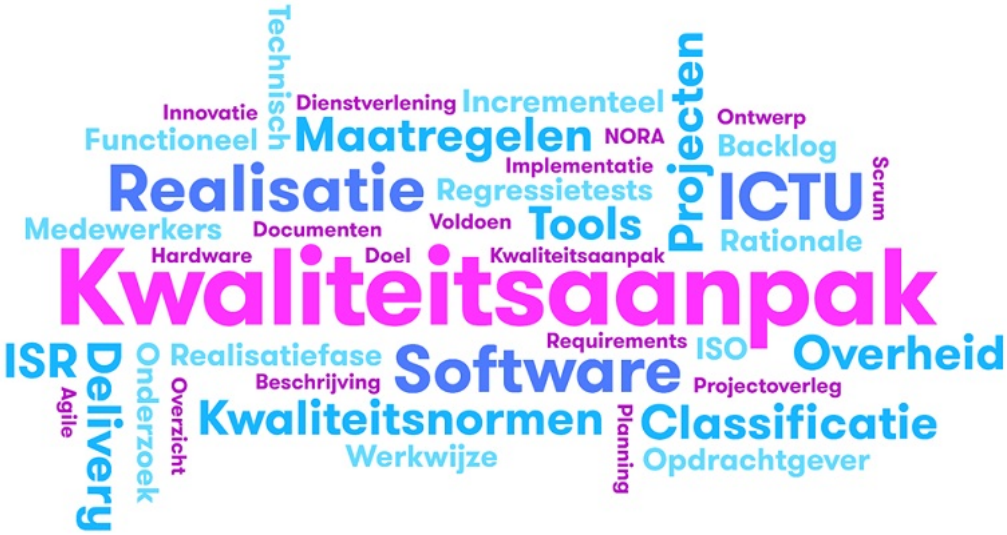
High-Level Design
{Projectnaam}
Versie {versienummer}
Datum {datum}



Inhoudsopgave

Revisiehistorie

Versie	Auteur	Datum	Status	Opmerkingen
{versie}	{naam}	{datum}	{concept/definitief}	{opmerkingen}

Vereiste goedkeuringen

Functie/rol	Naam	Datum	Versie
Projectleider ICTU	{naam}	{datum}	{versie}
Projectleider opdrachtgever	{naam}	{datum}	{versie}
Product owner	{naam}	{datum}	{versie}

Verzendlijst huidige versie

Naam	Organisatie	Functie/rol
{naam}	{opdrachtgever}	Projectleider
{naam}	{opdrachtgever}	Product owner
{naam}	ICTU	Projectleider
{naam}	ICTU	Software delivery manager

Template versie

Versie 1.3.1-build.9, 29-08-2019

1 Inleiding

1.1 Over dit document

Het High Level Design (HLD) heeft als doel om een high-level overzicht te geven van de technische infrastructuur van {systeem}. Hierbij is er vanuit gegaan dat de oplossing ten minste {aantal} jaar conform specificaties kan functioneren. In de praktijk is een kortere of langere periode mogelijk.

Onder infrastructuur wordt verstaan het samenstel van alle generieke off-the-shelf ICT-componenten die nodig zijn om de applicatie te kunnen installeren, operationeel te maken en houden. De infrastructuur eindigt daar waar specifieke elementen (code) en applicatiespecifieke configuraties ontstaan. Concreet omvat de infrastructuur:

- Housing (locatie, gebouw, racks, stroom, koeling, bekabeling, brandbeveiliging, e.d.),
- Hardware (harddisk, SAN, servers, netwerkswitches, proxy's, load balancers, HSMs, e.d.),
- Virtuals (virtual cpu, virtual memory, virtual disk, virtual servers, virtual network, e.d.),
- Standaardsoftware / middleware (zoals operating system, webserver, applicatieserver, databaseserver, messaging- en ontwikkelplatform).

Dit document beschrijft de lagen van de hardware en virtuals op het niveau van globale settings, netwerkarchitectuur en guidelines. Er zal dus niet op technisch detailniveau worden ingegaan op instellingen van routers, firewalls, virtualisatie en (virtuele) servers.

1.2 Doelgroep

De doelgroep van het opgestelde high level design zijn de partijen betrokken bij of verantwoordelijk voor de daadwerkelijke realisatie van de infrastructuur en de partijen betrokken bij softwarerealisatie, aangezien de software op deze infrastructuur zal moeten draaien.

1.3 Kaders

De volgende kaders zijn van toepassing op het projectresultaat:

Volgnummer	Kader
K01	ISO/IEC 27001:2005, ISO/IEC 27001:2013, 27002:2007, VIR, VIR-BI en BIO:2017 voor het inrichten en beheren van informatiebeveiliging in brede zin.
K02	WCAG2.1 (Web Content Accessibility Guidelines) voor eisen met betrekking tot toegankelijkheid
{volgnummer}	{kader}

1.4 Uitgangspunten

De volgende uitgangspunten zijn van toepassing op dit document:

Volgnummer	Uitgangspunt
U01	{Toepassing OTAP-omgevingen. NB. Er kan sprake zijn van additionele O-, T- en A-omgevingen, maar ook van omgevingen als pre-productie- en experimenteeromgevingen.}
U03	{3-Tier}
U03	{IP-adres onafhankelijkheid. IP adressen van systemen of systeemonderdelen zijn te wijzigen zonder dat dit een significantie impact heeft op de applicatie, bijvoorbeeld door ontkoppeling middels DNS}
U04	Communicatie via {netwerk} of {internet}. Systeem-naar-systeemcommunicatie van en naar {systeem} verloopt via {netwerk}.
U05	{Virtualisatie tenzij. In het kader van schaalbaarheid, flexibiliteit en kosten wordt gebruik gemaakt van virtuele servers tenzij dit technisch niet kan.}
U06	{Uitgangspunten met betrekking tot redundantie.}
{volgnummer}	{uitgangspunt}

1.5 Relatie met andere documenten

In verschillende documenten zijn eisen en wensen opgenomen die ten grondslag liggen aan dit High Level Design. Deze zijn opgenomen in de volgende documenten:

- Globaal Functioneel Ontwerp (GFO) (functionele requirements), {documentreferentie}
- Projectstartarchitectuur (PSA), {documentreferentie}
- Software-architectuurdocument (SAD), {documentreferentie}
- Niet-functionele eisen (NFE), {documentreferentie}
- Informatiebeveiligingsplan (bevat eisen en wensen en mogelijk maatregelen tegen informatiebeveiligingsrisico's), {documentreferentie}

Het Low Level Design (LLD) beschrijft elk sub-component van de infrastructuur op een gedetailleerder niveau, zoals de instellingen van routers en firewalls.
{IS ER NIET ALTIJD, NIET IN STANDAARD LIJST}

1.6 Leeswijzer

Hoofdstuk 2 bevat de technische eisen die nog niet zijn opgenomen in andere documenten. Hoofdstuk 3 beschrijft het ontwerp, waarbij in de eerste paragraaf het netwerkontwerp wordt beschreven en daarna de verschillende subcomponenten.

2 Technische infrastructuureisen

De technische eisen die ten grondslag liggen aan dit document zijn en niet beschreven staan in andere documenten uit de voorfase:

Volgnummer	Infrastructuureis
TR01	{eis}
TR02	{eis}

3 Ontwerp

3.1 Netwerkomgeving

{Beschrijving van de netwerkomgeving}

{Infrastructuurplaat met zonering, waarbij aangegeven in welke laag van de architectuur je alles configureert}

{beschrijving OTAP-omgevingen}

3.2 {Component 1}

{Beschrijving component}

{Componentnaam} bestaat uit onderstaande subcomponenten:

Subcomponent	Rol	Specificatie	Zone
{naam subcomponent}	{rol}	{specificatie}	{zone}

{Beschrijving van subcomponenten en een gedetailleerde infrastructuurplaat opnemen, indien dit toegevoegde waarde heeft.}

3.3 {Component 2}

{Beschrijving component}

{Componentnaam} bestaat uit onderstaande subcomponenten:

Subcomponent	Rol	Specificatie	Zone
{naam subcomponent}	{rol}	{specificatie}	{zone}

{Beschrijving van subcomponenten en een gedetailleerde infrastructuurplaat opnemen, indien dit toegevoegde waarde heeft.}

3.4 Servers {optioneel}

De volgende servers zijn aanwezig:

Server	Beschrijving softwarecomponent	Ontwikkel	Test	Acceptatie	Productie
Databases					
{server}	{beschrijving}				
Applicatieservers					
{server}	{beschrijving}				

Server	Beschrijving softwarecomponent	Ontwikkel	Test	Acceptatie	Productie
Overige					
{server}	{beschrijving}				

{WAT STAAT IN DE OTAP-KOLOMMEN?}

3.5 Gebruikte URL's {optioneel}

Interne URL's

{Opsomming gebruikte interne URL's.}

Monitor-URL's

{Opsomming gebruikte monitor-URL's.}

Externe URL's

{Opsomming gebruikte externe URL's.}

3.6 Benodigde certificaten {optioneel}

De volgende certificaten zijn nodig:

- {certificaat}
- {certificaat}

Bijlagen

A Terminologie en afkortingen

De onderstaande tabel bevat afkortingen en termen die regelmatig voorkomen in ICTU-documentatie.

Term/afkorting	Toelichting
actor	Persoon die of extern systeem dat een handeling verricht op het systeem.
authenticatie	Vaststellen van de identiteit van een actor.
autorisatie	Aan een actor toegekende rechten.
BIO	Baseline Informatiebeveiliging Overheid
DoD	Definition of Done
DoR	Definition of Ready
GFO	globaal functioneel ontwerp
IPO	intern projectoverleg
ISR	ICTU Softwarerealisatie
Jira	Tool om use cases, user stories, logische testgevallen en issues vast te leggen.
KPI	key performance indicator
Minimum Viable Product	De eerste versie van een product of dienst, die zo vroeg mogelijk wordt uitgerold naar de opdrachtgever. Het bevat net voldoende functionaliteit om het gestelde doel te behalen, en niet meer dan dat.
MVP	Minimum Viable Product
OTAP	Ontwikkel, Test, Acceptatie, Productie
PID	projectinitiatiedocument
PSA	projectstartarchitectuur
PvE	programma van eisen
use case	Een afgebakende eenheid van interactie tussen een actor en het systeem.
VIR	Voorschrift Informatiebeveiliging Rijksdienst
VIR-BI	Voorschrift Informatiebeveiliging Rijksdienst Bijzondere Informatie

B Kwaliteitsaanpak ICTU Software Realisatie

Met de verregaande automatisering en digitalisering wordt software meer en meer belangrijk, ook binnen de overheid. Naast het op orde hebben van zaken als licenties van standaardsoftware, ligt er een uitdaging als het gaat om de ontwikkeling van maatwerksoftware. Projecten waarin software wordt ontwikkeld of onderhouden kampen nog vaak met vertraging, budgetoverschrijding of een eindresultaat met te lage kwaliteit. Zo concludeerde de commissie-Elias bijvoorbeeld in haar eindrapport: '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 erkend en gemitigeerd. Dit terwijl de risico's bij de ontwikkeling van maatwerksoftware, binnen het ICT-domein, inmiddels algemeen bekend zijn en er ook voor veel risico's passende maatregelen bestaan.

ICTU werkt sinds 2010 met de agile softwareontwikkelaanpak Scrum en heeft deze aanpak aangevuld en uitgebreid om zoveel mogelijk de kans op die risico's te verminderen. Denk hierbij aan geautomatiseerde regressietesten om het risico op fouten bij nieuwe opleveringen van de software (die bij Scrum elke twee of drie weken plaatsvinden) te voorkomen. Een ander voorbeeld is het zeer frequent – meerdere keren per uur - geautomatiseerd rapporteren over de kwaliteit van de software om zogenaamde 'technische schuld' te voorkomen.

Met behulp van de ICTU Kwaliteitsaanpak Software Realisatie 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. Daarom stelt ICTU deze kwaliteitsaanpak ter beschikking aan andere partijen en overheden die zelf maatwerksoftware ontwikkelen of dit laten doen.

De kwaliteitsaanpak heeft drie doelstellingen:

1 **Opdrachtgevers 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 softwareontwikkeling en -onderhoudsprojecten te vergroten. De maatregelen zijn beschreven in algemene termen; waar van toepassing is ook de ICTU-specifieke invulling van de maatregel telkens separaat bijgevoegd.

De beschrijving van de kwaliteitsaanpak is gebaseerd op de huidige aanpak van softwareontwikkeling en -onderhoud bij ICTU. De kwaliteitsaanpak evolueert op basis van praktijkervaringen bij ICTU en bij andere organisaties.