compleetheid |  | M |  | ●● |  |  |  |  |
| Functionele correctheid | {deel 1} | M |  | ●●● | ●●● |  | ●● |  |
| Functionele correctheid | {deel 2} | M |  | ●● | ●●● |  | ●● |  |
| Functionele toepasselijkheid |  | M |  |  |  | ●● |  |  |

##### Prestatie-efficiëntie

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Kenmerk | Onderdeel | RK | Risico-nummers | UT | UIT | ST | FAT | {...} |
| Snelheid |  | M |  | ●● |  |  |  |  |
| Middelenbeslag | {deel 1} | H |  | ●●● | ●●● |  | ●● |  |
| Middelenbeslag | {deel 2} | M |  | ●● | ●●● |  | ●● |  |
| Capaciteit |  | M |  | ●● |  |  |  |  |

##### Uitwisselbaarheid

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Kenmerk | Onderdeel | RK | Risico-nummers | UT | UIT | ST | FAT | {...} |
| Beïnvloedbaarheid | {deel 1} | M |  | ●● | ●● |  |  |  |
| Beïnvloedbaarheid | {deel 2} | M |  | ●● | ●● |  |  |  |
| Koppelbaarheid |  | M |  | ●● | ●● |  |  |  |

##### Bruikbaarheid

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Kenmerk | Onderdeel | RK | Risico-nummers | UT | UIT | ST | FAT | {...} |
| Herkenbaarheid van geschiktheid |  | M |  | ●● | ●● |  |  |  |
| Leerbaarheid | {deel 1} | M |  | ●● | ●● |  |  |  |
| Leerbaarheid | {deel 2} | M |  | ●● | ●● |  |  |  |
| Bedienbaarheid |  | M |  | ●● | ●● |  |  |  |
| Voorkomen gebruiksfouten | {deel 1} | H |  | ●● | ●●● |  |  |  |
| Voorkomen gebruiksfouten | {deel 2} | L |  | ●● | ●● |  |  |  |
| Volmaaktheid gebruikersinteractie |  | M |  | ●● | ●● |  |  |  |
| Toegankelijkheid |  | M |  | ●● | ●● |  |  |  |

##### Betrouwbaarheid

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Kenmerk | Onderdeel | RK | Risico-nummers | UT | UIT | ST | FAT | {...} |
| Volwassenheid |  | M |  | ●●● | ●● |  |  | ●● |
| Beschikbaarheid |  | M |  | ●●● | ●● |  |  | ●● |
| Foutbestendigheid |  |  |  |  |  |  |  |  |
| Herstelbaarheid |  | M |  | ●●● | ●● |  |  | ●● |

##### Beveiligbaarheid

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Kenmerk | Onderdeel | RK | Risico-nummers | UT | UIT | ST | FAT | {...} |
| Vertrouwelijkheid |  | L |  | ●●● | ● | ● |  |  |
| Integriteit | {deel 1} | H |  | ●● |  | ●●● |  |  |
| Integriteit | {deel 2} | L |  | ● |  | ● |  |  |
| Onweerlegbaarheid |  | L |  | ●●● | ● | ● |  |  |
| Verantwoording |  | L |  | ● | ● | ● | ● |  |
| Authenticiteit |  | L |  | ●●● | ● | ● |  |  |

##### Onderhoudbaarheid

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Kenmerk | Onderdeel | RK | Risico-nummers | UT | UIT | ST | FAT | {...} |
| Modulariteit |  |  |  |  |  |  |  |  |
| Herbruikbaarheid |  | L |  | ●●● | ● | ● |  |  |
| Analyseerbaarheid | {deel 1} | H |  | ●● |  | ●●● |  |  |
| Analyseerbaarheid | {deel 2} | L |  | ● |  | ● |  |  |
| Wijzigbaarheid |  |  |  |  |  |  |  |  |
| Testbaarheid |  |  |  |  |  |  |  |  |

##### Overdraagbaarheid

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Kenmerk | Onderdeel | RK | Risico-nummers | UT | UIT | ST | FAT | {...} |
| Aanpasbaarheid |  |  |  |  |  |  |  |  |
| Installeerbaarheid |  | M |  | ●●● | ●● |  |  | ●● |
| Vervangbaarheid |  | M |  | ●●● | ●● |  |  | ●● |

# Testaanpak

In dit hoofdstuk is de teststrategie vertaald naar een concrete testaanpak (het hoe). Voor iedere testsoort is hiervoor een deelparagraaf opgenomen.

{Zorg er hierbij voor dat de beschreven testaanpak de teststrategie uit het vorige hoofdstuk reflecteert. Elk element uit de teststrategie moet hier dan ook terugkomen. Als er, volgend op het MTP, voor een testsoort ook nog een detailtestplan (DTP) wordt opgesteld, dan is de betreffende paragraaf minder diepgaand dan wanneer dit niet het geval is (refereer dan wel aan het DTP). Twee belangrijke factoren die bepalen of er wel/niet DTP's worden opgesteld zijn

de omvang van het project en de mate waarin er bij het opstellen van het MTP nog onzekerheden/onduidelijkheden zijn.}

## Aansluiting op de agile werkwijze

In onderstaand plaatje is weergegeven welke testproducten worden gerealiseerd op welke momenten in het agile proces van ontwikkeling, ingebruikname en inbeheername:



De op de tests toe te passen acceptatiecriteria worden als volgt bepaald:

|  |  |
| --- | --- |
| Testsoort | Wanneer en hoe bepalen van acceptatiecriteria |
| Unittest (UT) | {Beschrijf wanneer en hoe de acceptatiecriteria worden bepaald} |
| Unit-integratietest (UIT) | {Beschrijf wanneer en hoe de acceptatiecriteria worden bepaald} |
| Systeemtest (ST) | De testspecificaties voor de systeemtest worden tijdens de sprints gemaakt, parallel aan de realisatie. |
| Functionele acceptatietest (FAT) | Functionele acceptatiecriteria voor een sprint worden vóór een sprint opgesteld (product backlog refinement). {Alternatief: De testspecificaties voor de functionele acceptatietest worden tijdens de sprints gemaakt, parallel aan de realisatie.} |
| Gebruikers-acceptatietest (GAT) | {Beschrijf wanneer en hoe de acceptatiecriteria worden bepaald} |
| Penetratietest (PEN) | {Beschrijf wanneer en hoe de acceptatiecriteria worden bepaald} |
| Performancetest (PERF) | {Beschrijf wanneer en hoe de acceptatiecriteria worden bepaald} |
| Gebruikerstevreden-heidstest (GEBR) | {Beschrijf wanneer en hoe de acceptatiecriteria worden bepaald} |
| Toegankelijkheidstest (TOEG) | {Beschrijf wanneer en hoe de acceptatiecriteria worden bepaald} |
| Productie-acceptatietest (PAT) | {Beschrijf wanneer en hoe de acceptatiecriteria worden bepaald} |
| Conformancetest (CONF) | {Beschrijf wanneer en hoe de acceptatiecriteria worden bepaald} |

## Aanpak van de toetsen

{Indien toetsen onderdeel zijn van de teststrategie, beschrijf dan hier de aanpak van deze toetsen of: “Toetsen zijn geen onderdeel van de teststrategie”.}.

## Aanpak van de testsoorten

{In deze template is de Systeemtest als voorbeeld genomen. Deze paragraaf wordt herhaald voor iedere toe te passen testsoort.}

### Aanpak systeemtest

{Als er voor een testsoort een detailtestplan wordt opgesteld, geef dan hier alleen een korte beschrijving van de testsoort en verwijs naar het detailtestplan.}

##### Beschrijving systeemtest algemeen

|  |  |
| --- | --- |
| Onderwerp | Beschrijving |
| Aandachtspunten | {Denk aan: bijzondere risico’s, en voor welke kenmerken; of de test voor specifieke onderdelen van het systeem moet worden uitgevoerd} |
| Acceptatiecriteria | {De criteria waaraan de testresultaten moeten voldoen voor de acceptatie van een increment of een release} |
| Uitvoering door | Scrumteam |
| Frequentie | {Bij welke (oplever)momenten in het voortbrengingsproces de test wordt uitgevoerd} |
| Specificaties opgesteld door |  |
| Testbasis | {GFO, use cases en user stories, informatiebeveiligingsplan} |
| Entry-criteria | {Wanneer met de testsoort kan worden begonnen. Denk aan: welke andere tests afgerond moeten zijn; noodzakelijke voorbereidingen} |
| Exit-criteria | {Wanneer de testsoort naar tevredenheid is afgerond. Heeft een relatie met de Definition of Done. Denk aan: vereiste testdekking; maximum aantal blokkerende bevindingen} |
| Infrastructuur |  |
| Testomgeving |  |

{Controleer of de hieronder beschreven testvormen voldoende invulling geven aan de teststrategie voor deze testsoort: per kwaliteitskenmerk, eventueel specifiek voor bepaalde onderdelen, met de gewenste zwaarte.}

##### Testvorm functionele test

|  |  |
| --- | --- |
| Onderwerp | Beschrijving |
| Omschrijving | Controleren of de functionaliteit correct is gerealiseerd conform de beschrijving in de functionele documentatie en procesbeschrijving |
| Onderdelen | Autorisaties t.a.v. wachtwoord, beheer en logging |
| Diepgang | Alle goed- en foutpaden |

##### Testvorm securitytest

|  |  |
| --- | --- |
| Onderwerp | Beschrijving |
| Omschrijving | Controleren of de security ‘functionals’ (wachtwoordinvoer, autorisatiebeheer, logging, toegang etc.) correct functioneren |
| Onderdelen |  |
| Diepgang |  |

##### Testvorm volledigheidstest

|  |  |
| --- | --- |
| Onderwerp | Beschrijving |
| Omschrijving | Steekproefsgewijs controleren of de mechanismes die de volledigheid moeten borgen (bijvoorbeeld tellingen) correct functioneren |
| Onderdelen |  |
| Diepgang |  |

##### Testvorm regressietest

|  |  |
| --- | --- |
| Onderwerp | Beschrijving |
| Omschrijving | Geautomatiseerd controleren dat de functionaliteit uit voorgaande sprints nog steeds correct functioneert |
| Onderdelen |  |
| Diepgang |  |

### Aanpak {testsoort}

{Gebruik de vorige paragraaf als template en vul deze in voor de andere testsoorten.}

## Op te leveren producten

De volgende producten worden door het testteam opgeleverd:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Testsoort | Product | Doel | Doelgroep | Opstellers | Reviewers |
| {testsoort} | {het product} | {doel} | {doelgroep} | {opstellers} | {reviewers} |

## Go / No go voor vervolg

{Beschrijf hier het Go/No go beslissingsproces. Het vervolg kan de realisatie van een volgend increment zijn of de vrijgave van een release voor ingebruikname van het systeem. In dat laatste geval wordt na afronding van het gehele testtraject een vrijgaveadvies opgesteld. Het vrijgaveadvies bevat alle nog openstaande testbevindingen en geconstateerde beveiligingsbevindingen. Er wordt vermeld in hoeverre aan de gestelde acceptatiecriteria is voldaan. Op basis van de resultaten van de verschillende testsoorten zijn de risico’s voor gebruikers en beheerorganisaties vastgesteld bij eventuele ingebruikname van het informatiesysteem.}

# Organisatie

## Organisatiestructuur

{Neem hier het organogram van de testorganisatie op en de relatie tot de projectorganisatie.}

## Rollen, taken en verantwoordelijkheden

In onderstaande tabel zijn per rol de taken en verantwoordelijkheden beschreven.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Rol | Naam | Testsoort | Uren per week | Periode | Taken en verantwooordelijkheden |
| {rol} | {naam} | {testsoort} | {uren per week} | {periode} | {taken en verantwoordelijkheden} |

{Maak onderscheid tussen overallniveau en per testsoort. Voeg extra personen toe en geef per rol de specifieke taken en verantwoordelijkheden aan. Alle betrokkenen die genoemd zijn in de testaanpak dienen hier terug te komen.}

{Optioneel: Per rol niet alleen de taken en verantwoordelijkheden benoemen, maar ook de bevoegdheden. Zonder bevoegdheden geen verantwoordelijkheden.}

## Opleidings- en coachingsbehoefte

{Vermeld hier de opleidings- en coachingsbehoefte van de testmedewerkers ten behoeve van de juiste materie- en/of testkennis.}

## Overlegstructuur

In de volgende overleggen worden afspraken gemaakt over het plannen en uitvoeren van tests:

|  |  |  |  |
| --- | --- | --- | --- |
| Overleg | Doel | Frequentie | Deelnemers |
| Projectoverleg | Voortgang project bespreken, risicolog behandelen | wekelijks | Projectleiders, testmanager |
| Werkoverleg per testsoort | Stand van zaken per testsoort bespreken | wekelijks | Testmanager, testers |
| Bevindingenoverleg | Bespreking inhoud, risico en planning rondom bevindingen | {frequentie} | {deelnemers} |
| {overleg} | {doel} | {frequentie} | {deelnemers} |

# Testinfrastructuur

Het uitgangspunt voor de testinfrastructuur is dat verschillende testsoorten in verschillende testomgevingen kunnen worden gedaan. Deze scheiding is nodig om te kunnen voldoen aan de verschillende eisen van de testsoorten; een performancetest stelt bijvoorbeeld andere eisen aan een omgeving dan een usabilitytest.

{Zie eventuele detailtestplannen.}

## Testomgevingen

Onderstaande tabel bevat een opsomming van de testomgevingen per testsoort. Zie de volgende paragraaf voor het beheer van de testomgevingen.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Test-soort | Testomgeving | Infrastructuur | Testgegevens | Periode | Kosten |
| UT | {testomgeving} | {infrastructuur} | {testgegevens} | {periode} | {kosten} |
| UIT | {testomgeving} | {infrastructuur} | {testgegevens} | {periode} | {kosten} |
| ST | Ontwikkel-omgeving | {infrastructuur} | {testgegevens} | {periode} | {kosten} |
| FAT | Testomgeving | {infrastructuur} | {testgegevens} | {periode} | {kosten} |
| GAT | Acceptatie-omgeving | {infrastructuur} | {testgegevens} | {periode} | {kosten} |
| PEN | Productie-omgeving | {infrastructuur} | {testgegevens} | {periode} | {kosten} |
| PERF | {testomgeving} | {infrastructuur} | {testgegevens} | {periode} | {kosten} |
| GEBR | {testomgeving} | {infrastructuur} | {testgegevens} | {periode} | {kosten} |
| TOEG | {testomgeving} | {infrastructuur} | {testgegevens} | {periode} | {kosten} |
| PAT | {testomgeving} | {infrastructuur} | {testgegevens} | {periode} | {kosten} |
| CONF | {testomgeving} | {infrastructuur} | {testgegevens} | {periode} | {kosten} |

# Testtools

Onderstaande tabel bevat een opsomming van de testtools per testsoort of testvorm.

{Deze tabel kan ook worden gecombineerd met die in de vorige paragraaf.}

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Testsoort | Testvorm | Testtool | Toelichting | Periode | Kosten |
| UT | {testvorm} | {testtool} | {toelichting} | {periode} | {kosten} |
| UIT | {testvorm} | {testtool} | {toelichting} | {periode} | {kosten} |
| ST | {testvorm} | {testtool} | {toelichting} | {periode} | {kosten} |
| FAT | {testvorm} | {testtool} | {toelichting} | {periode} | {kosten} |
| GAT | {testvorm} | {testtool} | {toelichting} | {periode} | {kosten} |
| PEN | {testvorm} | {testtool} | {toelichting} | {periode} | {kosten} |
| PERF | {testvorm} | {testtool} | {toelichting} | {periode} | {kosten} |
| GEBR | {testvorm} | {testtool} | {toelichting} | {periode} | {kosten} |
| TOEG | {testvorm} | {testtool} | {toelichting} | {periode} | {kosten} |
| PAT | {testvorm} | {testtool} | {toelichting} | {periode} | {kosten} |
| CONF | {testvorm} | {testtool} | {toelichting} | {periode} | {kosten} |

# Beheer

In dit hoofdstuk wordt beschreven hoe de activiteiten uit het testplan worden uitgevoerd, bewaakt en eventueel bijgestuurd.

## Testproductbeheer

{Beschrijf hier hoe de op te leveren producten worden beheerd en bewaakt. Beschrijf hiervoor procedures, sjablonen, tools en de omgeving waar de producten worden vastgelegd.}

## Bevindingenprocedures

Het bevindingenbeheer is ingericht conform de {in de methode X beschreven bevindingenprocedure, of, de bij de klantorganisatie vigerende bevindingenprocedure}. Voor het registreren en onderhouden van bevindingen wordt gebruik gemaakt van {tool}.

De verantwoordelijkheid voor de naleving van het bevindingenbeheer ligt bij de {testmanager}.

Onderstaande tabel bevat de bevindingenprocedures per testsoort en/of testvorm.

|  |  |  |
| --- | --- | --- |
| Testsoort | Testvorm | Bevindingenprocedure |
| UT | {testvorm} | {bevindingenprocedure} |
| UIT | {testvorm} | {bevingingenprocedure} |
| ST | {testvorm} | De bevindingen die niet binnen de sprint worden opgelost zullen geregistreerd worden in het bevindingenregistratiesysteem. Deze bevindingen komen op de product backlog en zullen op basis van prioriteit door de product owner in één van de volgende sprints worden gepland. |
| FAT | {testvorm} | De bevindingen worden vastgelegd in het bevindingenregistratiesysteem en in bevindingenoverleg besproken. Deze bevindingen komen op de product backlog en worden op basis van prioriteit door de product owner in dezelfde sprint of in één van de volgende sprints gepland. Een uitzondering hierop vormen blokkerende testbevindingen. Voor deze bevindingen moet zo snel mogelijk binnen de lopende sprint een oplossing worden gerealiseerd. Dit zal in overleg met het ontwikkelteam door de product owner worden bepaald. |
| GAT | {testvorm} | Zie FAT. |
| PEN | {testvorm} | Zie FAT. |
| PERF | {testvorm} | {bevingingenprocedure} |
| GEBR | {testvorm} | {bevingingenprocedure} |
| TOEG | {testvorm} | {bevingingenprocedure} |
| PAT | {testvorm} | Zie FAT. |
| CONF | {testvorm} | {bevingingenprocedure} |

# Testprocesrisico's en maatregelen

In dit hoofdstuk zijn de belangrijkste potentiële projectrisico's voor het testproces van {project} beschreven. Door te anticiperen op wat mogelijk kan gebeuren kunnen nu reeds maatregelen worden getroffen. De risico’s kunnen optreden in het testproces zelf of projectrisico’s zijn met gevolgen voor het testproces.

Het in kaart brengen en bewaken van deze risico’s is ook na het opstellen van dit MTP een continu proces: bij uitwerking van de testaanpak en bij aanvang van de testuitvoering van elke testsoort zal de risicotaxatie worden geactualiseerd.

De volgende risico’s voor het testproces zijn onderkend. Zie ook het risicolog {documentreferentie}.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Risicovolg-nummer | Testsoort | Gebeurtenis | Gevolg | Impact | Kans | Score |
| {risicovolg-nummer} | {testsoort} | {gebeurtenis} | {gevolg} | {impact} | {kans} | {score} |

De volgende maatregelen zijn getroffen.

|  |  |  |  |
| --- | --- | --- | --- |
| Risicovolgnummer | Maatregelvolgnummer | Maatregel | Eigenaar |
| {risicovolgnummer} | {maatregelvolgnummer} | {maatregel} | {eigenaar} |

De testmanager is op deze punten extra alert en bewaakt de te nemen maatregelen.

De maatregelen die niet binnen de testorganisatie kunnen worden genomen worden in het projectoverleg besproken en toegewezen. Ze worden vastgelegd in de actielijst en bewaakt tijdens het projectoverleg.

# Globale planning

{Neem in de globale planning de volgende onderdelen op:}

* {de onderscheiden testsoorten en testvormen}
* {op te leveren producten;}
* {te behalen mijlpalen;}
* {uit te voeren activiteiten (op faseniveau per testsoort) met start- en einddatum;}
* {relaties met en afhankelijkheden van andere activiteiten (binnen of buiten het testproces en tussen de diverse testsoorten);}
* {te besteden tijd per testsoort;}
* {benodigde en beschikbare resources (organisatie en infrastructuur);}
* {benodigde en beschikbare doorlooptijd.}

Bijlagen

1. Terminologie en afkortingen

De onderstaande tabel bevat afkortingen en termen die voorkomen in de ICTU Kwaliteitsaanpak Softwareontwikkeling en bijbehorende templates.

|  |  |
| --- | --- |
| Term/afkorting | Toelichting |
| **actor** | een persoon die, of een extern informatiesysteem dat, een handeling verricht op het **informatiesysteem** |
| **architectuur** | een beschrijving van de structuur van een systeem, inclusief onderdelen, relaties tussen die onderdelen en eigenschappen van die onderdelen en relaties |
| **API** | application programming interface |
| **ART** | automatische **regressietest** |
| **auditing** | Vastlegging van de door een actor verrichte handelingen |
| **authenticatie** | het vaststellen van de identiteit van een **actor** |
| **autorisatie** | aan een **actor** toegekende rechten |
| **beheerorganisatie** | een (samenwerkingsverband van) organisatie(s) die in opdracht van een **opdrachtgevende organisatie** het **operationeel beheer**, applicatief beheer en/of functioneel beheer van **software** uitvoert |
| **BIA** | Een business impact analyse is een methode om de mogelijke bedrijfsimpact te bepalen die een organisatie zou kunnen ervaren door een incident, dat de functionaliteit van of de informatie in een applicatie in gevaar brengt [NORA] |
| **BIO** | Baseline Informatiebeveiliging Overheid |
| **broncode** | **software** in een vorm die leesbaar is voor mensen en de intentie van een programmeur uitdrukt |
| **deployment** | installatie van **software** op een systeem waardoor de software beschikbaar wordt gemaakt voor gebruik door **actor**en |
| **developers** | Developers zijn de mensen in het **Scrumteam** die iedere sprint gecommitteerd zijn aan het maken van elk aspect van een bruikbaar increment [Scrumgids] |
| **DevOps** | een praktijk die tot doel heeft **softwareontwikkeling** en **operationeel beheer** samen te brengen |
| **DoD** | definition of done |
| **DoR** | definition of ready |
| **DPIA** | Een data protection impact assessment is een instrument om vooraf de privacyrisico’s van een gegevensverwerking in kaart te brengen zodat de organisatie maatregelen kan nemen om deze risico’s te verkleinen |
| **gebruikskwaliteit** | mate waarin een systeem, product of dienst kan worden gebruikt door gespecificeerde gebruikers, voor het bereiken van gespecificeerde doelen, met effectiviteit, efficiëntie en tevredenheid in een gespecificeerde gebruikscontext |
| **GFO** | Een globaal functioneel ontwerp beschrijft de functionele werking van een product op hoofdlijnen, voor specifieke use cases |
| **IB-plan** | Een informatiebeveiligingsplan beschrijft binnen welke kaders bescherming geleverd wordt tegen welke dreigingen en met welke maatregelen die bescherming vorm krijgt |
| **informatiesysteem** | een samenhangend geheel van gegevensverzamelingen en de daarbij behorende personen, procedures, processen en **programmatuur** alsmede de voor het informatiesysteem getroffen voorzieningen voor opslag, verwerking en communicatie [VIR 2007, NORA] |
| **infrastructuurarchitectuur** | De infrastructuurarchitectuur beschrijft de technische infrastructuur van een product op hoofdlijnen, in termen van hardwareonderdelen en -relaties (housing, hardware, virtuals, standaard software en middleware) |
| **interactie-ontwerp** | Een interactie-ontwerp beschrijft de interacties tussen gebruikers en het systeem en de user experience daarbij |
| **IPO** | intern projectoverleg |
| **ISD** | ICTU Software Diensten, afdeling van ICTU die **softwareontwikkelprojecten** ondersteunt met ontwikkel- en testomgevingen, tools en diensten |
| **ISE** | ICTU Software Expertise, afdeling van ICTU die **softwareontwikkelprojecten** ondersteunt met expertise op het gebied van **softwareontwikkeling** en die de ICTU Kwaliteitsaanpak Softwareontwikkeling onderhoudt |
| **ISO** | International Organization for Standardization |
| **Jira** | tool om **use cases**, user stories, logische testgevallen en issues vast te leggen |
| **klantreis** | alle directe en indirecte interactie van een klant of gebruiker met een product of dienst |
| **KPI** | key performance indicator |
| **kwaliteitsmanager** | controleert en borgt de kwaliteit van **software** conform de vastgestelde eisen en de Kwaliteitsaanpak en rapporteert aan de **projectleider** |
| **minimum viable product** | Een minimum viable product is een eerste versie van een product, die zo vroeg mogelijk wordt uitgerold naar de gebruikers, met net voldoende functionaliteit om het gestelde doel te behalen en niet meer dan dat |
| **MTP** | Een mastertestplan beschrijft de aanpak van het testen van een product op hoofdlijnen, in termen van strategie, activiteiten, afhankelijkheden en de op te leveren resultaten |
| **MVP** | **minimum viable product** |
| **NFE** | Niet-functionele eisen specificeren criteria om de kwaliteit van de software te beoordelen |
| **NORA** | Nederlandse Overheidsreferentie-architectuur |
| **NPR** | Nederlandse Praktijkrichtlijn |
| **ontwikkelaars** | Ontwikkelaars (*developers* in de Scrumgids) zijn de mensen in het **Scrumteam** die iedere sprint gecommitteerd zijn aan het maken van elk aspect van een bruikbaar increment [Scrumgids] |
| **opdrachtgevende organisatie** | overheidsorganisatie die opdracht geeft aan ICTU tot ontwikkeling en/of onderhoud van **software** |
| **opdrachtgever** | medewerker van de **opdrachtgevende organisatie** die eindverantwoordelijk is voor de opdracht aan ICTU |
| **operationeel beheer** | activiteiten die zorgen dat software operationeel is en blijft, zoals het oplossen van incidenten, het uitvoeren van onderhoud, het implementeren van upgrades en patches, het beheren van configuraties, en het monitoren van prestaties en beschikbaarheid |
| **OTAP** | ontwikkel, test, acceptatie, productie; gebruikt om verschillende soorten omgevingen aan te duiden |
| **persona** | een min of meer realistische beschrijving van een fictief persoon, veelal met naam, persoonskenmerken, drijfveren en behoeften, die een groep gebruikers representeert en gebruikt wordt om te redeneren over de gewenste functionele en niet-functionele eigenschappen van de **software** |
| **PKI** | public key infrastructure |
| **PRA** | Een productrisicoanalyse is een analyse van het te testen product die resulteert in een overzicht van wat de meer of minder risicovolle kenmerken en delen van het te testen product zijn, zodat de grondigheid van testen hieraan gerelateerd kan worden |
| **product backlog** | De product backlog is een levende, geordende lijst van wat nodig is om het product te verbeteren. Het is de enige bron van het werk dat door het **Scrumteam** gedaan wordt [Scrumgids] |
| **product owner** | De product owner is verantwoordelijk voor het maximaliseren van de waarde van het product, dat het resultaat is van het werk van het **Scrumteam** [Scrumgids] |
| **programmatuur** | zie **software** |
| **project** | een tijdelijke organisatie voor het realiseren van een resultaat - bij ICTU bestaat een **softwareontwikkelproject** uit medewerkers van ICTU, de **opdrachtgevende organisatie**, beheerorganisatie en eventueel andere partijen |
| **projectleider** | medewerker eindverantwoordelijk voor het projectresultaat - bij ICTU-softwareontwikkelprojecten is de projectleider een medewerker van ICTU |
| **PSA** | De projectstartarchitectuur is een concreet en doelgericht ICT-architectuurkader waarbinnen het **project** moet worden uitgevoerd [NORA] |
| **PvE** | programma van eisen |
| **Quality-time** | een door ICTU ontwikkeld, open source, geautomatiseerd kwaliteitssysteem |
| **realisatiefase** | fase van een **softwareontwikkelproject** waarin de **software** daadwerkelijk wordt gebouwd en onderhouden, en bij een **DevOps** werkwijze ook operationeel wordt beheerd |
| **regressietest** | test die na een wijziging controleert of niet-gewijzigde delen van een systeem nog steeds correct functioneren |
| **release notes** | een overzicht van de wijzigingen in een **release** |
| **release** | een voor gebruik vrijgegeven versie van de **software** |
| **SAD** | Een software-architectuurdocument beschrijft de technische werking van een product op hoofdlijnen, in termen van softwarecomponenten, hun functies en hun onderlinge interacties en samenhang voor specifieke use cases |
| **Scrum** | Scrum is een lichtgewicht raamwerk dat mensen, teams en organisaties helpt om waarde te creёren door middel van adaptieve oplossingen voor complexe problemen [Scrumgids] |
| **Scrummaster** | De Scrummaster is verantwoordelijk voor het opzetten van **Scrum**, zoals staat beschreven in de Scrumgids [Scrumgids] |
| **Scrumteam** | Een Scrumteam bestaat uit één **Scrummaster**, één **product owner** en **ontwikkelaars** (*developers* in de Scrumgids) [Scrumgids] |
| **softwarearchitectuur** | een **architectuur** die vooral de softwareonderdelen en -relaties (processen, modules, interfaces, datamodel) van een systeem beschrijft |
| **software delivery manager** | organiseert het ontwikkelen en opleveren van **software** conform de vastgestelde eisen en de Kwaliteitsaanpak en rapporteert aan de **projectleider** |
| **software** | software is de verzameling instructies die bepalen wat een computer uitvoert en is uiteindelijk wat de gebruiker ziet, ervaart en waarmee hij interacteert |
| **softwareontwikkeling** | een activiteit die nieuwe **software** maakt en/of bestaande software aanpast |
| **softwareontwikkelproject** | een **project** dat de oplevering van **software** als enige of voornaamste projectresultaat heeft |
| **solution architectuur** | beschrijving van de gewenste oplossing van een specifiek probleem, of het eindresultaat van een **project** [NORA] |
| **technische schuld** | eigenschappen van de **software** die de lange-termijninzetbaarheid en onderhoudbaarheid bedreigen |
| **TVA** | Een threat and vulnerability assessment inventariseert de betrouwbaarheidseisen die aan de bedrijfsprocessen en dientengevolge aan het product worden gesteld, gevolgd door identificatie en analyse van bedreigingen |
| **usability** | gebruiksvriendelijkheid |
| **use case** | een afgebakende eenheid van interactie tussen een **actor** en het systeem |
| **UX** | user experience |
| **VIR** | Voorschrift Informatiebeveiliging Rijksdienst |
| **VIRBI** | Voorschrift Informatiebeveiliging Rijksdienst Bijzondere Informatie |
| **VM** | virtual machine, virtuele machine |
| **voorfase** | fase van een **softwareontwikkelproject**, voorafgaande aan de **realisatiefase**, waarin de uitgangspunten, risico's en randvoorwaarden voor de realisatiefase worden bepaald en waarin wordt gezorgd dat aan de randvoorwaarden wordt voldaan en dat voor zoveel mogelijk risico's maatregelen getroffen zijn |
| **vrijgaveadvies** | advies om een **release** vrij te geven voor ingebruikname, met een testverslag dat tenminste alle nog openstaande testbevindingen en geconstateerde beveiligingsbevindingen bevat |

1. Bronnen

De onderstaande tabel verwijst naar regelmatig gebruikte bronnen.

|  |  |
| --- | --- |
| Bron | Toelichting |
| [BIO](https://bio-overheid.nl/media/1572/bio-versie-104zv_def.pdf) | Baseline Informatiebeveiliging Overheid. |
| [ISO 9241-210:2019](https://www.iso.org/standard/77520.html) | Ergonomics of human-system interaction — Part 210: Human-centred design for interactive systems. |
| [NCSC ICT-beveiligingsrichtlijnen voor webapplicaties](https://www.ncsc.nl/documenten/publicaties/2019/mei/01/ict-beveiligingsrichtlijnen-voor-webapplicaties) | De ICT-beveiligingsrichtlijnen voor webapplicaties geven een leidraad voor veiliger ontwikkelen, beheren en aanbieden van webapplicaties en bijbehorende infrastructuur. |
| [NEN-ISO/IEC 25010:2023](https://www.nen.nl/nen-iso-iec-25010-2023-en-318088) | Systems and software engineering - Systems and software Quality Requirements and Evaluation (SQuaRE) - System and software quality models. |
| [NEN-ISO/IEC 27001:2017](https://www.nen.nl/nen-en-iso-iec-27001-2017-a11-2020-nl-265545) | Informatietechnologie - Beveiligingstechnieken - Managementsystemen voor informatiebeveiliging - Eisen |
| [NEN-ISO/IEC 27002:2017](https://www.nen.nl/nen-en-iso-iec-27002-2017-nl-245390) | Informatietechnologie - Beveiligingstechnieken - Praktijkrichtlijn met beheersmaatregelen op het gebied van informatiebeveiliging |
| [NEN 7510:2017](https://www.nen.nl/nen-7510-1-2017-a1-2020-nl-267179) | Informatiebeveiliging in de zorg. |
| [NEN NPR 5325:2017](https://www.nen.nl/npr-5325-2017-nl-238298) | Praktijkrichtlijn voor het overdragen van software. |
| [NEN NPR 5326:2019](https://www.nen.nl/npr-5326-2019-nl-262885) | Praktijkrichtlijn voor risicobeheersing bij softwareontwikkeling. |
| [NORA](https://www.noraonline.nl) | Referentiearchitectuur voor de Nederlandse Overheid. |
| [OWASP Top-10](https://owasp.org/www-project-top-ten/) | De OWASP Top-10 is een op consensus gebaseerd overzicht van de meest kritische beveiligingsrisico's voor webapplicaties. |
| [Scrumgids](https://scrumguides.org/docs/scrumguide/v2020/2020-Scrum-Guide-Dutch.pdf) | De Scrum Gids - De Definitieve Gids voor Scrum: De Regels van het Spel. |
| [VIR 2007](https://wetten.overheid.nl/BWBR0022141/2007-07-01) | Besluit Voorschrift Informatiebeveiliging Rijksdienst 2007. |
| [VIRBI 2013](https://wetten.overheid.nl/BWBR0033507/2013-06-01) | Besluit Voorschrift Informatiebeveiliging Rijksdienst Bijzondere Informatie 2013. |
| [Wbni 2018](https://wetten.overheid.nl/BWBR0041515/2020-07-15) | Wet Beveiliging Netwerk- en Informatiesystemen. Beschrijft de meldplicht en de zorgplicht die van toepassing zijn op organisaties die vitaal zijn én op digitale dienstverleners. |

1. De ICTU Kwaliteitsaanpak Softwareontwikkeling

De overheid is in hoge mate afhankelijk van informatiesystemen voor de uitvoering van haar taken. Veel van die informatiesystemen zijn dusdanig specifiek dat de benodigde software “op maat” gemaakt moet worden. De totstandkoming van op maat gemaakte software is meestal een complex proces, waarin vele belangen en behoeften worden afgewogen en afgezet tegen de mogelijkheden die technologie biedt. Eenmaal operationeel zal een informatiesysteem verantwoord onderhouden moeten worden; behoeften en technologie veranderen in de loop van de tijd.

Overheidsprojecten waarin software wordt ontwikkeld of onderhouden kampen nog vaak met vertraging, budgetoverschrijding of een eindresultaat met te lage kwaliteit. Zo concludeerde de commissie-Elias in haar [eindrapport](https://www.tweedekamer.nl/sites/default/files/field_uploads/33326-5-Eindrapport_tcm181-239826.pdf): "De Rijksoverheid heeft haar ICT (Informatie- en communicatietechnologie)-projecten niet onder controle". Eén van de fundamentele problemen is dat de risico's, die inherent zijn aan softwareontwikkeling, door organisaties nog onvoldoende worden herkend, erkend en gemitigeerd. Dit terwijl de risico's bij de ontwikkeling van software, binnen het ICT-domein, algemeen bekend zijn en er ook voor veel risico's passende maatregelen bestaan.

ICTU heeft jarenlange ervaring met het realiseren van software en past de opgedane ervaring toe bij de ontwikkeling van nieuwe software. Die ervaring is vastgelegd in een werkwijze, deze “ICTU Kwaliteitsaanpak Softwareontwikkeling”, die telkens wordt aangepast en aangevuld op basis van de praktijk.

ICTU is ervan overtuigd dat het bouwen van duurzame software, die goed aansluit bij de behoeften van gebruikers en andere belanghebbenden, bijdraagt aan betere informatiesystemen en een betere dienstverlening door de overheid. Dienstverlening die betrouwbaar moet zijn voor burgers, bedrijven en ambtenaren. Om samen met opdrachtgevende organisaties passende oplossingen te realiseren ontwikkelt ICTU daarom software volgens een agile proces. En om de duurzaamheid en betrouwbaarheid te bevorderen besteedt ICTU standaard aandacht aan beveiliging, privacy, performance, gebruikskwaliteit en toegankelijkheid. De Kwaliteitsaanpak dient daarvoor als leidraad, maar de aanpak voorziet ook in mogelijkheden om het project en het eindproduct aan te passen aan de specifieke situatie.

Om projecten, die software realiseren volgens de Kwaliteitsaanpak, efficiënt en effectief te ondersteunen, heeft ICTU twee gespecialiseerde afdelingen in het leven geroepen. Deze afdelingen staan projecten bij door middel van kennis, menskracht en technische hulpmiddelen. Zo profiteren projecten van schaalgrootte en hergebruik van inzichten.

Met behulp van de ICTU Kwaliteitsaanpak Softwareontwikkeling heeft ICTU samen met andere overheden inmiddels enige tientallen projecten succesvol uitgevoerd. ICTU wil deze aanpak graag aanvullen met de ervaringen en geleerde lessen van andere organisaties en deze overdraagbaar maken en breder uitdragen. Om die reden stelt ICTU deze Kwaliteitsaanpak aan iedereen beschikbaar via <https://www.ictu.nl/kwaliteitsaanpak> en heeft zij, samen met normalisatie-instituut NEN en partijen uit overheid en markt, een praktijkrichtlijn “Risicobeheersing bij ontwikkeling en onderhoud van maatwerksoftware” [NEN NPR 5326:2019] gepubliceerd, die mede is gebaseerd op de ICTU Kwaliteitsaanpak Softwareontwikkeling.

De ICTU Kwaliteitsaanpak Softwareontwikkeling heeft drie doelstellingen:

1. Opdrachtgevende organisaties helpen bekende risico's bij softwareontwikkeling, zoals technische schuld, vertraging en defecten, zo veel mogelijk te voorkomen.
2. ICTU helpen om software te ontwikkelen die de missie van ICTU, namelijk bijdragen aan een betere digitale overheid, ondersteunt.
3. De overheid als geheel helpen bij het zo goed mogelijk ontwikkelen van software.

De Kwaliteitsaanpak zelf is geformuleerd in de vorm van maatregelen die elke software-ontwikkelende organisatie kan treffen om risico's van softwareontwikkeling te mitigeren en de kans op succesvolle softwareontwikkelprojecten te vergroten. De maatregelen zijn gebaseerd op geleerde lessen uit de praktijk van ICTU.

De Kwaliteitsaanpak is een evoluerende aanpak, gebaseerd op de ervaringen die ICTU continu opdoet in de projecten waarin ICTU samen met opdrachtgevende organisaties maatwerksoftware ontwikkelt en onderhoudt. ICTU hanteert daarbij de vuistregel dat als tenminste 80% van de projecten minstens 80% van de tijd een bepaalde werkwijze hanteren, voor die werkwijze een maatregel in de Kwaliteitsaanpak wordt opgenomen. Maar het kan ook voorkomen dat maatregelen om andere redenen landen in de Kwaliteitsaanpak; denk aan het toegankelijk maken van software dat wettelijk verplicht is. Zie ook de wijzigingsgeschiedenis in [PDF-formaat](https://ictu.github.io/Kwaliteitsaanpak/wip/ICTU-Kwaliteitsaanpak-Wijzigingsgeschiedenis.pdf) of [HTML-formaat](https://ictu.github.io/Kwaliteitsaanpak/wip/ICTU-Kwaliteitsaanpak-Wijzigingsgeschiedenis.html).

De maatregelen vormen het startpunt voor de aanpak van ieder ICTU-softwareproject, waarbij ruimte wordt geboden voor variatie of alternatieve invulling. Bijvoorbeeld stelt de Kwaliteitsaanpak: software wordt minimaal bij iedere grote release of tenminste twee keer per jaar onderworpen aan een beveiligingstest door beveiligingsexperts die ICTU daarvoor inhuurt (zie M26: Het project laat de beveiliging van het ontwikkelde product periodiek beoordelen). Een alternatief is dat de opdrachtgevende organisatie de verantwoordelijkheid neemt voor het laten uitvoeren van beveiligingstests. Hierover maakt de projectleider nadere afspraken met de opdrachtgever.

De Kwaliteitsaanpak is dus zowel voorschrijvend als beschrijvend. Voorschrijvend omdat ICTU verwacht dat projecten die maatwerksoftware ontwikkelen en onderhouden de aanpak toepassen, en alleen aanpassen als daar een goede reden voor is, en mits dat wettelijk is toegestaan. Tegelijkertijd is de aanpak beschrijvend omdat de meeste maatregelen voortkomen uit de bestaande werkwijzen van de projecten. Zoals blijkt uit de self-assessment die ICTU regelmatig uitvoert op de toepassing van de Kwaliteitsaanpak.