

edustandaard

OKE MBO-toetsafname

Afsprakenet op basis van AMIGO-aanpak en OOAPI

Versie: 23 januari 2024 (Concept 0.9.3)

Auteur: Jos van der Arend (Kennisset) i.s.m. kerngroep OKE-techniek

Bij de totstandkoming van deze afsprakenet zijn de volgende organisaties in de ketensamenwerking "OKE" (Onderwijs Koppelingen Examinering) betrokken:

Instellingen

Alfa college, Noorderpoort, ROC Amersfoort, ROC Amsterdam/Flevoland, ROC Midden Nederland, Summa college, Albeda

Examen-/toetsleveranciers

Centrale examens MBO (CEM), Stichting Praktijkleren (SPL), Bureau ICE (TOA)

Systeemleveranciers

CACI (Osiris), Paragin (RemindoToets), Oracle (Peoplesoft), Topicus (Eduarte), Visma Advitrac (Xedule), Xebic (OnTrack)

Ketencoördinatie

MBO digitaal, MBO Raad (Kennispunt MBO Onderwijs & Examinering), Netwerk Examinering en Digitalisering (NED), Surf, Kennisset



Inhoudsopgave

Inhoudsopgave	2
Documentgeschiedenis	3
1 Inleiding.....	6
1.1 Begrippenlijst.....	6
1.2 Bronnen	8
2 Scenario.....	9
2.1 Scenariobeschrijving.....	9
2.2 Processen.....	10
2.3 Uitgangspunten	11
2.4 Voorbeelden van systeemlandschap.....	12
3 Informatiestromen.....	14
3.0 Toetscatalogus informatiestroom	14
3.0.1 Informatieobject Toetscatalogus.....	15
3.0.2 Interacties voor Toetscatalogus	15
3.0.3 Koppelvlak bij Toetsafname	17
3.1 Toetsdeelnemers informatiestroom	18
3.1.1 Informatieobject Toetsdeelnemers.....	18
3.1.2 Interacties voor Toetsdeelnemers.....	20
3.1.3 Koppelvlak bij Deelnemerregistratie	28
3.1.4 Koppelvlak bij Toetsplanning.....	33
3.2 Zittingsplan informatiestroom.....	35
3.2.1 Informatieobject Zittingsplan	35
3.2.2 Interacties voor Zittingsplan.....	36
3.2.3 Koppelvlak bij Toetsafname	41
3.3 Toetsdeelnemerresultaat informatiestroom	44
3.3.1 Informatieobject Toetsdeelnemerresultaat	44
3.3.2 Interacties voor Toetsdeelnemerresultaat.....	45
3.3.3 Koppelvlak bij Toetsplanning.....	48
3.3.4 Koppelvlak bij Toetsafname	49
3.4 Zittingsverslag informatiestroom	50
3.4.1 Informatieobject Zittingsverslag.....	50
3.4.2 Interacties voor Zittingsverslag	51
3.4.3 Koppelvlak bij Toetsplanning.....	51
3.4.4 Koppelvlak bij Toetsafname	53
3.5 Studentresultaat informatiestroom	54

3.5.1	Informatieobject Studentresultaat.....	54
3.5.2	Interacties bij Studentresultaat.....	55
3.5.3	Koppelvlak bij Deelnemerregistratie.....	56
3.5.4	Koppelvlak bij Toetsplanning.....	57
4	Gegevensmodel	59
4.1	Component (Toets) object.....	63
4.2	ComponentOffering (Planbare toets & Zitting) object.....	65
4.3	ComponentOfferingAssociation (Toetsinschrijving & Deelname) object.....	67
4.4	CourseOffering (Opleidingsdeel) object.....	70
4.5	CourseOfferingAssociation (Deelinschrijving) object.....	72
4.6	Document (Document) object.....	73
4.7	Group (Toetsgroep) object.....	74
4.8	MembershipItem (Groepslidmaatschap) object.....	75
4.9	Organization (Organisatie) object.....	76
4.10	Person (Persoon) object.....	77
4.11	Program (Opleidingsprogramma) object.....	79
4.12	ProgramOffering (Opleiding) object.....	81
4.13	ProgramOfferingAssociation (Opleidingsinschrijving) object.....	83
5	Koppelvlakken.....	84
5.1	Deelnemerregistratie koppelvlak.....	84
5.2	Toetsplanning koppelvlak.....	85
5.3	Toetsafname koppelvlak.....	86

Documentgeschiedenis

Versie	Datum	Omschrijving
0.1	30 november 2022	Eerste beperkte concept-publicatie van specificaties van gegevensuitwisselingen t.b.v. MBO-toetsafname voor pilot 1.
0.2	7 december 2022	Detailuitwerkingen van flow 2, 3 en 4 bijgesteld n.a.v. specificatie op Github. Consistentie in uitwerking van Informatiestromen doorgevoerd en hoofdstukken 4 "Gegevensmodel" en 5 "Koppelvlakken" toegevoegd.
0.3	16 december 2022	Detailuitwerkingen van flow 2, 3 en 4 afgestemd op de actuele technische specificatie op Github. Laatste controles op correctheid uitgevoerd.
0.4	12 januari 2023	Correcties, verbeteringen en aanvullingen: <ul style="list-style-type: none"> • Herformulering van uitgangspunten en aanvulling van impliciet uitgangspunt 4 (zie Hsk 2) • Aanvullingen m.b.t. voorbeelden van systeemlandschap t.o.v. plattegrond (zie Hsk 2) • Correctie offeringsId naar offeringId in pad /offerings/{offeringsId} (zie Hsk 3 en 5) • Correctie componentId naar offeringId (zie Tabel 2.1 in Hsk 3)
0.5	29 maart 2023	Wijzigingen en aanvullingen t.b.v. pilot 1: <ul style="list-style-type: none"> • Gegevens rawScore en maxRawScore toegevoegd in Result binnen Association (paragraaf 4.1), • additionalTestingTime hernoemd (was extraTime) en safety toegevoegd in Component (4.2), • Enkele nieuwe resultaatsoorten aan enumeratie ResultValueType toegevoegd (4.3), • Gegeven preferredName toegevoegd aan Person en andere naamdelen verduidelijkt (paragraaf 4.4). Verduidelijkingen en correcties: <ul style="list-style-type: none"> • Scenario is stapsgewijs uitschreven, de uitgangspunten onderbouwd en de relatie van de processen naar de MORA en ROSA gelegd (in hoofdstuk 2) • BIV-classificatie van de gegevensobjecten en eigenaarschap beschreven (in hoofdstuk 4) • Introductie en uitleg over nut van OOAPI gegevensmodel (in hoofdstuk 6) • Diverse figuren bijgewerkt n.a.v. wijzigingen of gecorrigeerd

0.6	5 juni 2023	<p>Functionele aanvullingen:</p> <ul style="list-style-type: none"> Detailuitwerkingen van informatiestroom 1 Toetsdeelnemers toegevoegd (hoofdstuk 3, 4, 5 en 6), Detailuitwerkingen van informatiestroom 5 Studentresultaat toegevoegd (hoofdstuk 3, 4, 5 en 6), Waarde 'notknown' aan waardelijst AttendanceType toegevoegd in Result binnen Association (paragraaf 4.1) Attribuut gradeDate toegevoegd in Result binnen Association (paragraaf 4.1) <p>Verduidelijkingen en correcties:</p> <ul style="list-style-type: none"> Waarde van attribuut consumerKey in alle objecten is nu "NL-TEST-ADMIN" (hoofdstuk 3 en 4)
0.7	19 juni 2023	<p>Tekstuele aanvullingen en correcties:</p> <ul style="list-style-type: none"> Paragrafen in hoofdstuk 3 herkenbaar henummerd (hoofdstuk 3) Object Association vervangen door drie Component-, Course- en ProgramOfferingAssociation (hoofdstuk 3) Heel veel figuren en enkele tabellen bijgewerkt.
0.8	30 juni 2023	<p>Functionele aanvullingen/wijzigingen op de specificaties:</p> <ul style="list-style-type: none"> Operaties PATCH /offerings/{offeringId} en PATCH /associations/{associationId} toegevoegd voor verwijderen van planbare toets resp. toetsinschrijving in flow 1 (paragraaf 3.1.2 t/m 3.1.4) Waarden referenceLevelRKTR (RK Taal en Rekenen) en referenceLevelERK (ERK voor talen) in enumeratie ResultValueType toegevoegd (Tabel 4.1.D in paragraaf 4.1) <p>Tekstuele aanvullingen en correcties:</p> <ul style="list-style-type: none"> Verheldering dat Studentresultaat naar Deelnemerregistratie niet altijd een definitief resultaat is (hoofdstuk 2). PUT operatie op /offerings/{offeringId}/associations/{associationId} in flow 1 vervangen door /associations/{associationId} (paragraaf 3.1.2 t/m 3.1.4) Gegeven result is niet verboden in ComponentOfferingAssociation in flow 1 en 2, maar wordt niet gebruikt (paragraaf 3.1 en 3.2) PUT /offerings/{offeringId}/associations/{associationId} in flow 2 vervangen door PUT /associations/{associationId}, of verwijderen als die er al is (paragraaf 3.2.2 en 3.2.3) Waarden in enumeratie SafetyType consistent gemaakt (Tabel 4.1.C in paragraaf 4.1) Gegevens testSupplier en testId hernoemd tot testProvider en componentId (paragraaf 4.2) Gegeven finalResultsAllowed toegevoegd aan ComponentOffering (paragraaf 4.2) Gegeven gradeDate in result in ComponentOfferingAssociation is hernoemd tot testDate (paragraaf 4.3) Gegevens person en offering zijn verplicht gemaakt in ComponentOfferingAssociation (paragraaf 4.3) Gegevens organization is optioneel gemaakt in Group (paragraaf 4.6) Ter correctie, consumers in Person is verwijderd (paragraaf 4.7) Gegeven levelOfQualification in ProgramOfferingAssociation heeft waarde uit enumeratie conform OOAPI (paragraaf 4.10) Hoofdstuk 6 weggehaald; gaf te veel verwarring, nu direct verwijzingen in hoofdstuk 4 gemaakt. Veel figuren bijgewerkt.
	6 juli 2023	<p>Aanvullingen en correcties:</p> <ul style="list-style-type: none"> Gegevensobject Program toegevoegd aan gegevensmodel (hoofdstuk 4) Gegeven course verwijderd uit CourseOffering (paragraaf 4.4) Gegevensobject Program toegevoegd (paragraaf 4.8)
	10 juli 2023	<p>Verduidelijkingen en correcties:</p> <ul style="list-style-type: none"> ResultValueType toegevoegd aan Planbare toets in flow 1 (paragraaf 3.1.3 en 4.2) Term Cursus vervangen door Opleidingsdeel, en tevens Cursusinschrijving door Deelinschrijving.
0.9	21 juli 2023	<p>Functionele aanvullingen/wijzigingen op de specificaties:</p> <ul style="list-style-type: none"> In flow 1 kan wijziging in gegevens over opleidingsinschrijving en persoon direct naar TPL worden gestuurd. In flow 1 het opleidingsprogramma (Program) aangevuld met mogelijkheid om opleidingscode (vh crebo) te specificeren. In flow 5 kunnen resultaten van meerdere pogingen o.b.v. één toetsinschrijving in flow 1 worden teruggestuurd naar DR. In flow 5 de informatie over de zitting 5 uitgebreid met originele naam van de zitting. <p>Technische aanvullingen/wijzigingen op de specificaties:</p> <ul style="list-style-type: none"> Bij GET /offerings zijn de query parameters gewijzigd in offeringType en since aan flow 1 (paragraaf 3.1) PUT /associations/{associationId} uitgebreid naar opleidingsinschrijving in flow 1 (paragraaf 3.1) PUT /persons/{personId} toegevoegd aan flow 1 (paragraaf 3.1 en 5.2) Gegeven sequenceCode toegevoegd aan ProgramOfferingAssociation (paragrafen 3.1 en 4.10) PUT /associations/{associationId} toegevoegd voor meerdere pogingresultaten flow 5 (paragrafen 3.5 en 5.1) Gegeven attempt toegevoegd aan ComponentOfferingAssociation in flow 5 (paragraaf 3.5) Gegeven executedOfferingName toegevoegd aan Result (paragrafen 3.5 en 4.3) Gegeven organization geactiveerd aan ComponentOffering (paragraaf 4.2) Namen van attributen gewijzigd (Id toegevoegd) bij programOfferingAssociationId en courseOfferingAssociationId in ComponentOfferingAssociation (paragraaf 4.3) Gegeven orgAssociationId toegevoegd aan ComponentOfferingAssociation (paragraaf 4.3) Gegeven otherCodes geactiveerd in Program t.b.v. crebocode (paragraaf 4.8) <p>Tekstuele aanvullingen en correcties:</p> <ul style="list-style-type: none"> Benamingen van consumers objecten consistent gemaakt (hoofdstuk 3 en 4).
	24 augustus 2023	<p>Functionele aanvullingen/wijzigingen op de specificaties t.b.v. pilots 2 en 3:</p> <ul style="list-style-type: none"> Flow 1 uitgebreid met interacties om alle inschrijvingen van student/medewerker en contextinformatie van student aan te leveren en bij te werken. In flow 1 bij opvraging duidelijker gedefinieerd welke gegevens default worden afgeleverd, en hoe extra gegevens tegelijk op te vragen (expand parameter). In flow 1 mogelijk gemaakt onderscheid te maken tussen geannuleerde en beëindigde opleidingsinschrijving. In flow 1 en 2 studentgegevens uitgebreid met persoonlijke hulpmiddelen en de namen voor de legitimatiecontrole (naast voorkeursnamen en weergavenamen). <p>Technische aanvullingen/wijzigingen op de specificaties:</p> <ul style="list-style-type: none"> Algemene toelichting over verplichte en optionele gegevens m.b.t. GET, PUT, POST en PATCH (Hoofdstuk 4) Waarde "finished" in enumeratie AssociationStateType toegevoegd (Tabel 4.3.C in paragraaf 4.3) Gegevens personalNeeds en idCheckName binnen consumers toegevoegd aan Person (paragraaf 4.7)

		<ul style="list-style-type: none"> • Waarde "selfpaced" in enumeratie voor modeOfStudy gewijzigd in "self-paced" cf OOAPI/YAML (par. 4.8) • Gegevens cohort en location toegevoegd in ProgramOffering, verplaatst uit PgmOfAssociation (paragraaf 4.9) • Gegevens levelOfQualification, modeOfStudy, cohort en location verwijderd uit ProgramOfferingAssociation (paragraaf 4.10) <p><i>Kleinere tekstuele aanvullingen en correcties.</i></p>
0.9.1	22 september 2023	<p>Functionele aanvullingen/wijzigingen op de specificaties t.b.v. pilots 2 en 3:</p> <ul style="list-style-type: none"> • Enkele correcties en aanvullingen in de beschrijving van het scenario aangebracht (paragraaf 2.1) • Periode door parameters since en until is altijd gedefinieerd als begindatum tot en met einddatum (par. 3.1) • displayName en idCheckName binnen Person zijn nu gelijkvormig en zonder studentnummer (paragraaf 4.7) <p><i>Technische aanvullingen/wijzigingen op de specificaties:</i></p> <ul style="list-style-type: none"> • Gegeven associationType altijd aanwezig in Association, ook in PATCH (paragraaf 3.1, 3.2, 3.3 en 3.5) • Gegeven offeringType altijd aanwezig in Offering, ook in PATCH (paragraaf 3.1, 3.2 en 3.4) • Gegevens startDateTime en endDateTime in ComponentOffering als verplicht gecorrigeerd (paragraaf 4.2) <p><i>Kleinere tekstuele correcties, aanvullingen en aanpassingen t.b.v. consistentie en leesbaarheid.</i></p>
0.9.2	14 november 2023	<p><i>Technische correcties/verbeteringen van de specificaties:</i></p> <ul style="list-style-type: none"> • Gegeven abbreviation toegevoegd aan Component, conform OAS3-defs (paragraaf 4.1) • Gegeven resultExpected toegevoegd aan figuur 4.4.A, conform overige specs (paragraaf 4.4) • Object Document (Document) verplaatst uit algemene objecten naar eigen paragraaf (paragraaf 4.6) • Object Organization (Organisatie) toegevoegd (paragraaf 4.8) • Maximale lengte van namen in object Person gespecificeerd, conform OAS3-defs (paragraaf 4.9) • Waarde extraneous aan ModeOfStudyType in Program toegevoegd, conform OAS3-defs (paragraaf 4.10) <p><i>Kleinere tekstuele correcties, aanvullingen en verduidelijkingen m.b.t. te negeren attributen en vermeldingen van afwijking van OOAPI v5.</i></p>
0.9.3	10 januari 2024	<p>Functionele aanvullingen/wijzigingen op de specificaties:</p> <ul style="list-style-type: none"> • In informatiestroom 1 Toetsdeelnemers is: <ul style="list-style-type: none"> ◦ Interactie "Organisatieinfo ophalen" toegevoegd t.b.v. hiërarchische organisatiestructuren; door GET /organizations/{organizationId} bij DR/SIS. ◦ De interactie "Personen bij toetsgroep ophalen" is overgenomen door "Groepslidmaatschappen ophalen"; door toevoegen van GET /groups/{groupId}/members bij DR/SIS en object MembershipItem (Groepslidmaatschap) met o.a. de gegevens rol, status, begin- en eindmomenten; het eerder gespecificeerde GET /groups/{groupId}/persons verdwijnt hiermee uit deze specs. ◦ De lijst personalNeeds o.b.v. enumeratie bij student omgezet naar assignedNeeds met vrij te definiëren lijstitems met gegevens code, omschrijving, begindatum en einddatum. • In informatiestroom 2 Zittingsplan is het ophalen van de toegangslink toegevoegd middels GET /associations/{associationId}/url (conform github specs); Het eerder gespecificeerde attribuut startUpURL verdwijnt hiermee uit de specs van ComponentOfferingAssociation. <p><i>Technische correcties/verbeteringen/verhelderingen van de specificaties conform OAS3-defs:</i></p> <ul style="list-style-type: none"> • Informatiestromen zijn nu gedefinieerd aan de hand van het informatieobject, in overzichtsplaatje van scenario en in aparte paragrafen 3.x.1 over uitwerking informatieobject (hsk 3) • De waarde van attribuut consumerKey in alle objecten is nu "nl-test-admin", ook in figuren (hsk 3 en 4) • Bij PUT wordt altijd het object inclusief id meegegeven; dit is op betreffende plaatsen benadrukt (hsk 3 en 5). • GET /organizations/{organizationId} voorwaardelijk toegevoegd (paragraaf 3.1) • GET /groups/{groupId}/members toegevoegd en GET /groups/{groupId}/persons verwijderd (paragraaf 3.1) • GET /associations/{associationId}/url toegevoegd (paragraaf 3.2) • Figuren met gegevensobjecten per informatiestroom zijn vervangen door beschrijving (hsk 4) • Gegeven startUpURL verwijderd uit object ComponentOfferingAssociation (paragraaf 4.3) • Gegevens person en offering in CourseOfferingAssociation zijn verplicht gemaakt (paragraaf 4.5) • Object MembershipItem (Groepslidmaatschap) toegevoegd (hsk 4 en paragraaf 4.8) • Gegeven preferredName binnen Person (Person) verplaatst naar consumer (paragraaf 4.10) • Gegeven personalNeeds binnen Person (Person) vervangen door lijst assignedNeeds (paragraaf 4.10) • Gegeven program in ProgramOffering wordt aanbevolen (paragraaf 4.12) • Gegevens person en offering in ProgramOfferingAssociation zijn verplicht gemaakt (paragraaf 4.13) • Afspraak over het wel of niet gebruik van id in objecten bij GET, PUT en PATCH toegevoegd (hsk 5) <p><i>Kleinere tekstuele correcties, aanvullingen en verduidelijkingen, ook in figuren.</i></p>
	23 januari 2024	<ul style="list-style-type: none"> • Correctie van typefout in tabel-benaming van Tabel 4.8.A (paragraaf 4.8) • Correctie van typefout in gegeven assignedNeeds binnen Person in Tabel 4.10.A (paragraaf 4.10)

1 Inleiding

In het MBO worden toetsen en examens afgenomen door gespecialiseerde organisaties, de toets-/examenleveranciers. Voor de uitwisseling van studentgegevens en andere planningsgegevens naar de toetsleverancier, en voor de resultaatgegevens van toetsen en examens terug op zijn afspraken opgesteld. Deze afspraken zijn uitgewerkt in de bouwbare technische specificaties van de API zoals vastgelegd en beschreven in Github: <https://github.com/NetwerkExamineringDigitalisering/NED-OOAPI>.

In dit document worden de specificaties van deze afsprakenet met betrekking tot de geautomatiseerde uitwisselingen in algemeen begrijpbare en leesbare taal beschreven.

In het ontwikkeltraject van de afspraak is besloten dat deze afspraak een toepassingsprofiel op Open Onderwijs API (OOAPI) is. Uit dit profiel volgt automatisch de keuze voor REST/JSON en OAS 3.0 (of nieuwer). Voor meer informatie over OOAPI zie: <https://openonderwijsapi.nl>

1.1 Begrippenlijst

Bij de samenstelling van dit document zijn de volgende begrippen en definities uit het [ROSA Begrippenkader](#), het [MORA Informatiemodel \(MORA-IM\)](#) en [MORA Applicaties \(MORA-Apps\)](#) gebruikt (zie betreffende link voor naslag en laatste definitie):

[aan- en afwezigheid](#) = Geregistreerde aan- of afwezigheid van een student op een gepland tijdstip in een gespecificeerde omgeving. [bron: MORA IM]

[aangeboden opleiding](#) = Een opleidingseenheid die door een onderwijsaanbieder aangeboden wordt in een bepaalde vorm, al dan niet op een bepaalde onderwijslocatie.

[afnameplanning](#) (toetsafnameplanning) = Specificeert de toets, het moment van de afname, de kandidaten en de locatie.

[afpraak](#) = Overeenkomst binnen een bepaalde ketensamenwerking over de inrichting en het toepassen van bepaalde voorzieningen en/of standaarden.

[attribuut](#) = Een eigenschap, kenmerk of kwaliteit van een natuurlijke of rechtspersoon of een entiteit, in elektronisch formaat.

[bericht](#) = Een informatiedrager waarmee gegevens van een bron via een aanbieder aan een ontvanger worden overgedragen.

[beschikbaarheid](#) = De mate waarin gegevens worden opgeslagen volgens duurzame normen en afhankelijk van de organisatiekeuze beschikbaar gesteld aan verschillende afnemers.

[bevraging](#) = Een enkelvoudige vraag die door een serviceafnemer aan een serviceaanbieder wordt gesteld en waar direct een antwoord op wordt verwacht.

[bronhouder](#) (gegevensverstrekker) = De verantwoordelijke voor het beheer en het ter beschikking stellen van de juiste en actuele gegevens.

[crebo-code](#) = Verouderd begrip: de code van de opleiding binnen het Centraal Register Beroepsopleidingen (CREBO).

[eindpunt](#) (endpoint) = Virtuele locatie (url) van het koppelveld van een dienstverlener door een eindorganisatie is gekozen voor de levering van een of meer diensten of onderdelen daarvan.

[examen](#) = Een onderzoek naar kennis, inzicht, houding en vaardigheden van de student, op basis waarvan de instelling na beoordeling een formeel bewijs van prestatie toekent. [bron: MORA IM]

[examen deelname](#) = De registratie van een formele activiteit (lees examensessie) waarin een examen wordt afgenomen. [bron: MORA IM]

[examen logistiek systeem](#) = Een systeem voor de examenvoorbereiding zoals het creëren van examengroepen, toekennen examenmateriaal, aanmaken examensets, toekennen afnameleider en surveillanten [bron: MORA Apps]

[examenreglement](#) = Het examenreglement bevat in elk geval informatie over de maatregelen en de toepassing daarvan, regels met betrekking tot de organisatie van het examen en de gang van zaken tijdens het examen, en de herkansingsmogelijkheden van het examen.

[examenuitslag](#) = Het resultaat na het afleggen van het examen.

[formatieve beoordeling](#) = Een formatieve beoordeling (bijvoorbeeld: cijfer aanwezigheid beschrijving) dat niet meetelt voor de uiteindelijke kwalificering maar de lerende informatie geeft over de kwaliteit van zijn of haar leren. [bron: MORA IM]

[foutafhandeling](#) = Het verwerken van fouten in de afhandeling van een bericht.

gegeven = Weergave van een feit, begrip of aanwijzing, geschikt voor overdracht, interpretatie of verwerking door een persoon of apparaat.

gegevensafnemer = Een natuurlijke persoon of organisatie aan wie gegevens worden verstrekt.

gegevensverstrekker = Een natuurlijke persoon of organisatie die gegevens verstrekt.

identificatie = Het bekend maken van de identiteit van personen, organisaties of IT-voorzieningen.

identifier = Een label (meestal een string of tekst) waarmee je een entiteit aanduidt.

integriteit = De mate waarin de inhoud van een bericht of gegeven gegarandeerd kan worden.

interactiepatroon = Vormen van berichtuitwisseling tussen twee partijen.

Kernregistratie systeem studenten (KRS) = Een systeem voor het beheren van gegevens van studenten zoals in- en uitschrijvingsgegevens, BPV-contractgegevens, bekostigingsgegevens, studieresultaten, diploma's, studentbegeleidingsgegevens, studievoortganggegevens.

Bijvoorbeeld: Eduarte, Osiris, Magister, TP Ellucian (Sungard/SCT) Banner, Student Oracle Peoplesoft Campus Solutions. [bron: MORA Apps]

keten = Een samenwerkingsverband tussen organisaties die naast hun eigen doelstellingen, één of meer gemeenschappelijk gekozen (of door de politiek opgelegde) doelstellingen nastreven. Deze ketenpartners zijn zelfstandig, maar zijn ook afhankelijk van elkaar waar het gaat om het bereiken van de gezamenlijke doelstellingen.

ketensamenwerking = Een specifieke invulling van een ketenproces, waarin sprake is van het afhandelen van minimaal een ketenprocesstap in de vorm van een interactie tussen twee of meer ketenpartners.

klacht = Een uiting van ontevredenheid over een bewezen dienst, een persoon of een product met het oogmerk dat er een oplossing op dat punt zal worden gerealiseerd.

koppelvlak (API) = Een interface die uitwisseling van gegevens tussen informatiesystemen verzorgt.

leerresultaat = Vastgelegde beoordeling van een door een onderwijsdeelnemer uitgevoerde onderwijsactiviteit.

machine-to-machine = De gegevensuitwisseling tussen systemen onderling zonder menselijke tussenkomst.

mandaat = De bevoegdheid om in naam van een ander te handelen en gegevens te verwerken, maar zonder de daarbij horende verantwoordelijkheid.

mandatering = De bevoegdheid om in naam van een ander te handelen, maar zonder de daarbij horende verantwoordelijkheid.

OER = Het onderwijs- en examenreglement bevat het onderwijsprogramma en het examenreglement en -handboek. In het OER staan de regels en afspraken die binnen de instelling gelden omtrent het onderwijs, de examinering en diplomering. Ieder cohort valt onder een eigen (specifiek voor het cohort) OER. [bron: MORA IM]

onderwijsaanbod = Datgene wat een onderwijsorganisatie organiseert aan onderwijsactiviteiten zowel inhoudelijk als qua aanpak.

onderwijsdeelname = Een afspraak voor het gaan volgen, volgen en hebben gevolgd van onderwijs.

onderwijsdeelnemer = Een natuurlijk persoon die deelneemt aan een onderwijsactiviteit om zich kennis, vaardigheden en attitudes eigen te maken.

onderwijsinstelling = Een onderwijsgerelateerde organisatie met het aanbieden van onderwijs als doel.

onderwijsmedewerker = Een natuurlijk persoon die op grond van een overeenkomst werkzaam is voor een onderwijsorganisatie.

opleiding = Een samenhangend geheel van vakken, gericht op de verwezenlijking van welomschreven doelstellingen op het gebied van kennis, inzicht en vaardigheden.

opleidingseenheid = Een geheel van kennis, inzicht en vaardigheden.

opleidingsonderdeel = Een onderscheidbaar geheel van kennis, inzicht en vaardigheden, dat onderdeel kan zijn van één of meer opleidingen.

organisatorische eenheid = Een groep mensen die een gezamenlijk doel nastreeft.

profiel = Een specifieke invulling van een set van koppelvlakstandaarden die een groep functionele eisen invult.

referentiecomponent = Een functionele afbakening van een modulair, zelfstandig inzetbaar en vervangbaar (deel van een) systeem.

referentieschaal = Instrument om scores te plaatsen op een schaal.

resource – Een fysieke component of een eenheid van informatie waartoe een entiteit toegang zou kunnen krijgen.

restful api = API's die opgezet zijn op basis van REST-principes.

scenario = Een chronologische beschrijving ('draaiboek') van een bepaalde gebeurtenis (of reeks gebeurtenissen) die heeft plaatsgevonden of nog moet plaatsvinden.

student = Een persoon ingeschreven bij een instelling die een onderwijsovereenkomst heeft of wil gaan afsluiten. Dit omvat deelnemende studenten, aspirant studenten en alumni. [bron: MORA IM]

toets = Materiaal dat gebruikt wordt bij het bepalen van de voortgang in ontwikkeling van de kennis en competenties van een onderwijsdeelnemer.

toets- en examen afname systeem = Systeem voor het afnemen van toetsen en examens, denk aan een veilige afnameomgeving (BLDC - bootable lock down client).

Bijvoorbeeld: Safetyboot, Facet en QMP. [bron: MORA Apps]

toets- en examenplanning- en inschrijfsysteem = Systeem voor het inplannen van toetsen en examens, publiceren van de planning, inschrijven door studenten. . [bron: MORA Apps]

toetscatalogus = De toetscatalogus biedt detailinformatie over de toetsen die worden aangeboden. De informatie omvat onder andere de dagen en uren waarop de toetsen moeten worden afgenomen en de toegestane hulpmiddelen die hierbij gebruikt mogen worden.

toetsdeelnemer = Iemand die deelneemt aan een toetsafname.

toetsleverancier = Partij die een toets ter beschikking stelt.

uitschrijving = De uitschrijving bestaande uit onder andere de exitformulieren Een exitformulier is een document waarop de reden en eventueel omstandigheden van het vertrek van een deelnemer wordt vastgelegd. [bron: MORA IM]

vertrouwelijkheid = De mate waarin uitsluitend geautoriseerde personen, programmatuur of apparatuur gebruik kunnen maken van de gegevens of programmatuur, al dan niet gereguleerd door (geautomatiseerde) procedures en/of technische maatregelen.

verwerken = Een bewerking of een geheel van bewerkingen met betrekking tot persoonsgegevens of een geheel van persoonsgegevens, al dan niet uitgevoerd via geautomatiseerde procedés, zoals het verzamelen, vastleggen, ordenen, structureren, opslaan, bijwerken of wijzigen, opvragen, raadplegen, gebruiken, verstrekken door middel van doorzending, verspreiden of op andere wijze ter beschikking stellen, aligneren of combineren, afschermen, wissen of vernietigen van gegevens.

verwerker = Een natuurlijke persoon of rechtspersoon, een overheidsinstantie, een dienst of een ander orgaan die/ dat ten behoeve van de verwerkingsverantwoordelijke persoonsgegevens verwerkt.

1.2 Bronnen

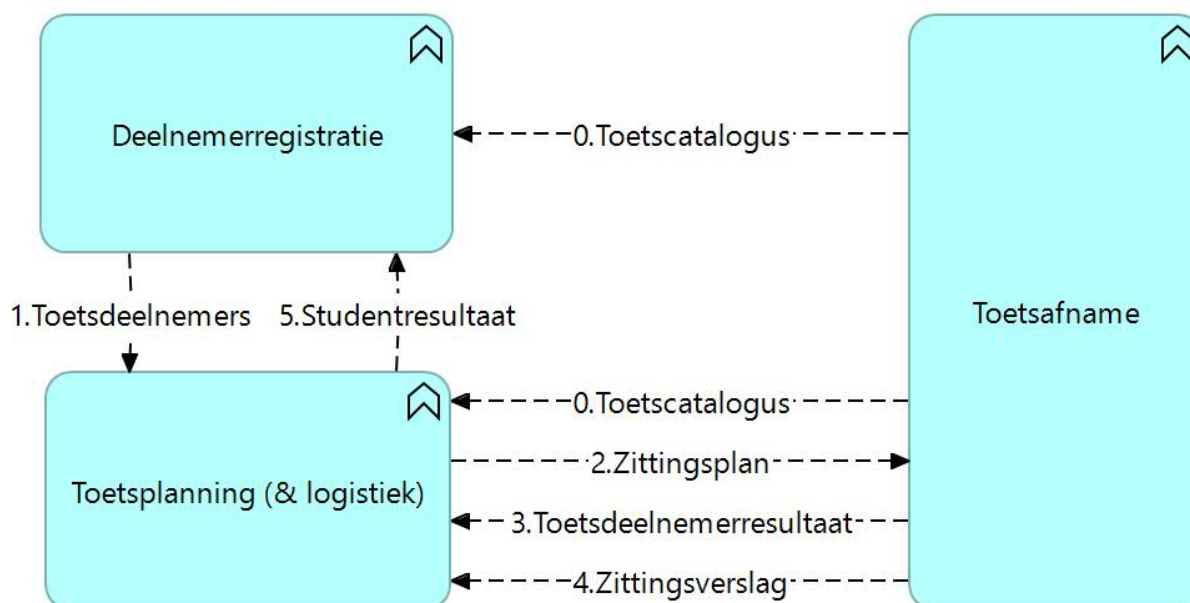
Bij de samenstelling van dit document zijn de volgende bronnen gebruikt:

- Document "Projectplan voor de PILOT van de OO-API", opgesteld door Hans Swart, versie 0.99 (November 2022)
- Specificatie "MBO-toetsafname" in Github repository, zie <https://github.com/NetwerkExamineringDigitalisering/NED-OOAPI>, inclusief:
 - Documentatie van specificatie "MBO-toetsafname" in Github repository, zie <https://netwerkexamineringdigitalisering.github.io/NED-OOAPI/>
 - Documentatie API-specificatie van "MBO-toetsafname" in Github repository, zie <https://netwerkexamineringdigitalisering.github.io/NED-OOAPI/specification/v5/docs.html>
 - API-specificatie (OAS3) van "MBO-toetsafname" in Github repository, zie <https://github.com/NetwerkExamineringDigitalisering/NED-OOAPI/blob/main/specification/v5/spec.yaml>
 - Werkdocumenten, zie <https://github.com/NetwerkExamineringDigitalisering/NED-OOAPI#working-documents>
- Open Education API (OOAPI), zie <https://openonderwijsapi.nl/#/>, inclusief:
 - Documentatie API-specificatie van Open Education API (5.0.0), zie <https://openonderwijsapi.nl/specification/v5/docs.html>
 - API-specificatie (OAS3) van Open Education API (5.0.0) in Github repository, zie <https://github.com/open-education-api/specification/tree/master/v5>
- Edustandaard afspraak Open Onderwijs API (OOAPI), zie https://www.edustandaard.nl/standaard_afspraken/open-onderwijs-api/
- Website "Netwerk Examinering en Digitalisering" (NED), zie <https://onderwijsenexaminering.nl/examinering/exameninstrumenten/>
- Website "Kennispunt MBO Onderwijs & Examinering", zie <https://onderwijsenexaminering.nl/>
- Website "MBO Raad", zie <https://www.mboraad.nl/>

2 Scenario

2.1 Scenariobeschrijving

Het scenario-plaatje is als volgt:



Figuur 2.A: Scenario van de ketensamenwerking “Toetsafname in het MBO examenproces”

In dit scenario zijn de volgende functionaliteiten betrokken:

- **Deelnemerregistratie**, ook wel SIS (Student Informatiesysteem) genoemd en afgekort tot **DR/SIS**, is waar studenten en diens inschrijvingen van betreffende toetsen zijn geregistreerd. Belangrijk hierbij is het beheer van de toegestane pogingen bij deze toets en de registratie van de context per toetsinschrijving, zoals programma, aanbod en inschrijving van betreffende opleidingsdeel en opleiding,
- **Toetsplanning (& logistiek)**, afgekort tot **TPL**, waar de afnameplanning voor de toets/examen wordt samengesteld op basis van informatie uit Deelnemerregistratie (1. Toetsdeelnemers), zoals of er meerdere pogingen zijn toegestaan, en informatie uit Toetsafname (3. Toetsdeelnemerresultaat), zoals het aanwezigheidsinformatie (wel/niet deelgenomen) en toetsresultaat, en
- **Toetsafname**, afgekort **TAS**, waar de toetsafname van de groep van studenten plaatsvindt en resultaten ontstaan.

De stappen in dit scenario zijn als volgt:

- In **Deelnemerregistratie** worden opleidingen, opleidingsdelen en bijbehorende toetsing ingericht met behulp van de toetsinstrumenten zoals beschreven in de toetscatalogus (**Informatiestroom 0 Toetscatalogus**). Studenten schrijven zich in voor een bepaalde opleiding en zullen daarmee de toetsen binnen deze opleiding gaan volgen.
- Op enig moment wordt de informatie over een toets en de potentiële toetsdeelnemers (studenten) en medewerkers aangeleverd bij Toetsplanning (**Informatiestroom 1. Toetsdeelnemers**) om de toets (zitting) in te plannen.
- In **Toetsplanning** wordt op basis van deze informatie een zitting gepland, d.w.z. een toetsafname op een bepaald moment op een bepaalde locatie. Het kan voorkomen dat de studenten bij aanlevering uit Deelnemerregistratie al zijn aangemeld voor de zitting. Het kan ook voorkomen dat iedere student zich individueel voor een zitting moet aanmelden; de student kiest hierbij voor de meest geschikte zitting.
- Bij voldoende aanmeldingen zal de informatie over de geplande zitting (het zittingsplan), inclusief studenten en medewerkers (in de rol van afnameleider, surveillant en/of beoordelaar) bij Toetsafname worden aangeleverd (**Informatiestroom 2. Zittingsplan**).

Eventueel wordt een zittingsplan naderhand aangevuld (bijvoorbeeld met extra deelnemers of medewerkers) of misschien zelfs verwijderd (als de zitting komt te vervallen).

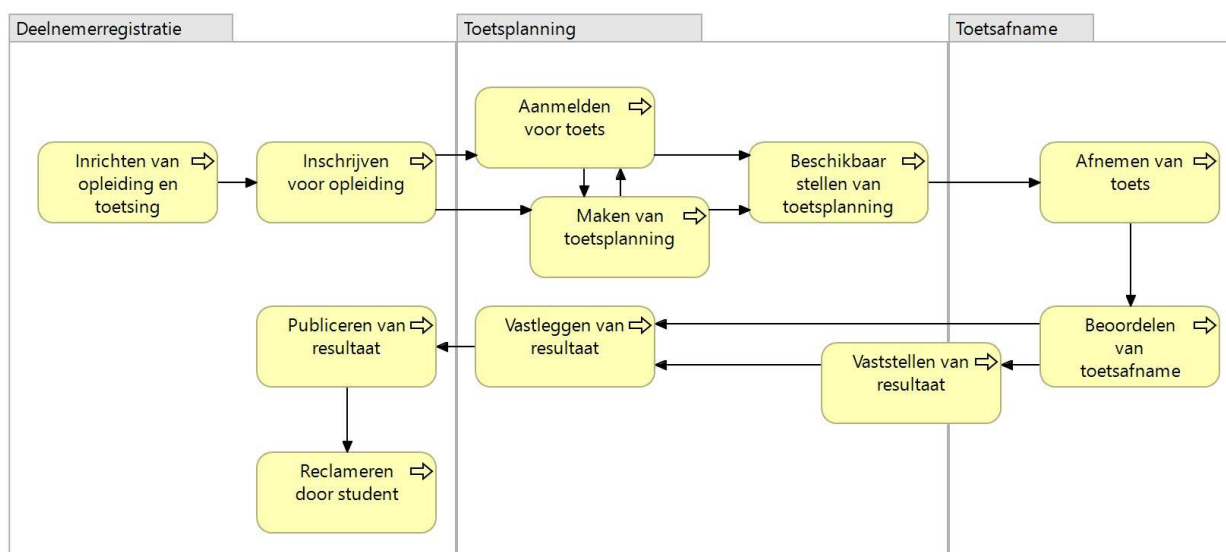
- E. **Toetsafname** verwerkt de aangeleverde informatie over het zittingsplan.
- F. Op het moment dat de zitting plaatsvindt zal de toets worden afgenomen onder leiding van de afnameleider. Bij iedere toetsafname wordt de aanwezigheid van de studenten geregistreerd; onregelmatigheden worden genoteerd in het 'proces verbaal'.
In het geval van synchrone beoordeling van de toetsafname (zoals bij praktijktoets) zal de beoordelaar direct de beoordeling maken. Het berekenen van het resultaat (zoals GVO of toetscijfer) zal direct plaatsvinden. In het geval van asynchrone beoordeling (zoals bij schriftelijke toets) zal de beoordeling achteraf worden uitgevoerd. De beoordeling omvat veelal het analyseren van de antwoorden en omzetten naar behaalde scorepunten. En daarna kan het berekenen van het resultaat (zoals het cijfer) pas plaatsvinden.
Indien mogelijk zal de examencommissie de individuele resultaten vaststellen.
- G. Direct nadat de zitting heeft plaatsgevonden wordt door **Toetsafname** de informatie over de aanwezigheid per student bij Toetsplanning aangeleverd; het resultaat van de beoordeling wordt indien mogelijk direct meegeleverd (**Informatiestroom 3. Toetsdeelnemerresultaat**). Het is belangrijk de aanwezigheidsinformatie (en eventueel het toetsresultaat) van de studenten zo spoedig mogelijk over te dragen; het is mogelijk direct van invloed op het opnieuw mogen aanmelden voor een volgende geplande zitting.
Direct na de zitting wordt ook het zittingsverslag bij Toetsplanning aangeleverd (**Informatiestroom 4. Zittingsverslag**).
- H. In het geval van asynchrone beoordeling, zal het resultaat van de toetsdeelnemer bij Toetsplanning worden nageleverd zodra het resultaat van deze toetsdeelnemer bekend is (informatiestroom 3).
- I. **Toetsplanning** legt de van Toetsafname ontvangen informatie over de zitting (toetsresultaten, zittingsverslag en bewijslastdocumenten) vast.
Zolang een studentresultaat nog niet is vastgesteld door de examencommissie zal Toetsplanning in veel gevallen wachten met de verdere verwerking van dit resultaat. Het vaststellen van de resultaten is mogelijk van invloed op het opnieuw mogen aanmelden van de student voor een volgende zitting.
- J. Zodra het studentresultaat is vastgesteld, zal dit resultaat met bewijslastdocumenten bij Deelnemerregistratie worden aangeleverd (**Informatiestroom 5. Studentresultaat**). Indien nodig, kan het studentresultaat ook worden aangeleverd bij Deelnemerregistratie als het nog niet is vastgesteld; Deelnemerregistratie biedt daarom ondersteuning om te kunnen omgaan met niet-vastgestelde studentresultaten.
In het geval dat door Deelnemerregistratie meerdere pogingen zijn toegestaan, kan Toetsplanning het studentresultaat als resultaat van een poging bij Deelnemerregistratie aanleveren en de student opnieuw aanmelden voor een volgende poging. Het is dus mogelijk om op basis van één toetsinschrijving in informatiestroom 1, meerdere toetsresultaten (van pogingen) over te dragen en te verwerken, waarbij Deelnemerregistratie altijd de regie over de pogingen heeft.
- K. **Deelnemerregistratie** verwerkt het aangeleverde studentresultaat, zodat ook publicatie naar de student mogelijk is. De student heeft dan de mogelijkheid om bezwaar te maken (te reclameren); de eventuele gevolgen van dit bezwaar worden in Deelnemerregistratie verwerkt.

2.2 Processen

In de drie functionele onderdelen Deelnemerregistratie, Toetsplanning (& logistiek) en Toetsafname worden processen uitgevoerd gerelateerd aan het scenario. Te weten:

- Inrichten van opleiding en toetsing
- Inschrijven voor opleiding
- Aanmelden voor toets
- Maken van toetsplanning
- Beschikbaar stellen van toetsplanning
- Afnemen van toets
- Beoordelen van toetsafname
- Vaststellen van resultaat
- Vastleggen van resultaat
- Publiceren van resultaat
- Reclameren door student

Deze processen zijn voor examens gedefinieerd en uitgewerkt in de [MORA](#), met name in de [procesketen Examineren](#). Voor formatieve toetsen valt dit voornamelijk onder het proces “Verzamelen leerresultaten”. De verdeling van deze processen over de functionele onderdelen zijn schematisch weergegeven in onderstaand Figuur 2.B.



Figuur 2.B: Scenario van de ketensamenwerking “Toetsafname in het MBO examenproces”

[ROSA](#) is de ketenreferentiearchitectuur voor het onderwijsdomein. De processen in bovenstaand figuur bevinden zich in de ROSA binnen het [ketenproces Toetsen](#) binnen het [ROSA ketendomein Examineren en Toetsen](#). In dit ketenproces Toetsen worden de volgende deelprocessen onderscheiden: aanmelden van de deelnemers, plannen van de toetsafname, afnemen van de toets, analyseren van de toetsafnames en presenteren van de toetsresultaten.

2.3 Uitgangspunten

Rondom dit scenario zijn de volgende uitgangspunten vastgesteld:

1. De gegevensuitwisselingen vinden altijd plaats namens een specifieke organisatorische eenheid binnen een mbo-onderwijsinstelling. Deze organisatorische eenheid kan de gehele mbo-instelling betreffen of een bepaalde locatie of afdeling binnen een mbo-instelling (de school).
2. Eén systeem/applicatie kan meerdere functionaliteiten vervullen. Er kunnen binnen een school systemen in gebruik die de functionaliteit Deelnemerregistratie met Toetsplanning omvatten en/of systemen die Toetsplanning met Toetsafname omvatten.
3. Binnen een school kunnen meerdere systemen dezelfde functionaliteit vervullen. Het kan voorkomen dat een school één systeem heeft voor Deelnemerregistratie maar dat binnen een school verschillende systemen voor Toetsplanning worden gebruikt afhankelijk van de toets/examen of de organisatorische eenheid binnen de school.
4. Ieder systeem/applicatie ondersteunt alle uitwisselingen die bij de betreffende functionaliteiten behoren, in de betreffende rol van de interacties.

Voorbeeld 1: Een systeem met Deelnemerregistratie ondersteunt informatiestromen 0 en 1 & 5 in die rol.

Voorbeeld 2: Een systeem met Deelnemerregistratie én Toetsplanning ondersteunt de informatiestromen 0 en 1 & 5 in de rol van Deelnemerregistratie en de informatiestromen 0 en 1 & 5 en 2, 3 & 4 in de rol van Toetsplanning.

Voorbeeld 3: Een systeem met Toetsplanning ondersteunt de informatiestromen 0 en 1 & 5 en 2, 3 & 4 in die rol.

Voorbeeld 4: Een systeem met Toetsplanning én Toetsafname ondersteunt de informatiestromen 0 en 1 & 5 en 2, 3 & 4 in de rol van Toetsplanning en de informatiestromen 0 en 2, 3 & 4 in de rol van Toetsafname.

Voorbeeld 5: Een systeem met Toetsafname ondersteunt de informatiestromen 0 en 2, 3 & 4 in die rol.

5. De systemen die betrokken zijn bij de informatiestromen 1 & 2 van de heenweg (heenroute), worden ook betrokken bij de informatiestromen 3, 4 & 5 van de terugweg; d.w.z. er worden geen andere systemen bij de terugweg betrokken.

Met andere woorden, de systemen die betrokken zijn bij informatiestroom 1 ten behoeve van een bepaalde toetsafname, zijn de systemen die betrokken zijn bij bijbehorende informatiestroom 5 na afloop van de toetsafname. En, de systemen die betrokken zijn bij informatiestroom 2 ten behoeve van een toetsafname, zijn de systemen die betrokken zijn bij de bijbehorende informatiestromen 3 en 4 na afloop van de toetsafname.

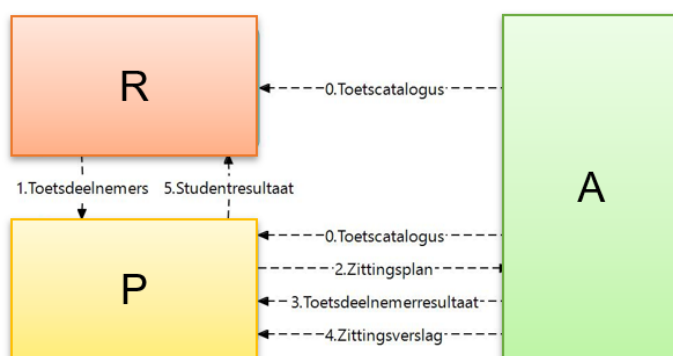
6. Deze gegevensuitwisselingen betreffen de volgende soorten toetsen:

- Dit betreft examens én toetsen, zowel summatieve als formatieve toetsen
- Dit betreft kennistoetsen én praktijktoetsen (zoals proeve van bekwaamheid, werkstuk of praktijkopdracht)
- Dit betreft online (digitale) én offline toetsen (zoals schriftelijke toets)

Buiten scope zijn toetsen/examens in het VO (voortgezet onderwijs) en voornamelijk de analyse van toetsresultaten.

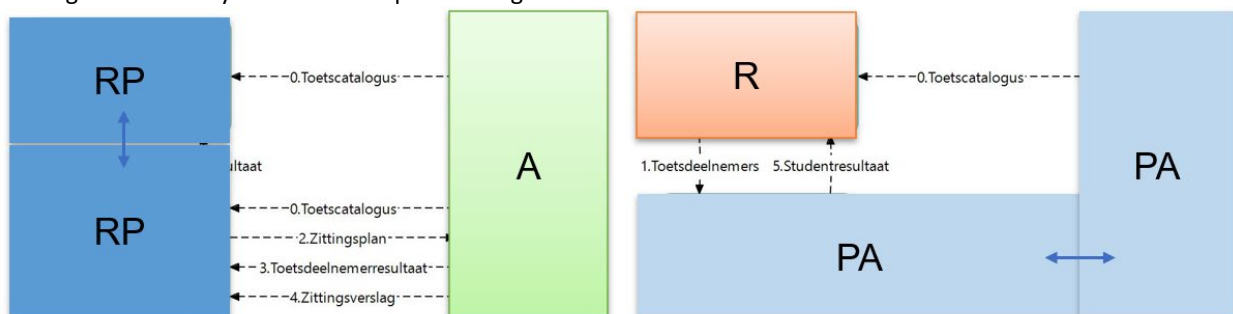
2.4 Voorbeelden van systeemlandschap

Bij specifieke systemen/applicaties per functionaliteit, d.w.z. Deelnemerregistratie (systeem R), Toetsplanning & Logistiek (systeem P) en Toetsafname (systeem A), ziet het systeemlandschap er als volgt uit:



Figuur 2.C: Systeemlandschap R+P+A bij enkelvoudige functionaliteit per systeem.

Als gevolg van bovenstaand uitgangspunt 2 zouden meerdere functionaliteiten in 1 systeem kunnen zijn verenigd, d.w.z. Deelnemerregistratie met Toetsplanning (systeem RP) of Toetsplanning met Toetsafname (systeem PA). Vervolgens zou het systeemlandschap er als volgt kunnen uitzien:

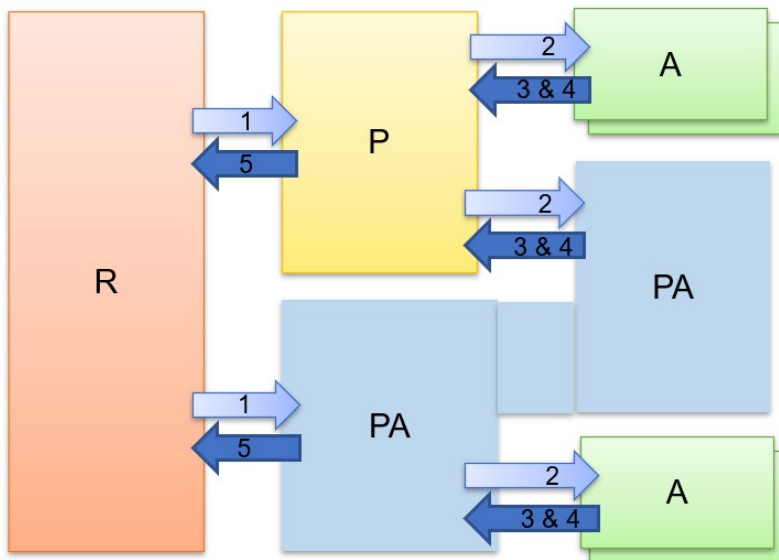


Figuur 2.D: Systeemlandschap RP+A en R+PA bij meervoudige functionaliteit per systeem.

Als gevolg van bovenstaand uitgangspunt 3 zouden functionaliteiten in het systeemlandschap door meerdere systemen kunnen worden ondersteund, met name meerdere systemen met Toetsplanning functionaliteit en meerdere systemen met Toetsafname functionaliteit. Dit betekent dat als Toetsplanning binnen de school door twee systemen wordt ondersteund dan zal zich één van de volgende situaties voordoen:

- Systeem RP (Deelnemerregistratie & Toetsplanning) én Planningssysteem P,
- Systeem RP (Deelnemerregistratie & Toetsplanning) én systeem PA (Toetsplanning & Toetsafname),
- Planningssysteem P1 én Planningssysteem P2,
- Planningssysteem P én systeem PA (Toetsplanning & Toetsafname).

Als gevolg van uitgangspunt 4 moet bijvoorbeeld een systeem PA vanuit de twee functionaliteiten alle informatiestromen ondersteunen, zie volgende figuur 2.E.



Figuur 2.E: Systeemlandschap bij twee systemen P+PA met Toetsplanning-functionaliiteit.

Merk op dat systeem PA bij informatiestromen 2 en 3 & 4 in een dubbelrol zowel aan de kant van Toetsafname van de uitwisselingen kan opereren als aan de kant van Toetsplanning kan opereren.

Als gevolg van bovenstaand uitgangspunt 5 wordt in het voorgaande figuur geschetste systeemlandschap de vier mogelijkheden voor heenroute en terugroute:

- heenroute: R-P-A & terugroute: A-P-R
- heenroute: R-P-PA & terugroute: PA-P-R
- heenroute: R-PA & terugroute: PA-R
- heenroute: R-PA-A & terugroute: A-PA-R

Informatiestromen

Het scenario bestaat uit de volgende informatiestromen:

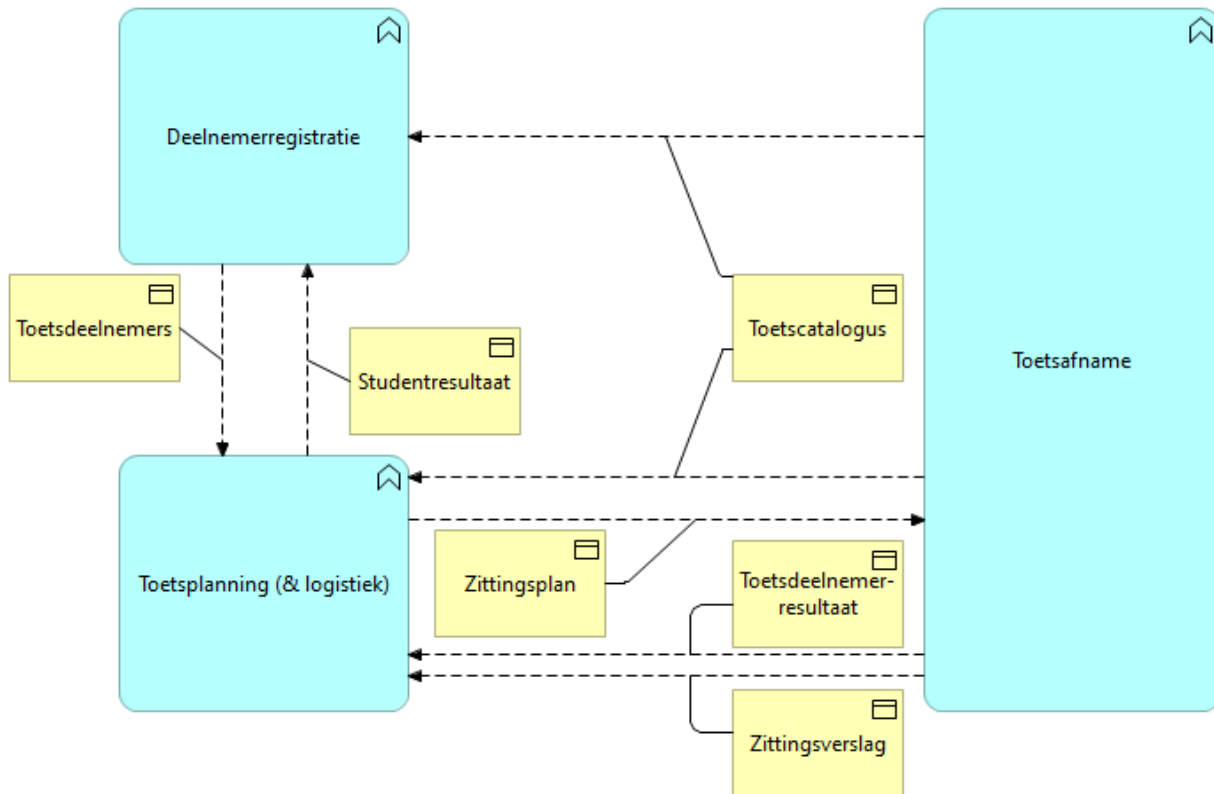
0. Toetscatalogus
1. Toetsdeelnemers
2. Zittingsplan
3. Toetsdeelnemerresultaat
4. Zittingsverslag
5. Studentresultaat

In een later stadium is met lagere prioriteit de zevende informatiestroom “Analyse-informatie” geïdentificeerd. Deze informatiestroom zal eventueel later worden toegevoegd en uitgewerkt.

In het volgende hoofdstuk worden de zes informatiestromen 0 t/m 5 één-voor-één beschreven.

3 Informatiestromen

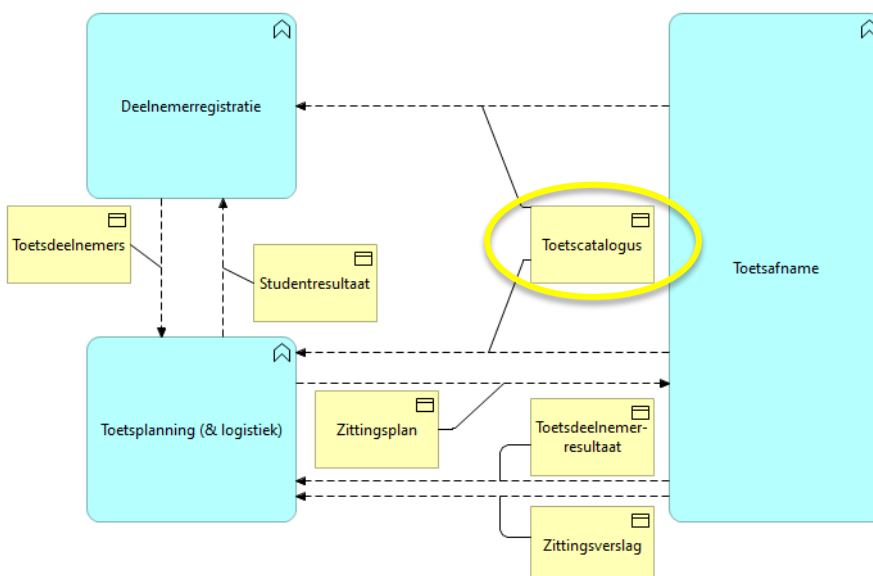
In het scenario met de functionaliteiten en informatiestromen (zie Figuur 2.A in Hoofdstuk 2) wordt per informatiestroom een specifiek informatieobject uitgewisseld. Schematisch ziet dat er dan als in onderstaand figuur uit.



Figuur 3.A: Informatieobjecten in de informatiestromen

3.0 Toetscatalogus informatiestroom

De beide informatiestromen Toetscatalogus (flow 0) betreft het informatieobject Toetscatalogus dat in onderstaand figuur met gele ovaal is gemarkeerd.



Figuur 3.0.1A: Informatieobject Toetscatalogus in het scenario

Ten behoeve van de processen “Inrichten van opleiding en toetsing” en “Inschrijving voor opleiding” (zie Figuur 2.B in Hoofdstuk 2) heeft **Deelnemerregistratie** behoefte aan informatie over de beschikbare toetsen en examens.

Toetsplanning (& logistiek) heeft eveneens behoefte aan deze informatie om de processen “Aanmelden voor toets”, “Maken van toetsplanning” en “Beschikbaar stellen van toetsplanning” te kunnen uitvoeren. Dit informatieobject Toetscatalogus over de beschikbare toetsen en examens wordt samengesteld en gepubliceerd door de toetsleverancier in Toetsafname en overgedragen in informatiestroom **0. Toetscatalogus**.

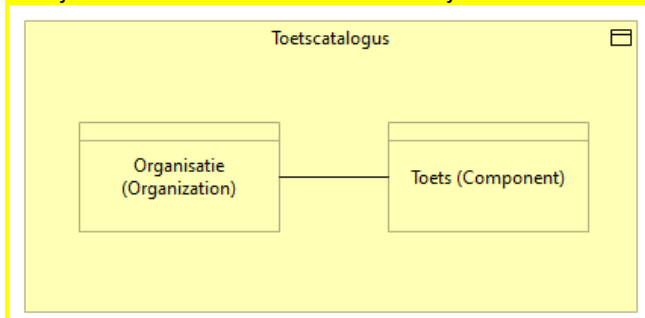
Gegevensverstrekker van Informatieobject Toetscatalogus met informatie over de beschikbare toetsen is het systeem/de applicatie van de toetsleverancier met de Toetsafname-functionaliteit. De gegevensafnemer zijn de systemen met de Deelnemerregistratie- en/of Toetsplanning-functionaliteit. De informatie over de Toetscatalogus gaat van Toetsafname (TAS) naar Deelnemerregistratie en Toetsplanning (& logistiek).

In de overdracht van de Toetscatalogus van TAS naar Deelnemerregistratie of Toetsplanning neemt Deelnemerregistratie c.q. Toetsplanning het initiatief. TAS zorgt ervoor dat de catalogus voor de betreffende aanvrager wordt samengesteld en teruggeleverd.

Voordat de gegevensuitwisseling kan plaatsvinden zijn Deelnemerregistratie of Toetsplanning en TAS al technisch met elkaar verbonden, m.b.t. base-URL, authenticatie etc..

3.0.1 Informatieobject Toetscatalogus

In het volgende figuur staan de informatieobjecten binnen het informatieobject Toetscatalogus, met tussen haakjes de naam van het betreffende object binnen OOAPI.



Figuur 3.0.1B: Informatieobject Toetscatalogus in detail

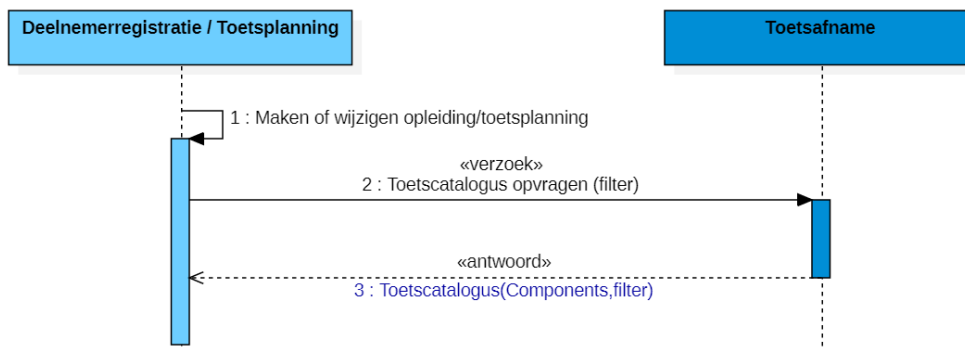
Het **informatieobject Toetscatalogus** bevat de informatie over iedere toets (of examen) die de leverancier beschikbaar heeft voor de onderwijsinstelling van het ontvangende systeem met Deelnemerregistratie en/of Toetsplanning functionaliteiten. De **Toetscatalogus** identificeert iedere toets en geeft de betreffende kenmerken die nodig zijn om de toets in te zetten in de opleiding en in te plannen voor afname. De unieke identifier van de toets wordt later gebruikt bij de aanlevering van het Zittingsplan (afnameplanning) om de betreffende toets voor afname te identificeren.

3.0.2 Interacties voor Toetscatalogus

Het algemene transactiepatroon is “Pull” zodat de gegevens bij Toetsafname (TAS) worden opgehaald. Deze informatiestroom bevat de interacties “Toetscatalogus ophalen” en “Toetsdetails ophalen”

Interactie “Toetscatalogus ophalen”

Deelnemerregistratie en/of Toetsplanning vraagt de **Toetscatalogus** op bij TAS. Voor het ophalen van de Toetscatalogus kan eventueel een filter worden gebruikt om de resultaatlijst van toetsen/examens in te perken. De interactie ziet er als volgt uit:



Figuur 3.0.2: Toetscatalogus opvragen interactiemodel

Deze overdracht van de Toetscatalogus is een **standlevering**, d.w.z. iedere keer worden alle gegevens volgens de actuele stand van zaken overgedragen. Deze interactie is daarmee **herhaalbaar** op elk gewenst moment. Om te bepalen of een levering een wijziging op eerdere aanlevering is zal Deelnemerregistratie of Toetsplanning de gegevens moeten vergelijken. Deze identificatie van de toets is hierbij leidend: als de id van de toets overeenkomt met de id van eerdere aanlevering dan betreft dit dezelfde toets. Eventueel kan een toets vervallen door de status hiervan op “inactive” te zetten.

OOAPI operatie en endpoint

Operatie en endpoint binnen OOAPI zijn bij de interactie “Toetscatalogus opvragen”:

GET /components

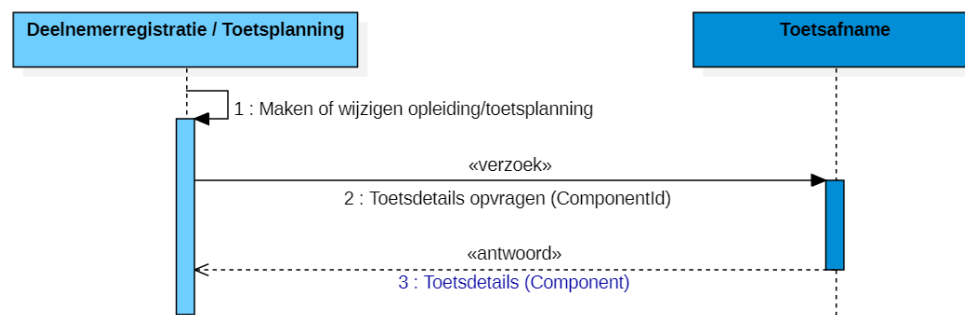
In het verzoek “2: Toetscatalogus opvragen” (zie Figuur 3.0.2) wordt de filtering gespecificeerd door query-parameters, bijvoorbeeld de (combinatie van) filters:

- componentType=test, bij verzoek om componenten van gevraagde type “Toets/examen”,
- q=zoektekst, bij verzoek om toetsen met zoektekst in de naam of beschrijving, bijvoorbeeld “q=rekenen”,
- since=zoekdatum, bij verzoek om toetsen met als startdatum de zoekdatum, bijvoorbeeld “since=2023-01-12”,
- license=true, bij verzoek om toetsen waarvoor de organisatie-eenheid een licentie heeft.

Het **antwoordbericht “3: Toetscatalogus”** bevat bij succesvolle aanroep een lijst van toetsen (**object Component**).

Interactie “Toetsdetails ophalen”

Voor de validatie van de actualiteit van de gegevens over een toets/examen kan Deelnemerregistratie of Toetsplanning de details van een bepaalde toets/examen uit de **Toetscatalogus** opvragen bij TAS. Voor het ophalen van de Toetsdetails moet de ID van de toets worden meegegeven bij de opvraging. De interactie ziet er als volgt uit:



Figuur 3.0.3: Toetsdetails opvragen interactiemodel

OOAPI operatie en endpoint

Operatie en endpoint binnen OOAPI zijn bij de interactie “Toetsdetails opvragen”:

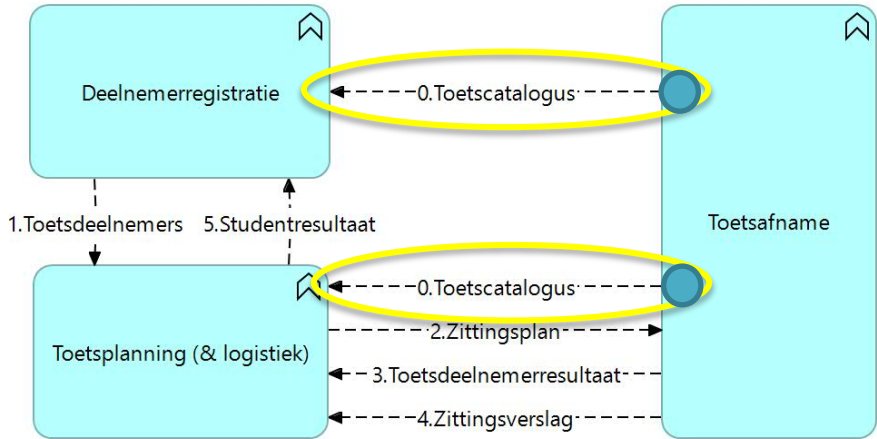
GET /components/{componentId}

In het **verzoek “2: Toetsdetails opvragen”** (zie Figuur 3.0.3) wordt de specifieke toets geïdentificeerd door componentId als padparameter.

Het **antwoordbericht “3: Toetsdetails”** bevat bij succesvolle aanroep de gegevens van de toets (**object Component**).

3.0.3 Koppelvlak bij Toetsafname

Het koppelvlak bij Toetsafname (TAS) voor informatiestroom Toetscatalogus is in onderstaand figuur met blauwe cirkel gemarkeerd.



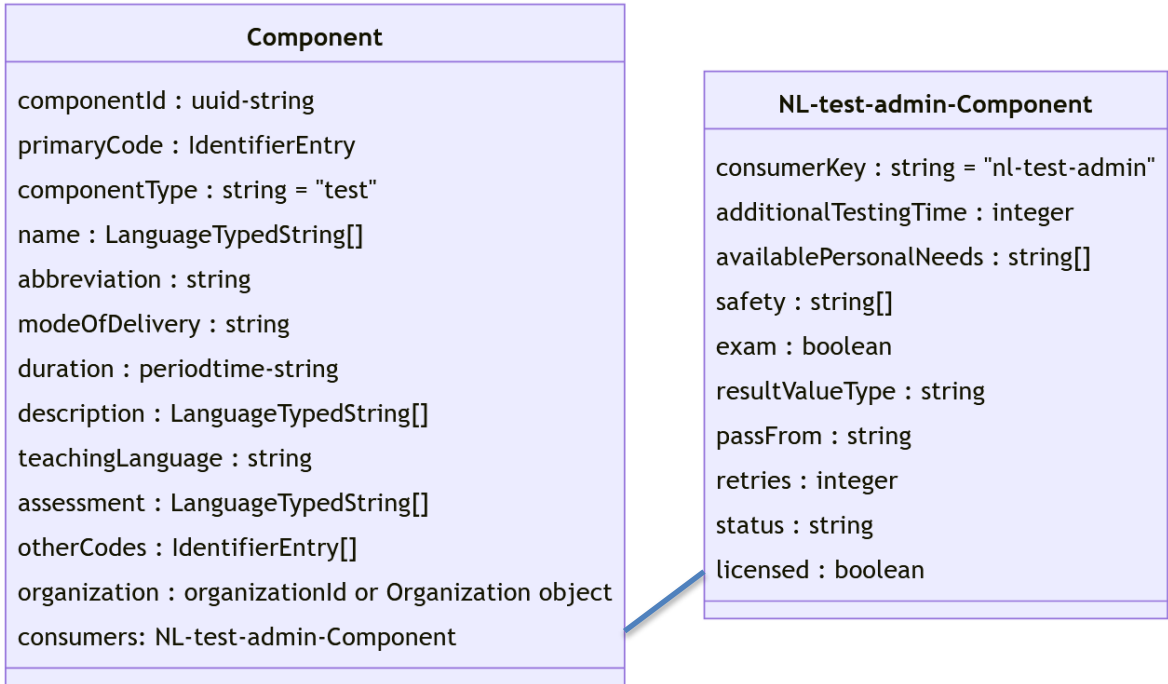
Figuur 3.0.4: Koppelvlak bij Toetsafname voor Toetscatalogus

Operatie & Endpoint	Verzoekbericht	Antwoordbericht
GET /components	Filterparameter: componentType=test q={zoektekst} in name of description since={datum} voor startdatum vanaf licensed={true/false} voor wel/niet geldige schoollicentie	Inhoud Lijst van Toetsen (Component), zie Figuur 3.0.5
GET /components/{componentid}	Padparameter: componentid=uuid	Inhoud Toets (Component), zie Figuur 3.0.5)

Tabel 3.0.1: Endpoints en operaties in koppelvlak bij Toetsafname voor Toetscatalogus

De aanvrager (verzoeker/initiator) van beide endpoint-operaties in bovenstaande tabel zijn Deelnemerregistratie of Toetsplanning (& logistiek).

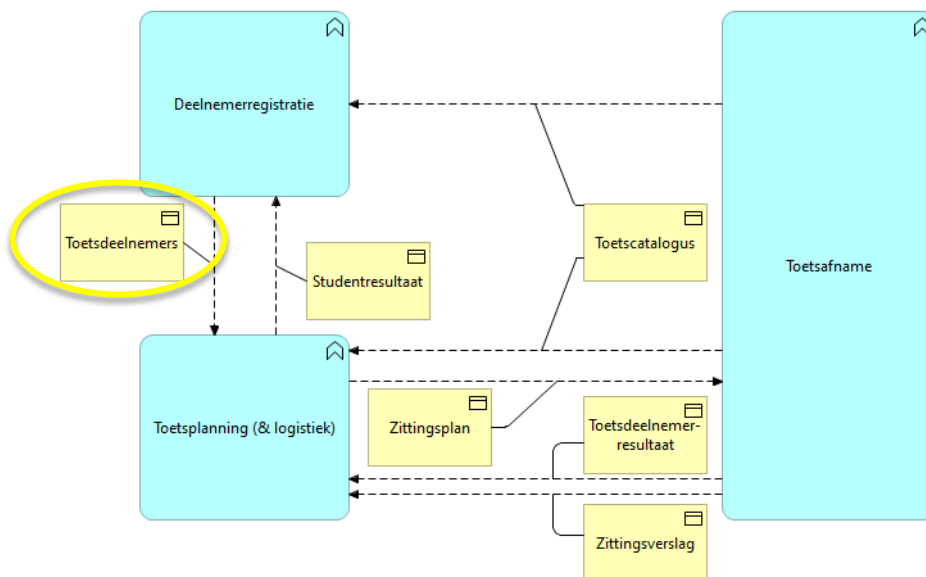
De antwoordberichten van beide operaties **GET /components** en **GET /components/{componentid}** bevatten het **object Component**, zie onderstaand figuur.



Figuur 3.0.5: Component (Toets) gegevensmodel in 0. Toetscatalogus

3.1 Toetsdeelnemers informatiestroom

De informatiestroom Toetsdeelnemers (flow 1) betreft het informatieobject Toetsdeelnemers dat in onderstaand figuur met gele ovaal is gemarkeerd.



Figuur 3.1.1A: Informatieobject Toetsdeelnemers in het scenario

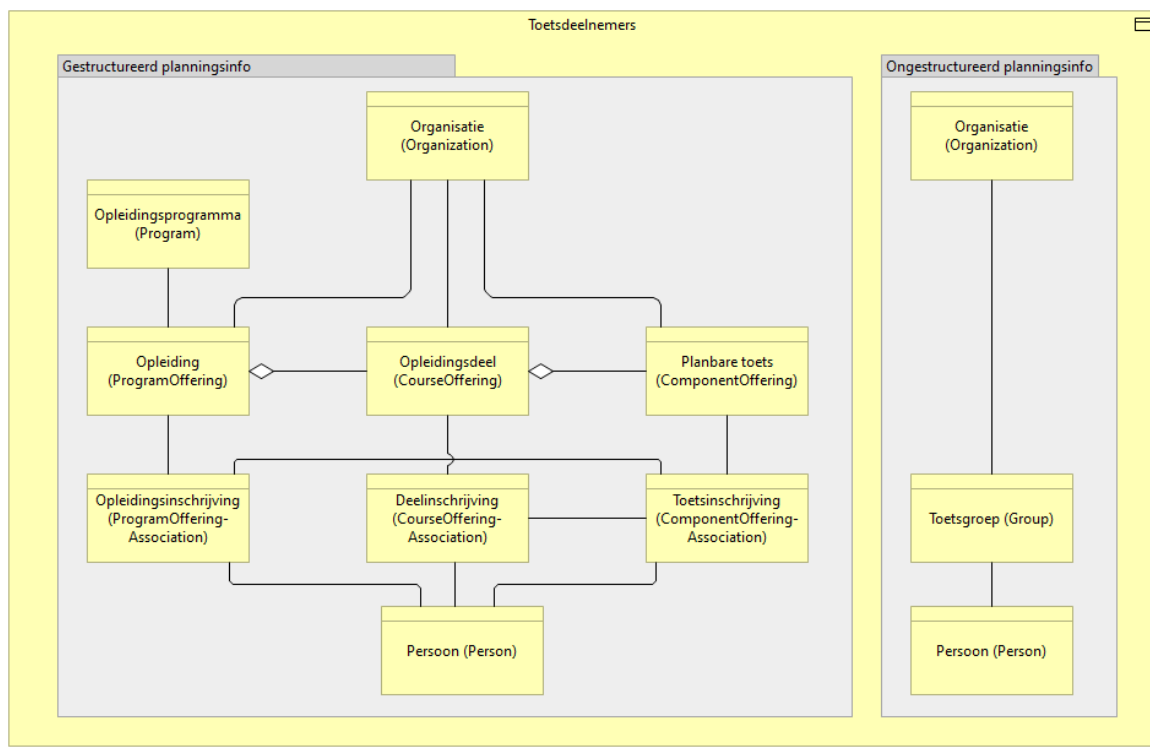
Ten behoeve van de processen “Aanmelden voor toets” en “Maken van toetsplanning” (zie Figuur 2.B in Hoofdstuk 2) heeft **Toetsplanning (& logistiek)**, afgekort TPL, behoefte aan informatie over studenten, opleidingen en de toetsen binnen deze opleidingen. Dit informatieobject Toetsdeelnemers wordt samengesteld en gepubliceerd in **Deelnemerregistratie** of SIS, afgekort DR/SIS, en overgedragen in informatiestroom 1. **Toetsdeelnemers**.

Gegevensverstrekker van de Toetsdeelnemers met informatie over de studenten, opleidingen en toetsen is het systeem/applicatie van de mbo-instelling met de Deelnemerregistratie-functionaliteit. De gegevensafnemers zijn de systemen met de Toetsplanning-functionaliteit. De informatie over de Toetsdeelnemers gaat van DR/SIS naar TPL.

In de overdracht van de Toetsdeelnemers van DR/SIS naar TPL neemt soms TPL het initiatief (zodra er toetsen/examens kunnen worden gepland) en soms DR/SIS (zodra er volgens het onderwijsprogramma toetsen/examens moeten worden gepland). TPL zorgt ervoor dat de informatie correct wordt verwerkt. Voordat de gegevensuitwisseling kan plaatsvinden zijn DR/SIS en TPL al technisch met elkaar verbonden, m.b.t. base-URL, authenticatie etc..

3.1.1 Informatieobject Toetsdeelnemers

In het volgende figuur staan de informatieobjecten binnen het informatieobject Toetsdeelnemers, met tussen haakjes de naam van het betreffende object binnen OOAPI.



Figuur 3.1.18: Informatieobject Toetsdeelnemers in detail

Het **informatieobject Toetsdeelnemers** bevat informatie over te plannen toetsen, inclusief betrokken studenten en medewerkers noodzakelijk om de zitting(en) in te plannen. De planningsinformatie heeft 2 varianten:

- (1) Gestructureerde planningsinformatie en
- (2) Ongestructureerde planningsinformatie.

Bij gestructureerde planningsinformatie (1) wordt bij de betreffende planbare toets (informatieobject Planbare toets) voldoende context van de betreffende student meegegeven om de planning goed te kunnen maken in één of meerdere zittingen. Voorbeelden van dergelijke planbare toetsen zijn:

- Het examen Nederlandse taal 3F moet voor een komend semester voor een grote groep studenten
- Alle drie de toetsen van een specifiek opleidingsdeel voor een bepaald semester
- Een specifieke praktijktest voor vier studenten in een bepaalde opleiding

Tevens kan bij de planbare toets de mogelijk betrokken medewerkers worden gekoppeld.

De studenten en medewerkers in informatieobject Persoon worden middels een object Toetsinschrijving gekoppeld aan de planbare toets (object Planbare toets). De context van de student bestaat uit objecten Opleidingsinschrijving met Opleiding en Deelinschrijving met Opleidingsdeel. De Opleiding is wordt aangevuld met object Opleidingsprogramma.

Voor gestructureerde planbare toetsen komen de resultaten voor deelgenomen studenten terug in de studentadministratie binnen Deelnemerregistratie.

Bij Ongestructureerde planningsinformatie (2) wordt ad hoc voor de betreffende groep studenten (Toetsgroep) direct een zitting gepland. In dit geval is het niet mogelijk resultaten terug te sturen in informatiestroom 5, daartoe ontbreekt ook de betreffende inschrijving van de student. Indien nodig dan zouden deze resultaten handmatig in de administratie moeten worden ingevoerd. Eventueel worden aan deze Toetsgroep ook medewerkers in Persoon gekoppeld.

Let op, een nauwkeurigere benaming voor dit informatieobject Toetsdeelnemers is eigenlijk 'Potentiële toetsdeelnemers' omdat niet iedere student waarvan de informatie wordt uitgewisseld ook daadwerkelijk als deelnemer aan een geplande toets (zitting) zal worden toegevoegd. Dergelijke uitzonderingssituaties zijn bijvoorbeeld studenten die gedurende het planningsproces zijn gestopt met de opleiding of vrijstelling hebben gekregen.

3.1.2 Interacties voor Toetsdeelnemers

Het algemene transactiepatroon is “Pull” zodat de informatie benodigd voor het plannen van de toetsen en bijbehorende studenten en medewerkers die deelnemen aan de toets, door Toetsplanning (TPL) bij Deelnemerregistratie (DR/SIS) wordt opgehaald.

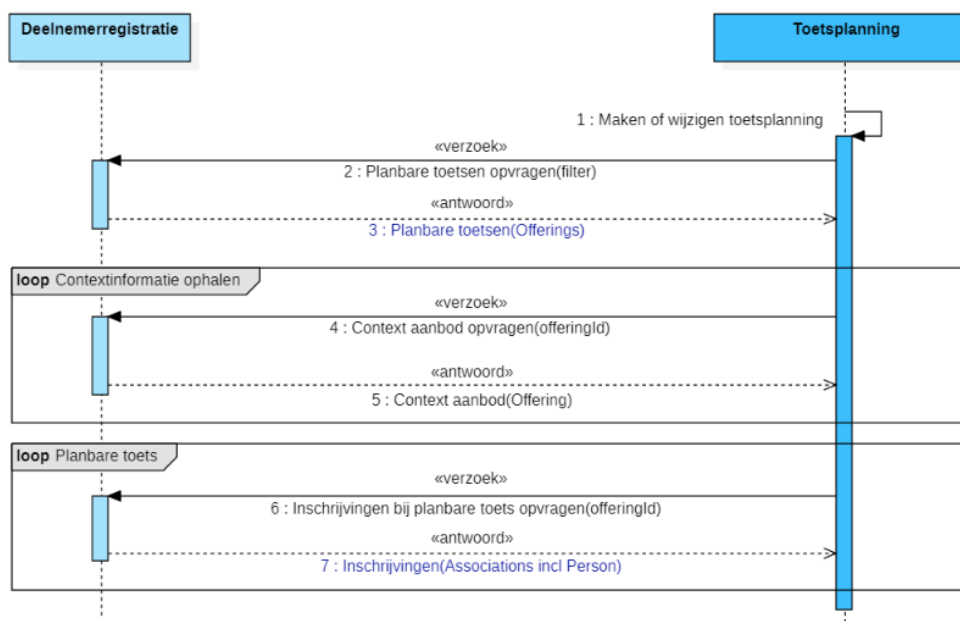
Deze informatiestroom bevat de interacties voor “Gestructureerde planningsinfo” en “Ongestructureerde planningsinfo”.

3.1.2.1 Interacties voor Gestructureerde planningsinfo

Interacties “Gestructureerde planningsinfo ophalen”

In TPL kan het voorkomen dat de informatie over te plannen toetsen moeten worden opgehaald bij DR/SIS. Dit betreft soms globale informatie over planbare toetsen, een afnameperiode en bijbehorende lijst van studenten en medewerkers; soms betreft dit de details van een toetsafname op een vast tijdstip met bijbehorende beperkte lijst van studenten en medewerkers. TPL gebruikt deze planningsinformatie om toetsen op specifieke momenten met bepaalde studenten en medewerkers in te plannen.

De interacties zien er als volgt uit:



Figuur 3.1.2: Gestructureerde planningsinfo ophalen interactiemodel

OOAPI operaties en endpoints

Operatie en endpoint van de eerste interactie in “Gestructureerde planningsinfo ophalen” zijn:

GET /offerings?offeringType=component&component.componentType=test&since={sinceDate}&until={untilDate}

In het verzoek “2: Planbare toetsen opvragen” (zie Figuur 3.1.2) wordt het verzoek van TPL gepreciseerd door de vier filters als query-parameters: offeringType, component.componentType, since en until.

DR/SIS selecteert de Planbare toetsen (objecten ComponentOffering) die voldoen aan het verzoek. Bij de parameters since en until gaat het om alle toetsen die maar enige overlap hebben met periode gespecificeerd door parameters since en until. De periode is gedefinieerd als vanaf begindatum (in parameter since) tot en met einddatum (in parameter until).

Het **antwoordbericht** “3: Planbare toetsen” (zie Figuur 3.1.2) bevat bij succesvolle aanroep de lijst van de **Planbare toetsen** (object ComponentOffering).

Operatie en endpoint van de tweede interactie “Contextinformatie ophalen” zijn:

GET /offerings/{offeringId}

In het verzoek “4: Contextinformatie opvragen” (zie Figuur 3.1.2) worden per aanbod de details opgevraagd; het aanbod wordt geïdentificeerd door offeringId als padparameter. Vanuit de voorgaande interactie “2: Planbare toetsen opvragen” worden gegevens over Planbare toetsen (ComponentOffering) verkregen.

Merk op, dat het aanbod (Offering) niet alleen planbare toetsen betreft maar ook gerelateerde Opleiding (object ProgramOffering) of Opleidingsdeel (object CourseOffering).

Het **antwoordbericht “5: Context aanbod”** (zie Figuur 3.1.2) bevat bij succesvolle aanroep de detailinformatie over de betreffende **Opleiding** (ProgramOffering) of **Opleidingsdeel** (CourseOffering).

Operatie en endpoint van de derde interactie “Inschrijvingen ophalen” zijn:

GET /offerings/{offeringId}/associations

In het verzoek “6: Inschrijvingen bij planbare toets opvragen” (zie Figuur 3.1.2) worden per Planbare toets, geïdentificeerd door offeringId als padparameter, de potentiële deelnemers opgevraagd middels de toetsinschrijvingen (ComponentOfferingAssociation).

Het **antwoordbericht “7: Inschrijvingen”** (zie Figuur 3.1.2) bevat bij succesvolle aanroep een lijst van **Toetsinschrijvingen** (ComponentOfferingAssociation), inclusief verwijzing naar de betreffende persoon.

Interacties “Gestructureerde planningsinfo aanleveren”

In DR/SIS kan het voorkomen dat de planningsinformatie over een gestructureerde planbare toets wordt aangemaakt of gewijzigd. TPL moet op de hoogte blijven van alle wijzigingen, bijvoorbeeld over de planbare toets of over de uitschrijving van studenten. De wijziging in de planningsinfo moeten dan worden aangeleverd bij TPL.

De interacties zien er als volgt uit:



Figuur 3.1.3: Gestructureerde planningsinfo aanleveren interactiemodel

Operatie en endpoint van de interactie “Planbare toets aanmaken” zijn:

PUT /offerings/{offeringId}

In het **verzoek “2: Planbare toets verwerken”** (zie Figuur 3.1.3) worden de gegevens van de **Planbare toets** (ComponentOffering) meegestuurd; de planbare toets wordt geïdentificeerd door offeringId als padparameter. Het aanmaakbericht bevat alle gegevens voor de Planbare toets.

Is offeringId bekend bij TPL dan zullen de gegevens worden overschreven door de gegevens; is offeringId niet bekend dan wordt de Planbare toets aangemaakt binnen TPL.

Het antwoordbericht “3: Verwerkingsresultaat planbare toets” (zie Figuur 3.1.3) bevat bij succesvolle verwerking een positieve terugmelding.

Operatie en endpoint van de interactie “Planbare toets wijzigen” zijn:

PATCH /offerings/{offeringId}

In het **verzoek “2: Planbare toets verwerken”** (zie Figuur 3.1.3) wordt de specifieke **Planbare toets** (ComponentOffering) geïdentificeerd door offeringId als padparameter. In het wijzigingsbericht worden alleen de gewijzigde gegevens van het object ComponentOffering meegestuurd.

Is offeringId bekend bij TPL dan zullen de aangeleverde gegevens worden overschreven; is offeringId niet bekend dan volgt een foutmelding.

Het antwoordbericht 3 (zie Figuur 3.1.3) bevat bij succesvolle verwerking een positieve terugmelding.

Operatie en endpoint van de interactie “Planbare toets verwijderen” zijn:

PATCH /offerings/{offeringId}

In het **verzoek “2: Planbare toets verwerken”** (zie Figuur 3.1.3) wordt de specifieke **Planbare toets** (ComponentOffering) geïdentificeerd door offeringId als padparameter. In het verwijderbericht wordt expliciet de statuswijziging “canceled” in gegeven offeringState van het object ComponentOffering meegestuurd.

Merk op dat als een Planbare toets in TPL wordt verwijderd dat tevens de bijbehorende toetsinschrijvingen moeten worden verwijderd, en ook de gegevens van betreffende personen (student of medewerker) die betrokken zijn bij deze toetsinschrijvingen.

Is offeringId bekend bij TPL dan wordt de Planbare toets verwijderd; is offeringId niet bekend dan volgt een foutmelding.

Het antwoordbericht 3 (zie Figuur 3.1.3) bevat bij succesvolle verwerking een positieve terugmelding.

Operatie en endpoint van de derde interactie **“Toetsinschrijving aanleveren”** zijn:

PUT /associations/{associationId}

In het **verzoek “5: Toetsinschrijving verwerken”** (zie Figuur 3.1.3) worden de inschrijfgegevens van de Planbare toets als **Toetsinschrijving** (ComponentOfferingAssociation) meegestuurd, eventueel inclusief de betrokken persoon (student of medewerker); de specifieke inschrijving wordt geïdentificeerd door associationId als padparameter.

Bij de gegevens van de Toetsinschrijving (ComponentOfferingAssociation) horen contextuele gegevens als de gerelateerde Opleidingsinschrijving en Deelinschrijving. Belangrijk is de informatie over deze gerelateerde inschrijvingen beschikbaar te stellen middels de elders beschreven interactie GET /associations/{associationId}. Het antwoordbericht “6: Verwerkingsresultaat inschrijving” (zie Figuur 3.1.3) bevat bij succesvolle aanroep een positieve terugmelding.

Operatie en endpoint binnen OOAPI van de interactie **“Toetsinschrijving wijzigen”** zijn:

PATCH /associations/{associationId}

In het **verzoek “5: Toetsinschrijving verwerken”** (zie Figuur 3.1.3) wordt de **Toetsinschrijving** (ComponentOfferingAssociation) van de Planbare toets geïdentificeerd door associationId als padparameter. In dit verwijderbericht wordt expliciet de statuswijziging “canceled” meegestuurd in de Toetsinschrijving.

Is associationId bekend bij TPL dan zullen de aangeleverde gegevens worden overschreven; is associationId niet bekend dan volgt een foutmelding.

Het antwoordbericht 6 (zie Figuur 3.1.3) bevat bij succesvolle verwerking een positieve terugmelding.

Operatie en endpoint binnen OOAPI van de interactie **“Toetsinschrijving verwijderen”** zijn:

PATCH /associations/{associationId}

In het **verzoek “5: Toetsinschrijving verwerken”** (zie Figuur 3.1.3) wordt de specifieke inschrijving geïdentificeerd door associationId als padparameter. In dit verwijderbericht wordt expliciet de statuswijziging “canceled” meegestuurd in de **Toetsinschrijving** (object Association).

Merk op dat als een Toetsinschrijving in TPL wordt verwijderd dat tevens de persoonsgegevens van de betreffende persoon (student of medewerker) moeten worden verwijderd.

Is associationId bekend bij TPL dan wordt de Planbare toets verwijderd; is associationId niet bekend dan volgt een foutmelding.

Het antwoordbericht 6 (zie Figuur 3.1.3) bevat bij succesvolle verwerking een positieve terugmelding.

Interacties “Contextinfo wijziging aanleveren”

In DR/SIS kan het voorkomen dat de contextinformatie van een student wordt aangeleverd of gewijzigd. TPL moet op de hoogte blijven van alle wijzigingen, bijvoorbeeld over de opleiding of over de inschrijving of uitschrijving van de opleiding. Ook aanleveringen of wijzigingen in de persoonsdetails kunnen belangrijk zijn voor TPL. De wijziging in de contextinfo moeten dan worden aangeleverd bij TPL.

De interacties zien er als volgt uit:



Figuur 3.1.4: Contextinfo aanleveren interactiemodel

Conform bovenstaande is het mogelijk om de contextinformatie van een student (Opleiding of Opleidingsdeel) bij TPL aan te leveren. Operatie en endpoint van deze **interactie “Opleiding/opleidingsdeel aanmaken”** zijn:

PUT /offerings/{offeringId}

In het **verzoekbericht “2: Opleiding/opleidingsdeel verwerken”** (zie Figuur 3.1.4) worden alle gegevens van de **Opleiding** (ProgramOffering) of het **Opleidingsdeel** (CourseOffering) meegestuurd. De offering wordt geïdentificeerd door offeringId als padparameter.

Is de aangeleverde offeringId al bekend dan worden alle gegevens in het bestaande object overschreven; is de offeringId nog niet bekend dan wordt een nieuw object aangemaakt.

Het antwoordbericht “3: Verwerkingsresultaat opleiding/opleidingsdeel” (zie Figuur 3.1.4) bevat bij succesvolle aanroep een positieve terugmelding.

Verder is het mogelijk om wijziging op een bestaande offering, als contextinformatie van een student, bij TPL aan te leveren. Het gaat hierbij om een Opleidingsdeel of Opleiding. Operatie en endpoint van deze **interactie “Opleiding/Opleidingsdeel wijzigen/verwijderen”** zijn:

PATCH /offerings/{offeringId}

In het **verzoekbericht “2: Opleiding/opleidingsdeel verwerken”** (zie Figuur 3.1.4) worden alleen de gewijzigde gegevens van de offering meegestuurd van **Opleiding** (ProgramOffering) of **Opleidingsdeel** (CourseOffering). De offering wordt geïdentificeerd door offeringId als padparameter.

Is de aangeleverde offeringId bekend dan worden de betreffende gegevens in het bestaande object overschreven; is de offeringId niet bekend dan volgt een foutmelding. Deze interactie kan ook worden gebruikt om de Opleiding of het Opleidingsdeel te verwijderen. En kan ook worden gebruikt om Planbare toets (ComponentOffering) te wijzigen of verwijderen.

Het antwoordbericht 3 (zie Figuur 3.1.4) bevat bij succesvolle aanroep een positieve terugmelding.

Ook is het mogelijk om een nieuwe opleidingsinschrijving of deelinschrijving bij TPL aan te leveren. Operatie en endpoint van deze **interactie “Opleidings-/deelinschrijving aanmaken”** zijn:

PUT /associations/{associationId}

In het **verzoekbericht “5: Opleidings-/deelinschrijving verwerken”** (zie Figuur 3.1.4) worden alle gegevens van de **Opleidingsinschrijving** (ProgramOfferingAssociation) of **Deelinschrijving** (CourseOfferingAssociation) meegestuurd. De inschrijving wordt geïdentificeerd door associationId als padparameter. Is de aangeleverde inschrijving al bekend dan worden alle gegevens in het bestaande object overschreven; is de inschrijving nog niet bekend dan wordt een nieuw aangemaakt.

Het antwoordbericht “6: Verwerkingsresultaat opleidings-/deelinschrijving” (zie Figuur 3.1.4) bevat bij succesvolle aanroep een positieve terugmelding.

Ook is het mogelijk om een wijziging op een opleidingsinschrijving of deelinschrijving bij TPL aan te leveren. Operatie en endpoint van deze **interactie “Opleidings-/Deelinschrijving wijzigen/verwijderen”** zijn:

PATCH /associations/{associationId}

In het **verzoekbericht “5: Opleidings-/deelinschrijving verwerken”** (zie Figuur 3.1.4) worden alleen de gewijzigde gegevens van de **Opleidingsinschrijving** (ProgramOfferingAssociation) of **Deelinschrijving** (CourseOfferingAssociation) meegestuurd. De inschrijving wordt geïdentificeerd door associationId als padparameter.

Is associationId bekend dan worden de betreffende gegevens in het bestaande object overschreven; is de aangeleverde associationId niet bekend dan volgt een foutmelding.

Let op, deze interactie kan hier alleen worden gebruikt om een Opleidingsinschrijving of Deelinschrijving uit de contextinformatie van een student te wijzigen; degelijke verzoeken in relatie tot de medewerker hebben hier geen waarde en zullen worden genegeerd. Deze interactie kan ook worden gebruikt om de inschrijving (Association) van een student te verwijderen. En kan ook worden gebruikt om Toetsinschrijving (ComponentOfferingAssociation) van een student of medewerker te wijzigen.

Het antwoordbericht 6 (zie Figuur 3.1.4) bevat bij succesvolle aanroep een positieve terugmelding.

In aanvulling op bovenstaande is het mogelijk om wijziging van gegevens van een persoon (student of medewerker) aan te leveren. Operatie en endpoint van deze **interactie “Persoonsinformatie aanleveren”** zijn:

PUT /persons/{personId}

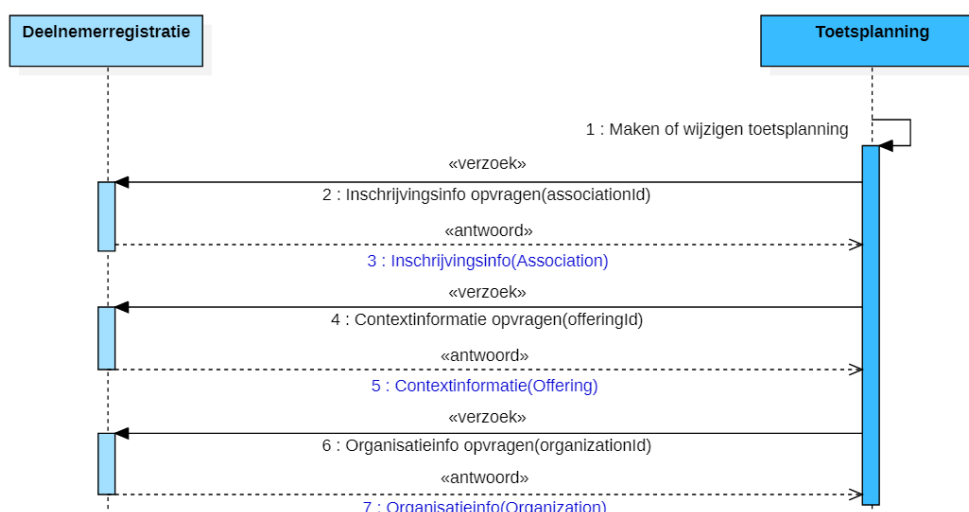
In het **verzoekbericht “8: Persoonsinformatie verwerken”** (zie Figuur 3.1.4) worden de gegevens van de **Persoon** (object Person), d.w.z. student of medewerker, meegestuurd; de persoon wordt geïdentificeerd door personId als padparameter.

Is de aangeleverde personId al bekend dan worden alle gegevens in het bestaande object overschreven; is de personId nog niet bekend dan wordt een nieuw object aangemaakt.

Het antwoordbericht “9: Verwerkingsresultaat persoonsinformatie” (zie Figuur 3.1.4) bevat bij succesvolle aanroep een positieve terugmelding.

Interactie “Inschrijvingsdetails ophalen”

In TPL is het op bepaalde momenten belangrijk om de actuele detailgegevens van een inschrijving te weten. Het gaat hierbij om de inschrijvingsgegevens van een student (of medewerker) zoals Opleidingsinschrijving (ProgramOfferingAssociation), Deelinschrijving (CourseOfferingAssociation) of Toetsinschrijving (ComponentOfferingAssociation). En het gaat hierbij om achter de inschrijving liggende informatie zoals over Opleiding (ProgramOffering), Opleidingsdeel (CourseOffering), Planbare toets (ComponentOffering) of Organisatie (Organization). Deze interacties zien er als volgt uit:



Figuur 3.1.5: Inschrijvingsdetails ophalen interactiemodel

OOAPI operaties en endpoints

Operatie en endpoint van de **interactie “Inschrijvingsinfo ophalen”** zijn:

GET /associations/{associationId}

In het verzoek “2: Inschrijvingsinfo opvragen” (zie Figuur 3.1.5) wordt de specifieke inschrijving geïdentificeerd door associationId als padparameter.

Het **antwoordbericht “3: Inschrijvinginfo”** bevat bij succesvolle aanroep de gegevens over de inschrijving van een student aan een opleiding/opleidingsdeel/planbare toets in **object Association**.

Bij de informatie over de **Toetsinschrijving** (ComponentOfferingAssociation) horen contextuele gegevens als de gerelateerde **Opleidingsinschrijving** (ProgramOfferingAssociation) en **Deelinschrijving** (CourseOfferingAssociation). Deze informatie over een opleidings- of deelinschrijving kan dus eveneens middels deze interactie worden opgevraagd.

Let op, bij het opvragen van de Opleidingsinschrijving is er middels een extra verzoekparameter (expand) de mogelijkheid om in het antwoordbericht tevens de gegevens over de Opleiding (ProgramOffering) en het Opleidingsprogramma (Program) mee te krijgen; en analoog bij de Deelinschrijving de gegevens over het Opleidingsdeel (CourseOffering) mee te krijgen (Deelprogramma in Course is in deze uitwisselingen overbodig), zoals “GET /associations/{associationId}?expand=offering.program”.

Bij de informatie over de opleidingsinschrijving hoort onder andere de gerelateerde Opleiding en bij de Deelinschrijving het gerelateerde Opleidingsdeel. Deze contextinformatie over Opleiding en/of Opleidingsdeel kan eveneens middels de eerder beschreven **interactie “Contextinformatie ophalen”** worden opgevraagd:

GET /offerings/{offeringId}

In het verzoek “5: Contextinformatie opvragen” (zie Figuur 3.1.5) worden van Opleiding (ProgramOffering) en Opleidingsdeel (CourseOffering) de details opgevraagd; het gevraagde object (Offering) wordt geïdentificeerd door offeringId als padparameter.

Het **antwoordbericht “5: Contextinformatie”** bevat bij succesvolle aanroep de detailinformatie over het gevraagde object als context van de inschrijving: **Opleiding** (ProgramOffering) of **Opleidingsdeel** (CourseOffering).

Deze interactie wordt ook gebruikt om de Planbare toets (ComponentOffering) op te halen.

Let op, bij het opvragen van de Opleiding (ProgramOffering) is er middels een extra verzoekparameter (expand) de mogelijkheid om in het antwoordbericht tevens de gegevens over het Opleidingsprogramma (Program) mee te krijgen, zoals door “GET /offerings/{offeringId}?expand=program”. Het Deelprogramma (Course) bij het Opleidingsdeel (CourseOffering) is in deze uitwisselingen overbodig.

Bij deze opgevraagde informatie over het aanbod van Opleiding (ProgramOffering), Deelopleiding (CourseOffering) en Planbare toets (ComponentOffering) kan door deelnemerregistratie informatie over de organisatie worden meegegeven in attribuut organization.

Let op, bij het opvragen van de offering is er middels een extra verzoekparameter (expand) de mogelijkheid om in het antwoordbericht tevens de gegevens over de organisatie (in object Organization) direct mee te krijgen, zoals door “GET /offerings/{offeringId}?expand=organization”.

Als er binnen DR/SIS meerdere niveaus van organisaties zijn, dat wil zeggen binnen het object Organization is verwijzing naar parent- en/of child-organisatie mogelijk dan moet DR/SIS de **interactie “Organisatieinfo ophalen”** ondersteunen:

GET /organizations/{organizationId}

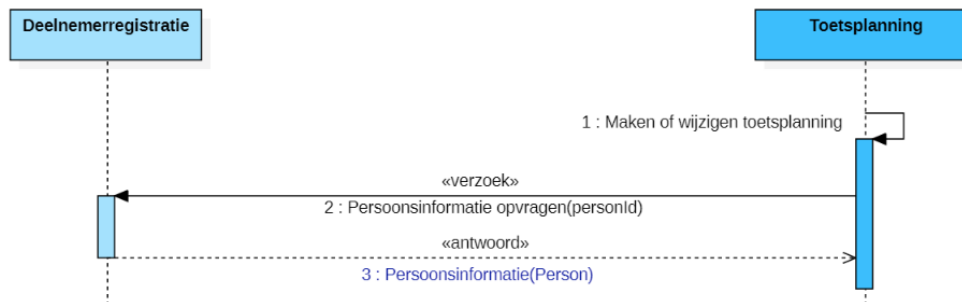
In het verzoek “5: Organisatieinfo opvragen” (zie Figuur 3.1.5) wordt van een specifieke Organisatie de details opgevraagd; het gevraagde object (Organization) wordt geïdentificeerd door organizationId als padparameter.

Het **antwoordbericht “5: Organisatieinfo”** bevat bij succesvolle aanroep de detailinformatie over de **Organisatie** (object Organization) als contextinformatie van de inschrijving.

Dus, als binnen DR/SIS de organisaties plat zijn (zonder hiërarchische onderlinge relaties) dan hoeft DR/SIS deze interactie “Organisatieinfo ophalen” niet te ondersteunen.

Interactie “Persoonsinformatie ophalen”

In TPL is het op bepaalde momenten belangrijk om de actuele detailgegevens van een persoon (student of medewerker) te weten. Deze interactie ziet er als volgt uit:



Figuur 3.1.6: Persoonsinformatie ophalen interactiemodel

OOAPI operaties en endpoints

Operatie en endpoint van de extra optionele interactie “**Persoonsinformatie ophalen**” zijn:

GET /persons/{personId}

In het verzoek “2: Persoonsinformatie opvragen” (zie Figuur 3.1.6) worden de gegevens van een specifieke persoon (student of medewerker) opgevraagd in **Persoon** (Person). De persoon is geïdentificeerd door personId als padparameter.

Het **antwoordbericht “3: Persoonsinformatie”** bevat bij succesvolle aanroep de gegevens over de Persoon (in object Person).

3.1.2.2 Interacties voor Ongestructureerde planningsinfo

Interacties “Ongestructureerde planningsinfo ophalen”

In TPL kan het voorkomen dat direct (ad hoc) voor een groep studenten een zitting moet worden gepland. TPL vraagt de groep studenten (Toetsgroep) bij DR/SIS op nadat deze is klaargezet. In dit geval van “Ongestructureerde planningsinfo ophalen” is het niet mogelijk resultaten van de studenten terug te sturen naar DR/SIS via informatiestroom 5 Studentresultaat.

De interacties zien er als volgt uit:



Figuur 3.1.7: Ongestructureerde planningsinfo ophalen interactiemodel

OOAPI operaties en endpoints

Operatie en endpoint van de eerste interactie “**Adhoc toetsgroepen ophalen**” zijn:

GET /groups?q={zoektekst}&since={sinceDate}&until={untilDate}

In het verzoek “2: Toetsgroepen opvragen” (zie Figuur 3.1.7) wordt het verzoek om Toetsgroepen (Group) gepreciseerd door een zoektekst bij query-parameter q mee te geven. Alleen die groepen met de opgegeven zoektekst in de naam worden teruggegeven. De zoekvraag kan ook worden uitgebreid met de parameters since en until om alle groepen op te vragen die maar enige overlap hebben met de periode gespecificeerd door deze parameters. De periode is gedefinieerd als vanaf begindatum (in parameter since) tot en met einddatum (in parameter until).

Het **antwoordbericht “3: Toetsgroepen”** (zie Figuur 3.1.7) bevat bij succesvolle aanroep de lijst van de **Toetsgroepen** (objecten Group) die voldoen.

Operatie en endpoint van de tweede interactie “Groepslidmaatschappen ophalen” zijn:

GET /groups/{groupid}/members

In het verzoek “4: Groepslidmaatschappen opvragen” (zie Figuur 3.1.7) worden alle groepslidmaatschappen van een specifieke toetsgroep opgevraagd; de Toetsgroep (Group) wordt geïdentificeerd door groupId als padparameter.

Het **antwoordbericht “5: Groepslidmaatschappen”** (zie Figuur 3.1.7) bevat bij succesvolle aanroep de lijst van **Groepslidmaatschappen** (object MemberschipItem) van de **Toetsgroep** (Group). Ieder Groepslidmaatschap bevat een verwijzing naar de betreffende persoon (student of medewerker).

Operatie en endpoint “GET /groups/{groupid}/persons” was voorheen in gebruik om de aan een Toetsgroep (Group) gekoppelde personen op te vragen. Deze operaties is onderdeel van OOAPI en kan eventueel door DR/SIS worden ondersteund, maar dit is niet verplicht binnen deze uitwisselingen.

In navolging van het voorgaande is het belangrijk om de informatie over een persoon te weten (student of medewerker). Deze interactie kan ook worden gebruikt om de meest actuele details over een bepaald persoon (student of medewerker) op te vragen. Operatie en endpoint van deze **interactie “Persoonsinformatie ophalen”** zijn:

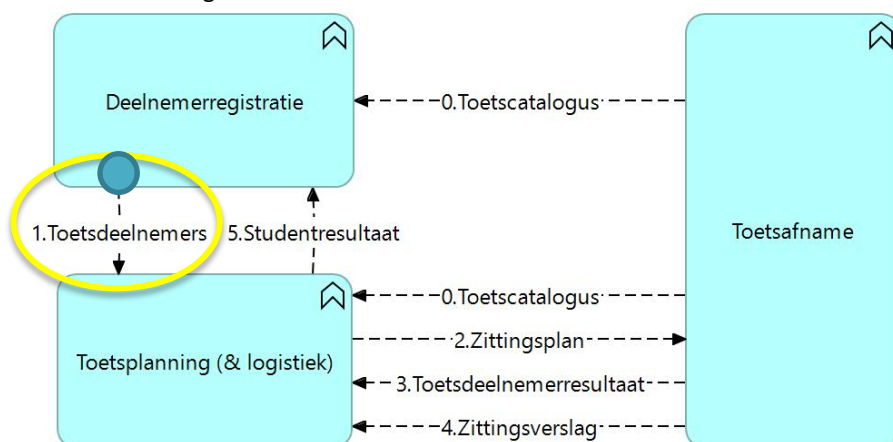
GET /persons/{personId}

In het verzoek “2: Persoonsinformatie opvragen” (zie Figuur 3.1.6) worden de gegevens van een specifieke persoon (student of medewerker) opgevraagd in Persoon (Person). De persoon is geïdentificeerd door personId als padparameter.

Het **antwoordbericht “3: Persoonsinformatie”** (zie Figuur 3.1.6) bevat bij succesvolle aanroep de gegevens over de **Persoon** (in object Person).

3.1.3 Koppelvlak bij Deelnemerregistratie

Het koppelvlak bij Deelnemerregistratie (DR/SIS) voor informatiestroom Toetsdeelnemers is in onderstaand figuur met blauwe cirkel gemarkeerd.



Figuur 3.1.8: Koppelvlak bij Deelnemerregistratie voor Toetsdeelnemers

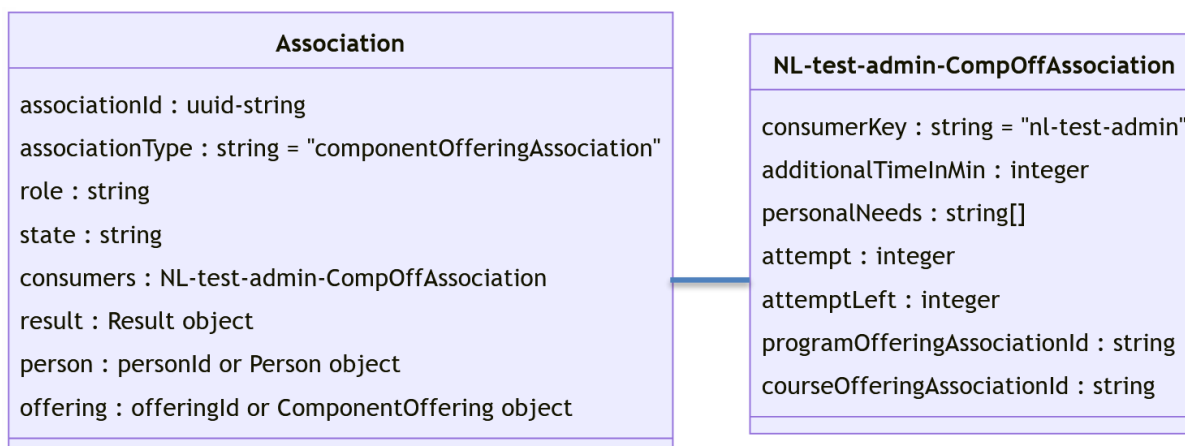
Operatie & Endpoint	Verzoekbericht	Antwoordbericht
GET /associations/{associationId}	Padparameter: associationId=uuid Query parameter: expand={attrnaam}, zoals offering, offering.program, person	Inhoud Inschrijving voor Opleiding (ProgramOfferingAssociation), Opleidingsdeel (CourseOfferingAssociation) of Planbare toets (ComponentOfferingAssociation), zie Figuur 3.1.9 en 3.1.10
GET /groups	Query parameter: q={zoektekst}	Inhoud Toetsgroep (Group), zie Figuur 3.1.11
GET /groups/{groupId}/members	Padparameter: groupId=uuid	Inhoud Groepslidmaatschap (MembershipItem), zie Figuur 3.1.11
GET /groups/{groupId}/persons	Padparameter: groupId=uuid	Inhoud Groepslid (Person), zie Figuur 3.1.12
GET /offerings	Query parameters: offeringType=component since={sinceDate} until={untilDate}	Inhoud Planbare toets (ComponentOffering), zie Figuur 3.1.9
GET /offerings/{offeringId}	Padparameter: offeringId=uuid Query parameter: expand={attrnaam}, zoals program, course, component, organization	Inhoud Opleiding (ProgramOffering), Opleidingsdeel (CourseOffering) of Planbare toets (ComponentOffering), zie Figuur 3.1.13 en 3.1.14
GET /offerings/{offeringId}/associations	Padparameter: offeringId=uuid	Inhoud Lijst van Toetsinschrijvingen (ComponentOfferingAssociation), zie Figuur 3.1.9
GET /organizations/{organizationId}	Padparameter: organizationId=uuid Query parameter: expand={attrnaam}, zoals parent, children	Inhoud Organisatie (object Organization), zie Figuur 3.1.15
GET /persons/{personId}	Padparameter: personId=uuid	Inhoud Persoon (object Person), zie Figuur 3.1.12

Tabel 3.1.1: Endpoints en operaties in koppelvlak bij Deelnemerregistratie voor Toetsdeelnemers

De aanvrager (verzoeker/initiator) van alle endpoint-operaties in bovenstaande tabel is TPL.

Het antwoordbericht van **GET /associations/{associationId}** bevat hier een object Association van het type **ComponentOfferingAssociation** (Toetsinschrijving), **ProgramOfferingAssociation** (Opleidingsinschrijving) en **CourseOfferingAssociation** (Deelinschrijving).

In deze situatie (Informatiestroom 1) wordt gegeven **result binnen object Association** niet gebruikt. Dit gegeven result wordt in informatiestroom 5 bij een ComponentOfferingAssociation gebruikt om het toetsresultaat van een student over te dragen. Wanneer deze inschrijving (toets-, opleidingsdeel- of opleidingsinschrijving) wel al een resultaat heeft in DR/SIS dan zal dit resultaat worden meegestuurd (zie figuren 3.1.9 en 3.1.10).

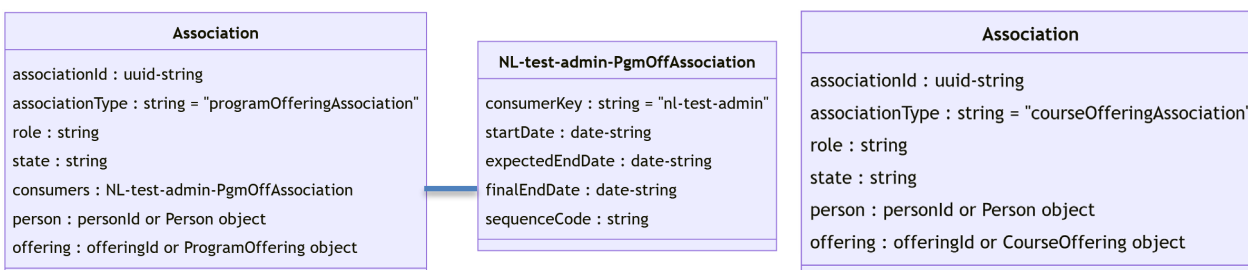


Figuur 3.1.9: ComponentOfferingAssociation (Toetsinschrijving) gegevensmodel in 1. Toetsdeelnemers

Let op, in het gegevensblok consumers van ComponentOfferingAssociation (figuur 3.1.9) wordt de verwijzing naar contextinformatie als Opleidingsinschrijving (in programOfferingAssociationId) en Deelinschrijving (in courseOfferingAssociationId) meegestuurd. De informatie over een opleidings- of deelinschrijving kan dus eveneens middels deze GET /associations/{associationId} worden opgevraagd.

Bij het opvragen van een inschrijving van een Planbare toets, Opleiding of Opleidingsdeel is er middels een extra verzoekparameter (expand) ook de mogelijkheid om in het antwoordbericht tevens de gegevens over de Persoon (Student of Medewerker) in object Person onder gegeven person mee te krijgen, zoals "GET /associations/{associationId}?expand=person". Deze informatie over de persoon kan ook direct door GET /persons/{personId} worden opgevraagd.

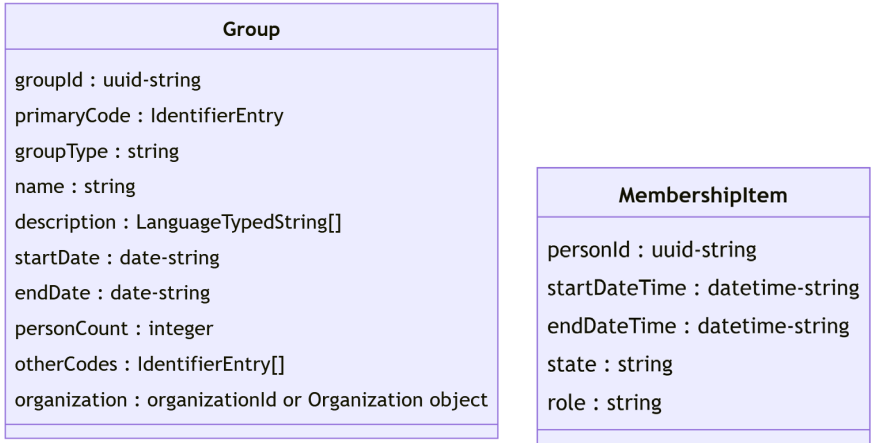
Ook is er de mogelijkheid om in het antwoordbericht de gegevens over de offering (Planbare toets, Opleiding of Opleidingsdeel) in het object ComponentOffering, ProgramOffering respectievelijk CourseOffering in gegeven offering mee te krijgen. Hiertoe moet de query-parameter expand worden gebruikt, zoals "GET /associations/{associationId}?expand=offering". Of zelfs, het Opleidingsprogramma (Program) binnen object ProgramOffering door de query: GET /associations/{associationId}?expand=offering.program".



Figuur 3.1.10: ProgramOfferingAssociation (links) en CourseOfferingAssociation (rechts) in 1. Toetsdeelnemers

Let op, in ProgramOfferingAssociation en CourseOfferingAssociation (figuur 3.1.10) wordt de informatie in gegevens person en offering bij default middels verwijzing meegestuurd, d.w.z. personId of offeringId. De informatie over de persoon zou inmiddels bekend moeten zijn vanuit de Toetsinschrijving (ComponentOfferingAssociation) of de Toetsgroep (Group) van de student en kan middels de expand query parameter of GET /persons/{personId} worden opgevraagd. De informatie over de offering (ProgramOffering of CourseOffering) kan middels de expand query parameter of GET /offerings/{offeringId} worden opgevraagd.

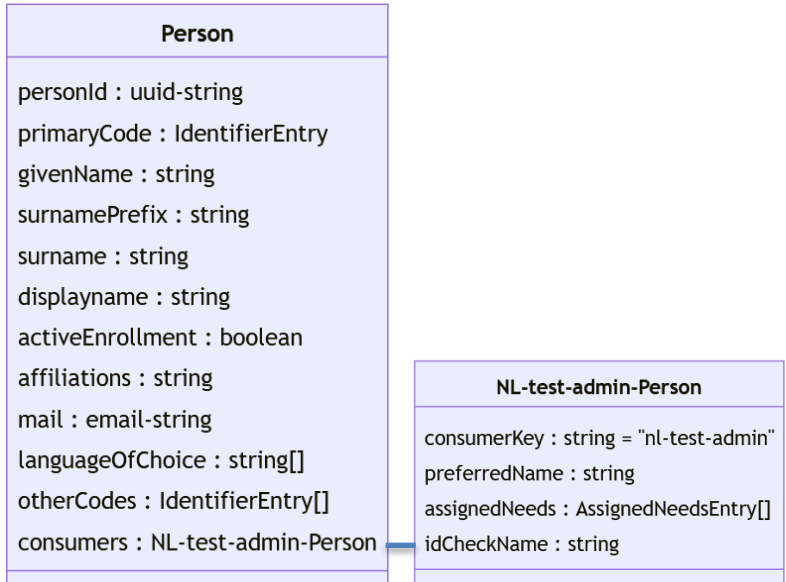
Het antwoordbericht van [GET /groups](#) bevat hier objecten **Group** (zie Figuur 3.1.11).



Figuur 3.1.11: Group (toetsgroep) en MembershipItem (Groepslidmaatschap) gegevensmodel in 1. Toetsdeelnemers

Het antwoordbericht van `GET /groups/{groupid}/members` bevat alle groepslidmaatschappen van de Toetsgroep in gegevensobjecten van type **MembershipItem** zoals weergegeven in bovenstaand Figuur 3.1.11.

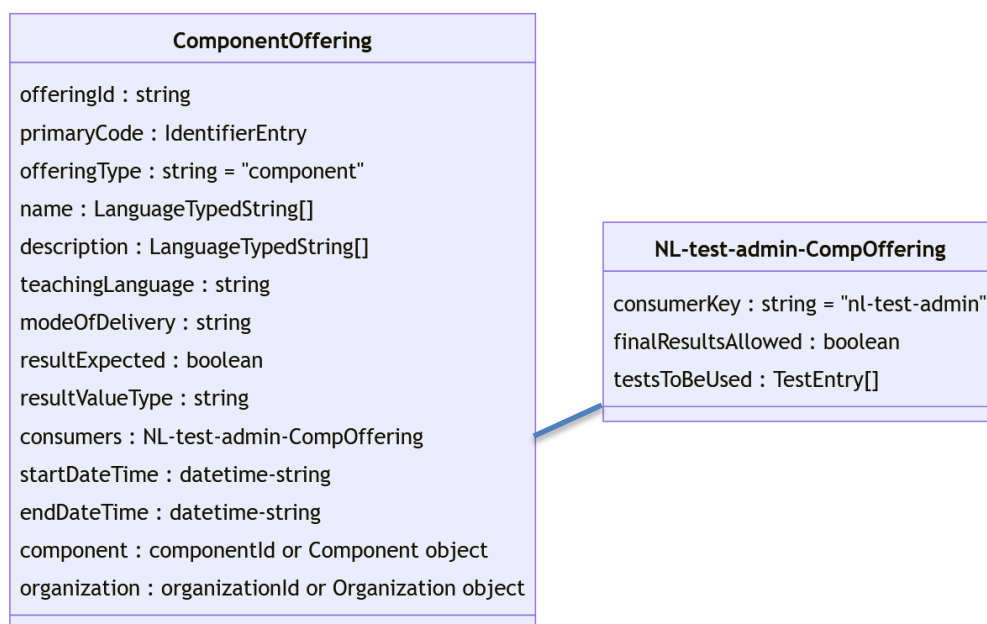
Het antwoordbericht van `GET /groups/{groupid}/persons` bevat alle personen die betrokken zijn bij de groep in gegevensobjecten van type **Persoon** zoals weergegeven in bovenstaand Figuur 3.1.11.



Figuur 3.1.12: Person (Persoon) gegevensmodel in 1. Toetsdeelnemers

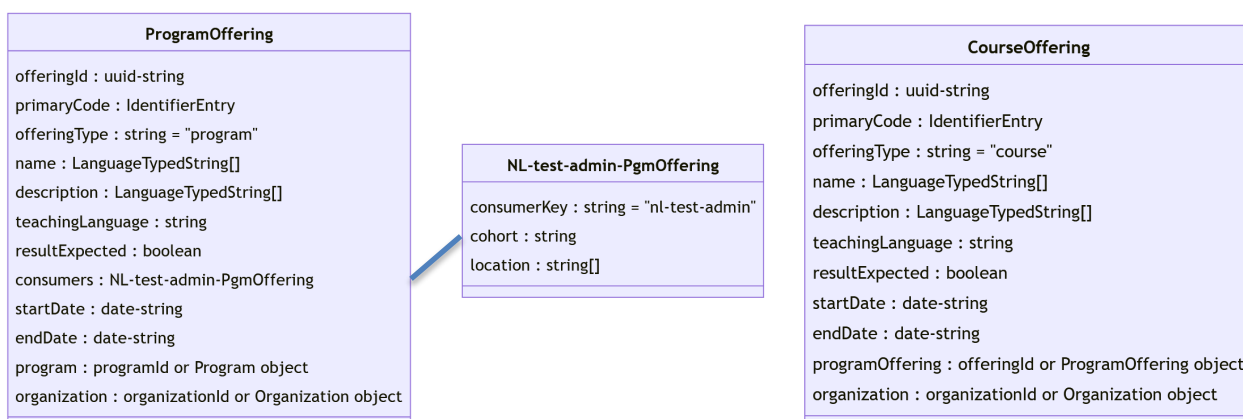
Het antwoordbericht van `GET /offerings` bevat object(en) Offering van het type **ComponentOffering** (Planbare toets).

In deze situatie (Informatiestroom 1) worden gegevens **irregularities en documents** in het gegevensblok consumers van object ComponentOffering niet gebruikt. Deze gegevens worden in informatiestroom 4 gebruikt om het resultaat van de zitting over te dragen (zie Figuur 3.1.13).



Figuur 3.1.13: ComponentOffering (Planbare toets) gegevensmodel in 1. Toetsdeelnemers

Het antwoordbericht van **GET /offerings/{offeringId}** bevat hier een object Offering van het type **ComponentOffering** (Planbare toets), zie figuur 3.1.13, of van het type **ProgramOffering** (Opleiding) of **CourseOffering** (Opleidingsdeel), zie Figuur 3.1.14.



Figuur 3.1.14: ProgramOffering (Opleiding) en CourseOffering (Opleidingsdeel) gegevensmodel in 1. Toetsdeelnemers

Let op, bij het opvragen van de opleiding (ProgramOffering) is er de mogelijkheid om in het antwoordbericht tevens de gegevens over het opleidingsprogramma (Program) mee te krijgen, zie Figuur 3.1.15. Hiertoe moet de query-parameter expand worden gebruikt, zoals "GET /offerings/{offeringId}?expand=program". Het Deelprogramma (Course) bij het Opleidingsdeel (CourseOffering) is in deze uitwisselingen overbodig.

Program	Organization
programId : uuid-string primaryCode : IdentifierEntry programType : string name : LanguageTypedString[] abbreviation : string description : LanguageTypedString[] teachingLanguage : string modeOfStudy : string duration: string firstStartDate : date-string levelOfQualification : string otherCodes : IdentifierEntry[] validFrom : date-string validTo : date-string	organizationId : uuid-string primaryCode : IdentifierEntry organizationType : string name : string shortname : string description : LanguageTypedString[] addresses : Address object[] link : uri-string logo : uri-string otherCodes : IdentifierEntry[] parent : organizationId or Organization object children : organizationId[] or Organization object[]

Figuur 3.1.15: Program (Opleidingsprogramma) en Organisatie (Organization) gegevensmodel in 1. Toetsdeelnemers

Het antwoordbericht van [GET /offerings/{offeringId}/associations](#) bevat hier objecten Association van het type **ComponentOfferingAssociation** (Toetsinschrijving).

In deze situatie (Informatiestroom 1) wordt gegeven **result** binnen **ComponentOfferingAssociation** niet gebruikt. Dit gegeven result wordt in informatiestroom 5 gebruikt om het toetsresultaat over te dragen. Wanneer deze toetsinschrijving inmiddels al een resultaat heeft in DR/SIS dan zal dit toetsresultaat in result worden meegestuurd (zie figuur 3.1.9).

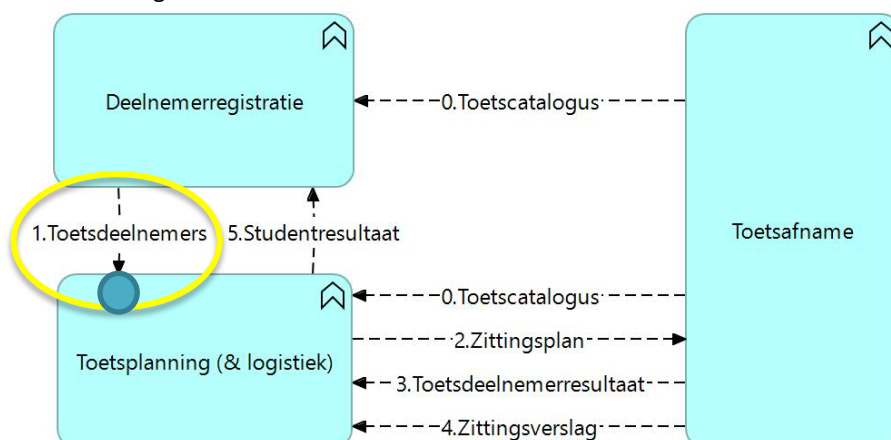
Het antwoordbericht van [GET /persons/{personId}](#) bevat alle gegevens van de betreffende persoon (Student of Medewerker) in gegevensobject **Person** zoals weergegeven in bovenstaand Figuur 3.1.12.

Het antwoordbericht van [GET /organizations/{organizationId}](#) bevat alle gegevens van de betreffende organisatie in gegevensobject **Organization** zoals weergegeven in bovenstaand Figuur 3.1.15.

Let op, als binnen DR/SIS de organisaties plat zijn (zonder hiërarchische onderlinge relaties) dan hoeft DR/SIS deze "GET /organizations/{organizationId}" niet te ondersteunen.

3.1.4 Koppelvlak bij Toetsplanning

Het koppelvlak bij Toetsplanning (TPL) voor informatiestroom Toetsdeelnemers is in onderstaand figuur met blauwe cirkel gemarkeerd.



Figuur 3.1.16: Koppelvlak bij Toetsplanning voor Toetsdeelnemers

Operatie & Endpoint	Verzoekbericht	Antwoordbericht
PATCH <code>/associations/{associationId}</code>	<u>Padparameter:</u> associationId=uuid <u>Inhoud</u> Opleidingsinschrijving (ProgramOfferingAssociation), Deelinschrijving (CourseOfferingAssociation) of Toetsinschrijving (ComponentOfferingAssociation), zie Figuur 3.1.9 en 3.1.10 en 3.1.17)	Statuscode
PUT <code>/associations/{associationId}</code>	<u>Padparameter:</u> associationId=uuid <u>Inhoud</u> Opleidingsinschrijving (ProgramOfferingAssociation), Deelinschrijving (CourseOfferingAssociation) of Toetsinschrijving (ComponentOfferingAssociation), zie Figuur 3.1.9 en 3.1.10	Statuscode
PATCH <code>/offerings/{offeringId}</code>	<u>Padparameter:</u> offeringId=uuid <u>Inhoud</u> Opleiding (ProgramOffering), Opleidingsdeel (CourseOffering) of Planbare toets (ComponentOffering), zie Figuur 3.1.13 en 3.1.14 en 3.1.18	Statuscode
PUT <code>/offerings/{offeringId}</code>	<u>Padparameter:</u> offeringId=uuid <u>Inhoud</u> Opleiding (ProgramOffering), Opleidingsdeel (CourseOffering) of Planbare toets (ComponentOffering), zie Figuur 3.1.13 en 3.1.14	Statuscode
PUT <code>/persons/{personId}</code>	<u>Padparameter:</u> personId=uuid <u>Inhoud</u> Persoon (Person), zie Figuur 3.1.12	Statuscode

Tabel 3.1.1: Endpoints en operaties in koppelvlak bij Toetsplanning in 1. Toetsdeelnemers

De aanvrager (verzoeker/initiator) van alle endpoint-operaties in bovenstaande tabel is DR/SIS.

Het verzoekbericht van `PATCH /associations/{associationId}` bevat hier gegevens volgens het object **Association** van het type **ProgramOfferingAssociation** (Opleidingsinschrijving), **CourseOfferingAssociation** (Deelinschrijving) of **ComponentOfferingAssociation** (Toetsinschrijving), zie Figuur 3.1.9 en 3.1.10).

Bij wijziging van de inschrijving bevat het verzoekbericht alleen de te wijzigen gegevens, daarbij is het gegeven **associationType** binnen object **Association** bij deze PATCH operatie altijd verplicht.

Alleen bij verwijdering van de inschrijving bevat het verzoekbericht de statuswijziging "canceled". In het verzoekbericht zijn in deze situatie (Verwijdering van toetsinschrijving) gegevens **state** (met waarde "canceled") en **associationType** binnen object **Association** verplicht; alle overige gegevens binnen Association zijn overbodig want deze zullen door TPL worden genegeerd (zie onderstaand Figuur 3.1.17).

Association
associationType : string
state : string = "canceled"

Figuur 3.1.17: Association (Toetsinschrijving) gegevensmodel t.b.v. Inschrijving verwijderen in 1. Toetsdeelnemers

Het verzoekbericht van `PUT /associations/{associationId}` bevat hier het **object Association** van het type **ProgramOfferingAssociation** (Opleidingsinschrijving) of **CourseOfferingAssociation** (Deelinschrijving) of **ComponentOfferingAssociation** (Toetsinschrijving), inclusief `associationId`, zie bovenstaand Figuur 3.1.9 en Figuur 3.1.10.

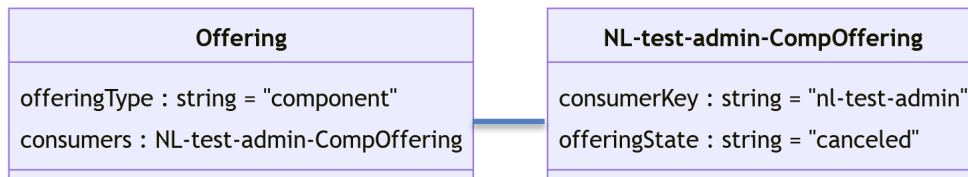
Let op, in deze situatie (Informatiestroom 1) wordt gegeven **result** binnen **Association** niet gebruikt; dit gegeven wordt in informatiestroom 5 gebruikt om het resultaat van een student over te dragen.

Let op, in al deze **Associations** wordt de verwijzing naar de student door de `personId` in gegeven **person** meegestuurd. De informatie over de persoon kan middels `GET /persons/{personId}` bij DR/SIS worden opgevraagd. En wordt de verwijzing naar de offering door de `offeringId` in gegeven **offering** meegestuurd. De informatie over de offering als **ProgramOffering** (Opleiding), **CourseOffering** (Opleidingsdeel) en **ComponentOffering** (Planbare toets) wordt ook door verwijzing (`offeringId` in `offering`) meegestuurd en kan middels `GET /offerings/{offeringId}` bij DR/SIS worden opgevraagd.

Het verzoekbericht van `PATCH /offerings/{offeringId}` bevat hier gegevens volgens het object **Offering** van het type **ProgramOffering** (Opleiding), **CourseOffering** (Opleidingsdeel) of **ComponentOffering** (Planbare toets), zie Figuur 3.1.13 en 3.1.14).

Bij wijziging van de offering bevat het verzoekbericht alleen de te wijzigen gegevens, daarbij is het gegeven **offeringType** binnen **object Offering** bij deze PATCH operatie altijd verplicht.

Alleen bij verwijdering van de offering bevat het verzoekbericht met de statuswijziging "canceled". In het verzoekbericht zijn in deze situatie (Verwijdering van offering) gegevens **offeringState** (met waarde "canceled" in **consumers**) en **offeringType** binnen **object Offering** verplicht; alle overige gegevens binnen **Offering** zijn overbodig want deze zullen door TPL worden genegeerd (zie onderstaand Figuur 3.1.18).



Figuur 3.1.18: ComponentOffering (Planbare toets) gegevensmodel t.b.v. Planbare toets verwijderen in 1. Toetsdeelnemers

Het verzoekbericht van `PUT /offerings/{offeringId}` bevat hier het object **Offering** van het type **ProgramOffering** (Opleiding), **CourseOffering** (Opleidingsdeel) of **ComponentOffering** (Planbare toets), inclusief `offeringId`, zie bovenstaand Figuur 3.1.13 en Figuur 3.1.14.

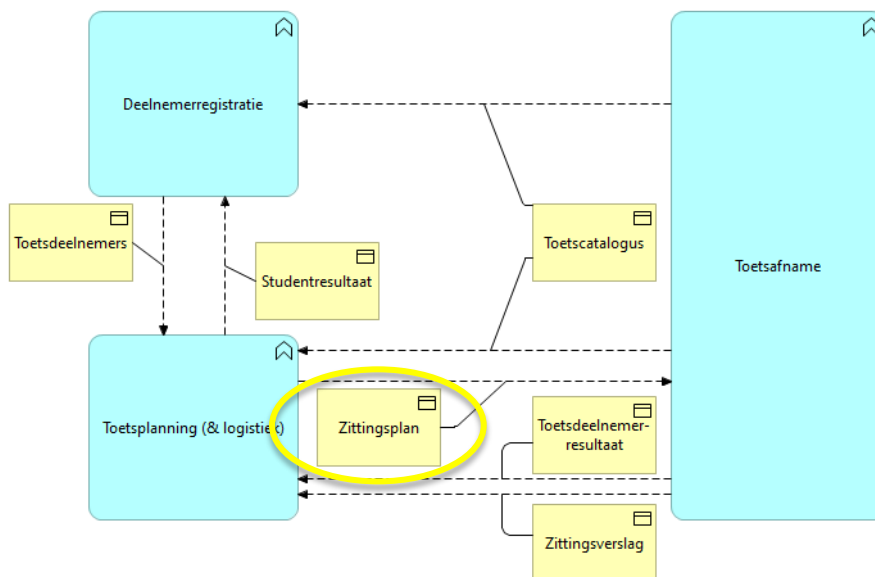
In deze situatie (Informatiestroom 1) worden gegevens **irregularities en documents** in het gegevensblok **consumers** van **object ComponentOffering** niet gebruikt; deze gegevens worden in informatiestroom 4 gebruikt om het resultaat van de zitting (zittingsverslag) over te dragen.

Het verzoekbericht van `PUT /persons/{personId}` bevat hier het **object Person** (Persoon).

In deze situatie (Informatiestroom 1) worden alle benodigde gegevens van **object Person** gebruikt, inclusief `personId` (zie bovenstaand Figuur 3.1.12).

3.2 Zittingsplan informatiestroom

De informatiestroom Zittingsplan (flow 2) betreft het informatieobject Zittingsplan dat in onderstaand figuur met gele ovaal is gemarkeerd.



Figuur 3.2.1A: Informatieobject Zittingsplan in het scenario

Ten behoeve van het proces “Afnemen van toets” (zie Figuur 2.B in Hoofdstuk 2) heeft **Toetsafname** behoefte aan informatie over de afnameplanning van toets/examen. Dit informatieobject Zittingsplan wordt beschikbaar gesteld in **Toetsplanning (& logistiek)** en overgedragen in informatiestroom 2. **Zittingsplan**.

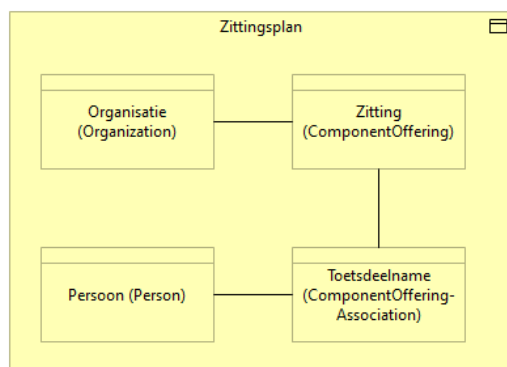
Gegevensverstrekker van het **Zittingsplan** is het systeem (de applicatie) van de mbo-onderwijsinstelling met de Toetsplanning-functionaliteit. De gegevensafnemer is het systeem met de Toetsafname-functionaliteit. De informatie over het Zittingsplan gaat van Toetsplanning (TPL) naar Toetsafname (TAS).

Om vanuit TPL de student de toegangslink (startup-url) naar de toetsapplicatie beschikbaar te kunnen stellen kan TPL deze toegangslink ophalen bij TAS. Deze toegangslink is specifiek voor de deelname aan de betreffende zitting.

In de overdracht van de Toetsdeelnemers van TPL naar TAS neemt TPL het initiatief zodra het **Zittingsplan** is samengesteld en gereed voor verzending. TAS zorgt ervoor dat het zittingsplan correct wordt verwerkt.

Voordat de gegevensuitwisseling kan plaatsvinden zijn TPL en TAS al technisch met elkaar verbonden, m.b.t. base-URL, authenticatie etc..

3.2.1 Informatieobject Zittingsplan



Figuur 3.2.1B: Informatieobject Zittingsplan in detail

Het informatieobject **Zittingsplan** omvat de informatie over de geplande afname van de toets (of het examen) in informatieobject Zitting, inclusief de betrokken studenten en medewerkers (afnameleider en surveillanten) in informatieobjecten Toetsdeelname en Persoon. De Zitting bevat eventueel informatie over de betreffende organisatie in Organisatie en tevens een verwijzing naar de betreffende toets uit de toetscatalogus. Het Zittingsplan heeft alle informatie die nodig is om de toetsafname uit te voeren.

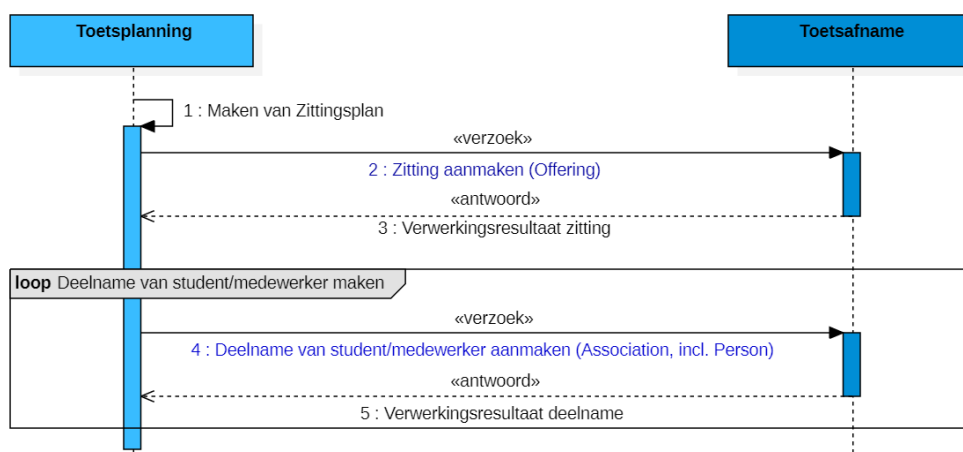
De unieke identifier van het Zittingsplan wordt later gebruikt bij vervolgleveringen van het gewijzigde Zittingsplan en de aanlevering van het Zittingsverslag (informatiestroom 4). De studenten en medewerkers worden gekoppeld aan de zitting door een deelname met een eigen unieke identifier. Deze identifier van de deelname wordt later gebruikt bij vervolgaanleveringen met gewijzigde deelname en tevens voor de aanlevering van het Toetsdeelnemersresultaat (informatiestroom 3) om de betreffende deelname te identificeren.

3.2.2 Interacties voor Zittingsplan

Het algemene transactiepatroon is “Push” zodat de gegevens bij Toetsafname (TAS) worden gebracht. Deze informatiestroom bevat de interacties “Zittingsplan maken”, “Zittingsplan wijzigen”, “Deelname wijzigen”, “Zittingsdetails ophalen” en “Toegangslink ophalen”.

Interacties “Zittingsplan maken”

In TPL moet het gemaakte **Zittingsplan** kunnen worden overgedragen aan TAS. Het brengen van het Zittingsplan bij TAS gaat in stappen: eerst de details over de zitting en vervolgens de deelnames van studenten en medewerkers (afnameleider, surveillanten en beoordelaars) aan de zitting. De interacties zien er als volgt uit:



Figuur 3.2.2: Zittingsplan maken interactiemodel

OOAPI operaties en endpoints

Operatie en endpoint binnen OOAPI van de eerste interactie “Zitting aanmaken” zijn:

PUT /offerings/{offeringId}

In **verzoek “2: Zitting aanmaken”** (zie Figuur 3.2.2) wordt de betreffende zitting geïdentificeerd door offeringId als padparameter (zie {offeringId}). Het verzoekbericht zelf bevat de gegevens van de zitting in een **object ComponentOffering**. Let op, dit endpoint en operatie ligt in het koppelvlak bij TAS.

Het antwoordbericht “3: Verwerkingsresultaat zitting” (zie Figuur 3.2.2) bevat bij succesvolle verwerking een positieve terugmelding.

Operatie en endpoint binnen OOAPI van de tweede interactie “Deelname van student/medewerker aanmaken” zijn:

PUT /associations/{associationId}

In **verzoek “4: Deelname van student/medewerker aanmaken”** (zie Figuur 3.2.2) wordt de deelname door associationId als padparameter (zie {associationId}). De betreffende zitting waar de deelname aan moeten worden toegevoegd wordt geïdentificeerd door gegeven offeringId in het object **Association**.

Het verzoekbericht zelf bevat de gegevens van de deelname in object **Association** van het type **ComponentOfferingAssociation** en bijbehorende persoonsgegevens in bijgaande object **Person**.

Deze interactie zal gewoonlijk in de reeks van alle deelnames bij de zitting worden uitgevoerd, voorafgegaan door de eerste interactie “Zitting aanmaken”. Maar deze interactie kan ook zelfstandig worden uitgevoerd, bijvoorbeeld wanneer één student zich afmeldt voor de toets.

Deze interactie moet voor iedere deelname van student of medewerker worden herhaald.

Let op, bij het maken/toevoegen van een deelname in deze informatiestroom ontbreekt altijd het deelgegeven result (in object Result)!

Het antwoordbericht “5: Verwerkingsresultaat deelname” (zie Figuur 3.2.2) bevat bij succesvolle verwerking een positieve terugmelding.

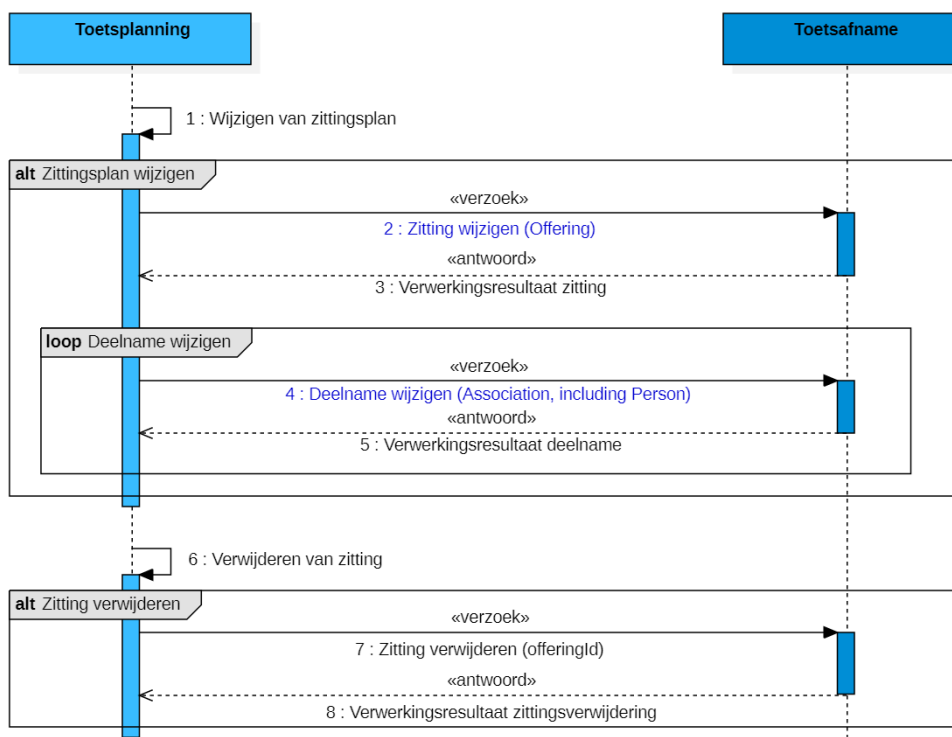
Interacties “Zittingsplan wijzigen”

In TPL moet het eerder aan TAS overgedragen **Zittingsplan** kunnen worden gewijzigd. Het brengen van het gewijzigde Zittingsplan valt uiteen in twee onderdelen:

- De details over de zitting wijzigen
- De deelnames van student of medewerker aan de zitting wijzigen

Ook kan het voorkomen dat de gegevens van de zitting moeten worden verwijderd.

De interacties zien er als volgt uit:



Figuur 3.2.3: Zittingsplan wijzigen interactiemodel

Deze overdracht van de gewijzigd Zitting en de Deelnames is een **standlevering**, d.w.z. iedere keer worden alle gegevens volgens de actuele stand van zaken overgedragen. Deze interactie is daarmee **herhaalbaar** op elk gewenst moment.

Om te bepalen of een levering een wijziging op eerdere aanlevering is zal TAS de gegevens moeten vergelijken. Deze identifier van de zitting respectievelijk deelname is hierbij leidend: als de id van de zitting/deelname overeenkomt met de id van eerdere aanlevering dan betreft dit dezelfde zitting/deelname.

OOAPI operaties en endpoints

Operatie en endpoint van de eerste interactie “Zitting wijzigen” in “Zittingsplan wijzigen” zijn:

PUT /offerings/{offeringId}

In **verzoek “2: Zitting wijzigen”** (zie Figuur 3.2.3) wordt de betreffende zitting geïdentificeerd door offeringId als padparameter (zie {offeringId}). Het verzoekbericht zelf bevat de gegevens van de zitting in een **object ComponentOffering**. Let op, dit endpoint en operatie ligt in het koppelvlak bij TAS.

Het antwoordbericht “3: Verwerkingsresultaat zitting” (zie Figuur 3.2.3) bevat bij succesvolle verwerking een positieve terugmelding.

Operatie en endpoint van de tweede interactie “Deelname wijzigen” in “Zittingsplan wijzigen” zijn:

PUT /associations/{associationId}

In het **verzoek “4: Deelname wijzigen”** (zie Figuur 3.2.3) wordt de betreffende deelname geïdentificeerd door associationId als padparameter (zie {associationId}). De betreffende zitting waarvan de deelname moeten worden gewijzigd wordt geïdentificeerd door gegeven **offeringId** in het object **Association**. In het verzoekbericht worden de gegevens van de Deelname meegegeven in object **Association**, inclusief de betreffende persoonsgegevens in gegeven person (object Person) binnen Association.

Deze interactie zal gewoonlijk in een reeks van een aantal deelnames bij de zitting worden uitgevoerd, eventueel voorafgegaan door de eerste interactie “Zitting wijzigen”. Maar deze interactie kan ook zelfstandig worden uitgevoerd, bijvoorbeeld wanneer één student zich afmeldt voor de toets.

Het antwoordbericht “5: Verwerkingsresultaat deelname” (zie Figuur 3.2.3) bevat bij succesvolle verwerking een positieve terugmelding.

Operatie en endpoint binnen OOAPI van de interactie “Zitting verwijderen” zijn:

PATCH /offerings/{offeringId}

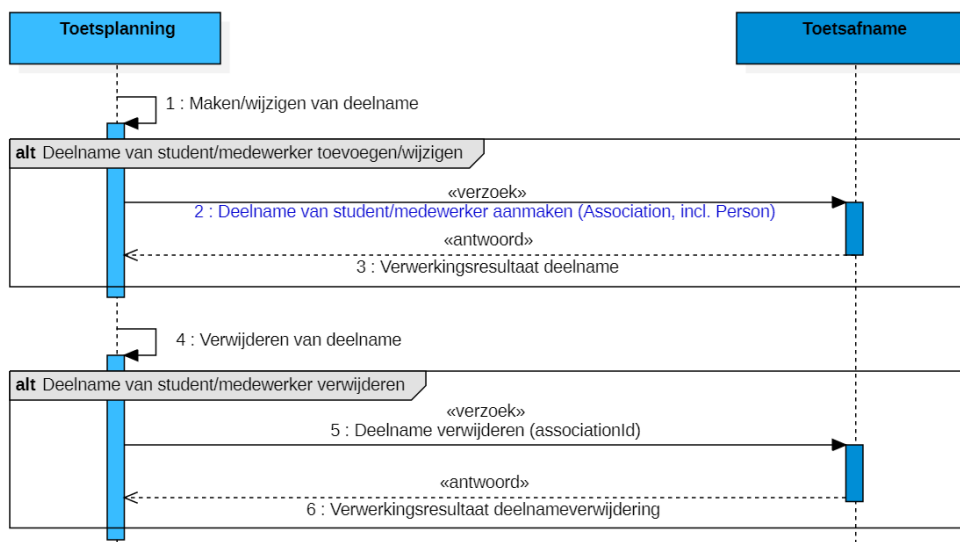
In het **verzoek “7: Zitting verwijderen”** (zie Figuur 3.2.3) wordt de specifieke Zitting (ComponentOffering) geïdentificeerd door offeringId als padparameter. In het verzoekbericht wordt expliciet de statuswijziging “canceled” meegestuurd door het gegeven **offeringState** binnen het object **ComponentOffering**.

Merk op dat als een Zitting in TAS wordt verwijderd dat tevens de bijbehorende deelnames aan deze zitting moeten worden verwijderd, en ook de gegevens van betreffende personen (student of medewerker) van deze deelnames.

Het antwoordbericht “8: Verwerkingsresultaat zittingsverwijdering” (zie Figuur 3.2.3) bevat bij succesvolle verwerking een positieve terugmelding.

Interacties “Deelname wijzigen”

In TPL moet een eerder aan TAS overgedragen deelname van student of medewerker (afnameleider, surveillant en beoordelaar) kunnen worden gewijzigd. Het brengen van het gewijzigde deelname valt uiteen in twee soorten wijzigingen: toevoegen/wijzigen of verwijderen. Het verwijderen van een deelname gebeurt door het insturen van een statuswijziging van de deelname. De interacties zien er als volgt uit:



Figuur 3.2.4: Deelnames wijzigen interactiemodel

Deze overdracht van de Deelnames is een **standlevering**, d.w.z. iedere keer worden alle gegevens volgens de actuele stand van zaken overgedragen. Deze interactie is daarmee **herhaalbaar** op elk gewenst moment.

Om te bepalen of een levering een wijziging op eerdere aanlevering is zal TAS de gegevens moeten vergelijken. Deze identifier van de deelname is hierbij leidend: als de id van de deelname overeenkomt met de id van eerdere aanlevering dan betreft dit dezelfde deelname.

OOAPI operaties en endpoints

Operatie en endpoint binnen OOAPI voor “Deelname van student/medewerker toevoegen/wijzigen” zijn:

PUT /associations/{associationId}

In het verzoek wordt de betreffende deelname geïdentificeerd door associationId als padparameter (zie {associationId}). In het **verzoekbericht** "2: Deelname van student/medewerker" (zie Figuur 3.2.4) worden de gegevens van de deelname en bijbehorende persoonsgegevens meegegeven in een **object Association**, inclusief object Person.

Het antwoordbericht "3: Verwerkingsresultaat deelname" (zie Figuur 3.2.4) bevat bij succesvolle verwerking een positieve terugmelding.

Operatie en endpoint binnen OOAPI van de interactie “Deelname verwijderen” zijn:

PATCH /associations/{associationId}

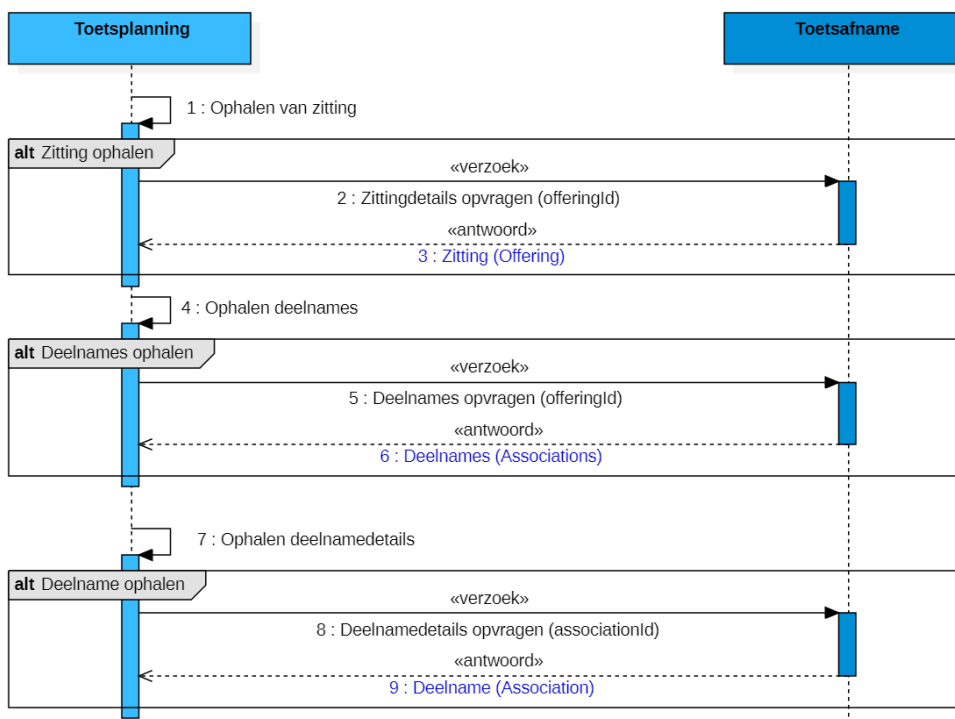
In het verzoek wordt de specifieke deelname geïdentificeerd door associationId als padparameter. In het **verzoekbericht** "5: Deelname verwijderen" (zie Figuur 3.2.4) wordt expliciet de statuswijziging “canceled” meegestuurd in de Deelname (**object Association**).

Merk op dat als een Deelname (Association) in TAS wordt verwijderd dat tevens de persoonsgegevens van de betreffende persoon (student of medewerker) moeten worden losgekoppeld/verwijderd.

Het antwoordbericht "6: Verwerkingsresultaat deelnameverwijdering" (zie Figuur 3.2.4) bevat bij succesvolle verwerking een positieve terugmelding.

Interacties “Zittingsdetails ophalen”

In TPL kan het voorkomen dat de actuele details van de zitting of de deelnames moeten worden opgehaald. Dit betreft details van een specifieke zitting of deelname. De interacties zien er als volgt uit:



Figuur 3.2.5: Zittingsdetails ophalen interactiemodel

Deze opvraging van de zittingsdetails is een **standlevering**, d.w.z. iedere keer worden alle gegevens volgens de actuele stand van zaken overgedragen. Deze interactie is daarmee **herhaalbaar** op elk gewenst moment. De identifier van de zitting respectievelijk deelname is leidend: als de id overeenkomt met de id van eerdere aanlevering dan zal dit gegevensverzoek tot de gewenste gegevens in het antwoordbericht leiden.

OOAPI operaties en endpoints

Operatie en endpoint van de eerste interactie in “Zitting ophalen” zijn:

GET /offerings/{offeringId}

In het verzoek “2: Zittingsdetails opvragen” (zie Figuur 3.2.5) wordt de gevraagde zitting geïdentificeerd door offeringId als padparameter.

Het **antwoordbericht** “3: Zitting” bevat bij succesvolle aanroep de gegevens van de zitting in een **object ComponentOffering**.

Operatie en endpoint van de tweede interactie “Deelnames ophalen” zijn:

GET /offerings/{offeringId}/associations

In het verzoek “5: Deelnames opvragen” (zie Figuur 3.2.5) wordt de specifieke zitting geïdentificeerd door offeringId als padparameter.

Het **antwoordbericht** “6: Deelnames” bevat bij succesvolle aanroep de gegevens over de deelnames in een lijst van Deelnames (**object Association**).

Operatie en endpoint van de derde interactie “Deelname ophalen” zijn:

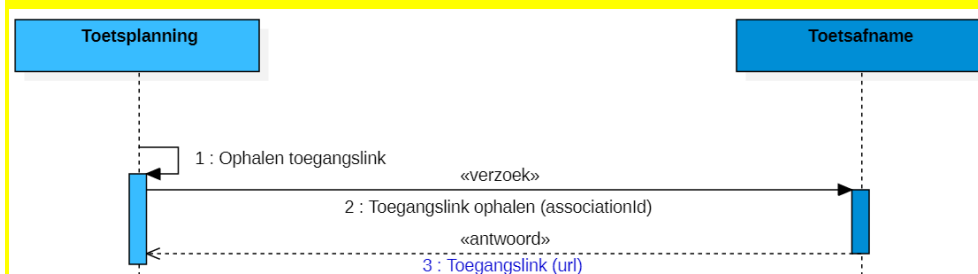
GET /associations/{associationId}

In het verzoek “8: Deelnamedetails opvragen” (zie Figuur 3.2.5) wordt de specifieke deelname geïdentificeerd door associationId als padparameter.

Het **antwoordbericht** “9: Deelname” bevat bij succesvolle aanroep de gegevens over de deelname in **object Association**.

Interactie “Toegangslink ophalen”

In TPL kan het nodig zijn om de student een toegangslink (startup-url) naar de toetsapplicatie toe te sturen betreffende de deelname aan een specifieke zitting. Met deze toegangslink kan de student eenvoudig naar de toetsapplicatie navigeren. TPL kan hiervoor de toegangslink ophalen bij TAS. De interactie ziet er als volgt uit:



Figuur 3.2.5A: Toegangslink ophalen interactiemodel

Operatie en endpoint van de derde interactie “Deelname ophalen” zijn:

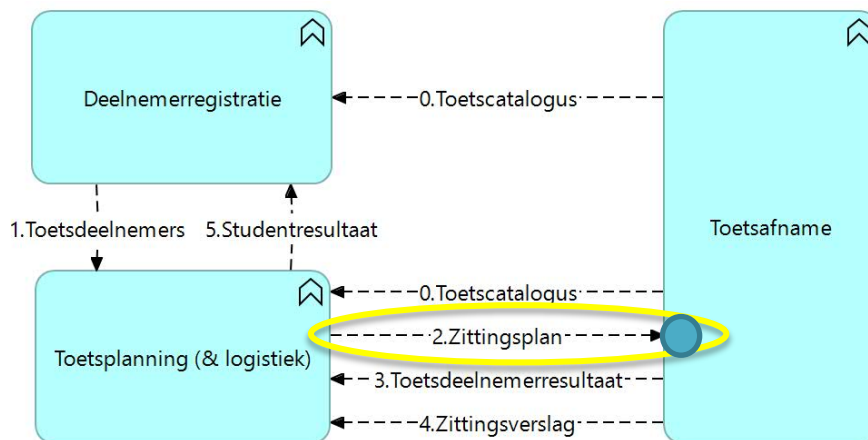
GET /associations/{associationId}/url

In het verzoek “8: Toegangslink ophalen” (zie Figuur 3.2.5A) wordt de toegangslink bij een specifieke deelname opgevraagd. De deelname wordt geïdentificeerd door associationId als padparameter.

Het **antwoordbericht** “3: Toegangslink” bevat bij succesvolle aanroep de url voor de betreffende deelname.

3.2.3 Koppelvlak bij Toetsafname

Het koppelvlak bij Toetsafname (TAS) voor informatiestroom Zittingsplan is in onderstaand figuur met blauwe cirkel gemarkeerd.



Figuur 3.2.6: Koppelvlak bij Toetsafname voor Zittingsplan

Operatie & Endpoint	Verzoekbericht	Antwoordbericht
GET /associations/{associationId}	Padparameters: associationId=uuid	Inhoud Deelname (ComponentOfferingAssociation), zie Figuur 3.2.7
PATCH /associations/{associationId}	Padparameters: associationId=uuid Inhoud Deelname (ComponentOfferingAssociation), zie Figuur 3.2.9	Statuscode
PUT /associations/{associationId}	Padparameters: associationId=uuid Inhoud Deelname (ComponentOfferingAssociation), zie Figuur 3.2.7	Statuscode
GET /associations/{associationId}/url	Padparameters: associationId=uuid	Inhoud url
GET /offerings/{offeringId}	Padparameter: offeringId=uuid	Inhoud Zitting (ComponentOffering), zie Figuur 3.2.10
PUT /offerings/{offeringId}	Padparameter: offeringId=uuid Inhoud Zitting (ComponentOffering), zie Figuur 3.2.10	Statuscode
PATCH /offerings/{offeringId}	Padparameter: offeringId=uuid Inhoud Zitting (ComponentOffering), zie Figuur 3.2.11	Statuscode
GET /offerings/{offeringId}/associations/	Padparameter: offeringId=uuid	Inhoud Lijst van Deelnames (ComponentOfferingAssociation), zie Figuur 3.2.7

Tabel 3.2.1: Endpoints en operaties in koppelvlak bij Toetsafname voor Zittingsplan

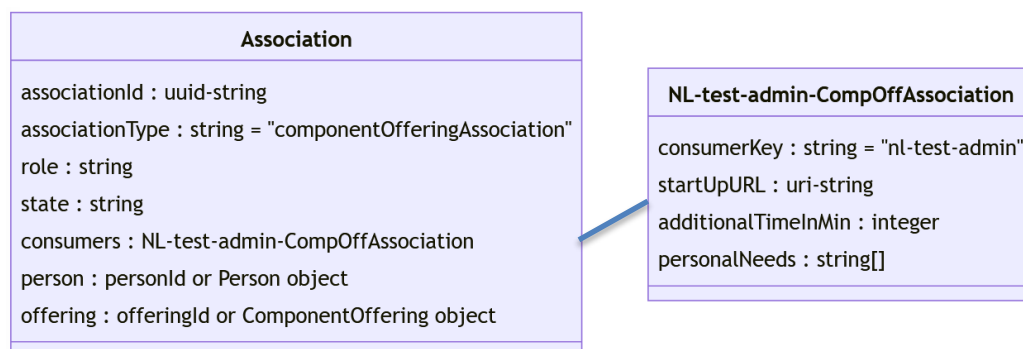
De aanvrager (verzoeker/initiator) van alle endpoint-operaties in bovenstaande tabel is TPL.

Het antwoordbericht van **GET /associations/{associationId}** en **GET /offerings/{offeringId}/associations** bevatten hier het object respectievelijk objecten Association van het type ComponentOfferingAssociation.

In deze situatie (Informatiestroom 2) wordt gegeven **result** binnen object **Association** niet gebruikt. Dit gegeven result wordt in informatiestroom 3 gebruikt om het toetsresultaat van een toetsdeelnemer over te dragen.

Wanneer deze deelname wel al een resultaat heeft in TAS dan zal dit resultaat in het antwoordbericht van de GET operatie worden meegestuurd. In deze situatie (Informatiestroom 2) worden gegevens **attempt**, **attemptLeft**, **programOfferingAssociationId** en **courseOfferingAssociationId** in het gegevensblok consumers van object **Association** niet gebruikt; deze zijn bedoeld voor informatiestroom 1. Bij deze opvraging is middels query parameters zowel het id als het object bij deze gegevens mogelijk (zie Association in onderstaand Figuur 3.2.7).

Let op, bij het opvragen van de Deelnames (objecten Association) is er de mogelijkheid om in het antwoordbericht tevens de gegevens over de betreffende persoon (Person) mee te krijgen. Hiertoe moet de query-parameter expand worden gebruikt, zoals "GET /associations/{associationId}?expand=person".

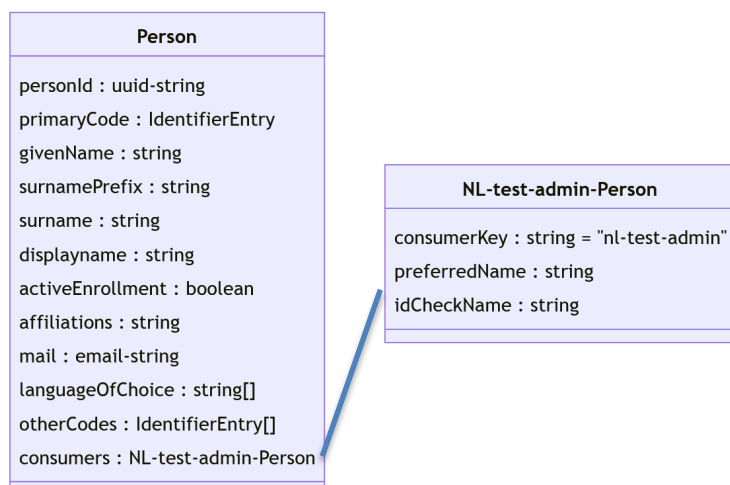


Figuur 3.2.7: Association (Deelname) gegevensmodel (volledig voor GET) in 2. Zittingsplan

Het verzoekbericht van `PUT /associations/{associationId}` bevat hier het object Association van het type ComponentOfferingAssociation, inclusief associationId.

In deze situatie (Informatiestroom 2) wordt gegeven **result** binnen object **Association** niet gebruikt. Dit gegeven result wordt in informatiestroom 3 gebruikt om het toetsresultaat van een toetsdeelnemer over te dragen. In deze situatie (Informatiestroom 2) worden gegevens **attempt**, **attemptLeft**, **programOfferingAssociationId** en **courseOfferingAssociationId** in het gegevensblok consumers van object **Association** niet gebruikt; deze zijn bedoeld voor informatiestroom 1.

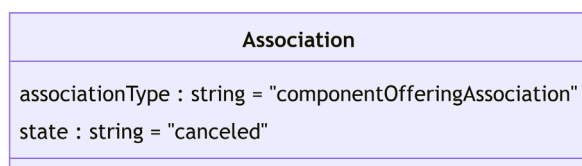
Bij aanlevering van een Deelname (**Association**) in de PUT operatie wordt het object Person onder gegeven person altijd direct meegeleverd (zie Figuur 3.2.8). Het object Offering onder gegeven **offering** wordt eerst apart zelf aangeleverd waardoor dit gegeven niet als object maar als verwijzing naar het object (offeringId) wordt gespecificeerd (zie Figuur 3.2.7).



Figuur 3.2.8: Person (Persoon) gegevensmodel in 2. Zittingsplan

Het verzoekbericht van `PATCH /associations/{associationId}` bevat hier het object Association van het type ComponentOfferingAssociation met de statuswijziging "canceled".

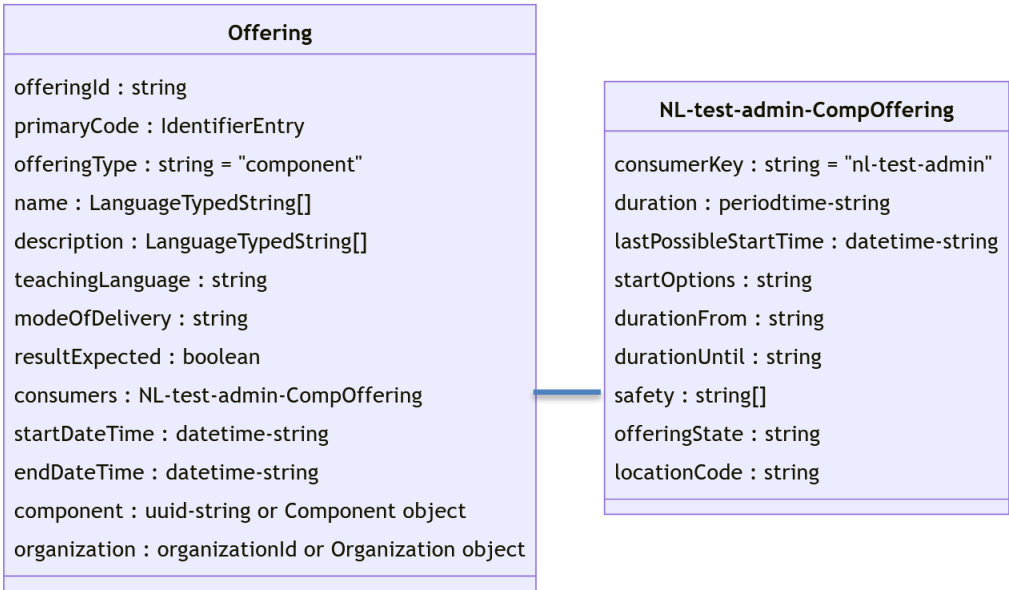
Aanvullende eis: In het verzoekbericht zijn in deze situatie (Verwijdering van deelname) gegevens **state** (met waarde "canceled") en **associationType** binnen **object Association** verplicht; alle overige gegevens binnen Association zijn overbodig want deze zullen door TAS worden genegeerd (zie onderstaand Figuur 3.2.9).



Figuur 3.2.9: Association (Deelname) gegevensmodel t.b.v. Deelname verwijderen in 2. Zittingsplan

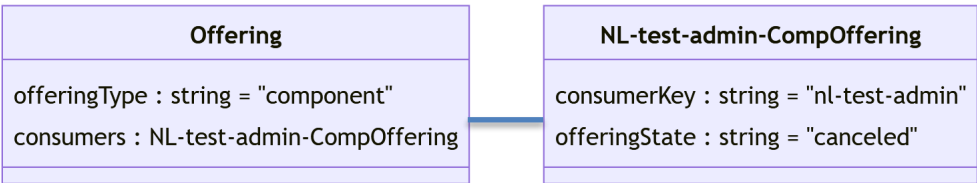
Het antwoordbericht van `GET /offerings/{offeringId}` en het verzoekbericht van `PUT /offerings/{offeringId}` bevatten hier het object ComponentOffering inclusief offeringId zonder de resultaatgegevens.

In deze situatie (Informatiestroom 2) worden **gegevens irregularities, documents en testsToBeUsed** in het gegevensblok consumers van **object ComponentOffering** niet gebruikt. De gegevens irregularities en documents worden in informatiestroom 4 gebruikt om het resultaat van de zitting over te dragen; gegeven testsToBeUsed wordt in informatiestroom 1 gebruikt om een planbare toets over te dragen (zie onderstaand Figuur 3.2.10).



Figuur 3.2.10: Offering (Zitting) gegevensmodel t.b.v. 2. Zittingsplan

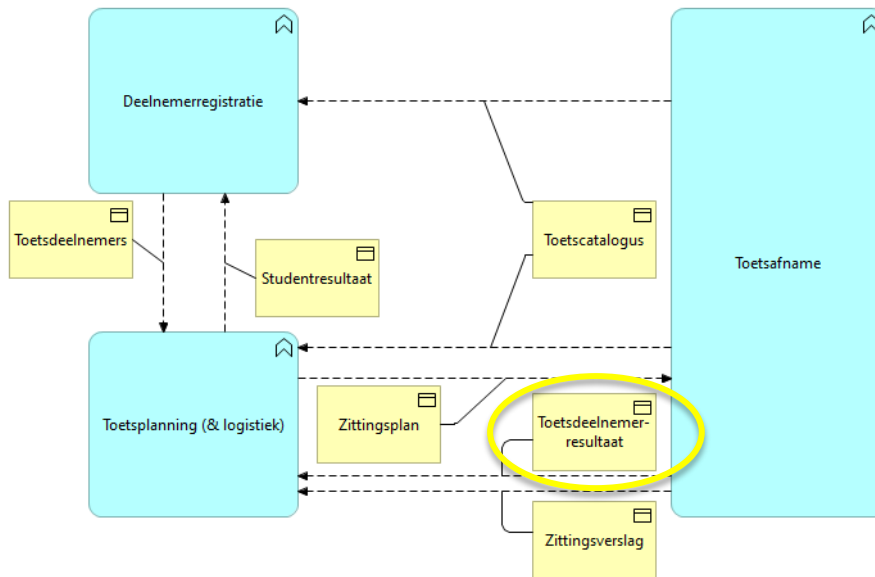
Het verzoekbericht van [PATCH /offerings/{offeringId}](#) bevat hier het object ComponentOffering met de statuswijziging "canceled".
Aanvullende eis: In het verzoekbericht zijn in deze situatie (Verwijdering van zitting) gegevens **offeringState** (met waarde "canceled" via **consumers**) en **offeringType** binnen object **ComponentOffering** verplicht; alle overige gegevens binnen Offering zijn overbodig want deze zullen door TAS worden genegeerd (zie onderstaand Figuur 3.2.11).



Figuur 3.2.11: Offering (Zitting) gegevensmodel t.b.v. Zitting verwijderen in 2. Zittingsplan

3.3 Toetsdeelnemerresultaat informatiestroom

De informatiestroom Toetsdeelnemerresultaat (flow 3) betreft het informatieobject Toetsdeelnemerresultaat dat in onderstaand figuur met gele ovaal is gemarkeerd.



Figuur 3.3.1A: Informatieobject Toetsdeelnemerresultaat in het scenario

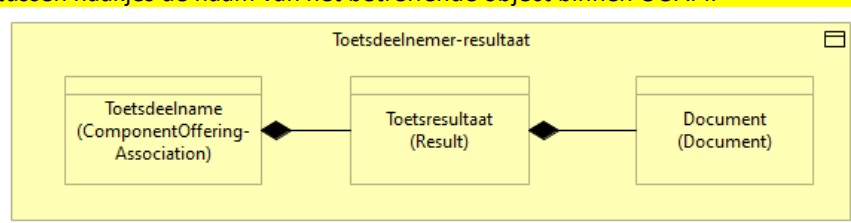
Ten behoeve van de processen “Vaststellen van resultaat” en “Vastleggen van resultaat” (zie Figuur 2.B in Hoofdstuk 2) heeft **Toetsplanning (& logistiek)** behoefte aan informatie over de individuele resultaten van de toets (examen). Dit informatieobject Toetsdeelnemerresultaat wordt beschikbaar gesteld in **Toetsafname** en overgedragen in informatiestroom 3. **Toetsdeelnemerresultaat**.

Gegevensverstrekker van het Toetsdeelnemerresultaat is Toetsafname (TAS). De gegevensafnemer is de Toetsplanning (TPL). De informatie over het Toetsdeelnemerresultaat gaat van TAS naar TPL. In de overdracht van de Toetsdeelnemerresultaat neemt TAS het initiatief zodra de informatie over de aanwezigheid en/of behaalde score van de student bekend is en gereed is voor verzending. TPL zorgt ervoor dat het Toetsdeelnemerresultaat correct wordt verwerkt.

Voordat de gegevensuitwisseling kan plaatsvinden zijn TPL en TAS al technisch met elkaar verbonden, m.b.t. base-URL, authenticatie, etc. en hebben eerder het zittingsplan met de deelnames onderling uitgewisseld.

3.3.1 Informatieobject Toetsdeelnemerresultaat

In het volgende figuur staan de informatieobjecten binnen het informatieobject Toetsdeelnemerresultaat, met tussen haakjes de naam van het betreffende object binnen OOAPI.



Figuur 3.3.1B: Informatieobject Toetsdeelnemerresultaat in detail

Het **informatieobject Toetsdeelnemerresultaat** omvat de informatie over de resultaten van de toetsafname door de individuele student. Dit resultaat omvat de informatie over de aanwezigheid en eventueel behaalde score bij de afname van de toets (of examen) van de betrokken student. Het informatieobject Toetsresultaat is onderdeel van de betreffende deelname en het informatieobject Toetsdeelname is daardoor ook onderdeel van Toetsdeelnemerresultaat. Via Toetsdeelname is het Toetsresultaat daarmee indirect gerelateerd aan de student en de zitting. Eventueel vormen gerelateerde documenten ook onderdeel van Toetsresultaat.

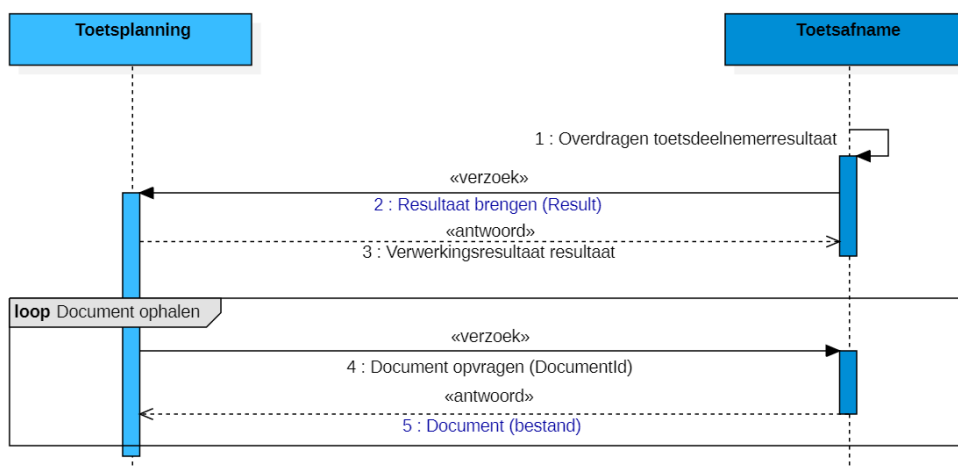
3.3.2 Interacties voor Toetsdeelnemerresultaat

Het algemene transactiepatroon is “Push” zodat de resultaatgegevens door Toetsafname (TAS) bij Toetsplanning (TPL) worden gebracht. In de resultaatgegevens zitten verwijzingen naar bijlagedocumenten die door TPL kunnen worden opgehaald. Voorbeelden van bijlagedocumenten zijn het beoordelingsformulier zoals ingevuld door de beoordelaar, het antwoordblad met beoordelingsnotities of het beoordelingsmodel/beoordelingsvoorschrift. Deze informatiestroom bevat de interacties “Deelnemerresultaat brengen”, “Eerst aanwezigheid en later resultaat brengen” en “Aanwezigheid/resultaten ophalen”.

Interacties “Toetsdeelnemerresultaat brengen”

In TAS moet het voorlopige of definitieve resultaat van de student kunnen worden overgedragen aan TPL. Een voorlopig resultaat is een voorlopige score dat nog niet is vastgesteld door de examencommissie; een definitief resultaat is de uiteindelijke score die is vastgesteld door de examencommissie.

Het brengen van het Toetsdeelnemerresultaat bij TPL gaat in stappen: eerst de gestructureerde gegevens over het resultaat en vervolgens kunnen de bijbehorende documenten worden opgehaald door TPL bij TAS. De interacties zien er als volgt uit:



Figuur 3.3.2: Toetsdeelnemerresultaat brengen interactiemodel

OOAPI operaties en endpoints

Operatie en endpoint binnen OOAPI van de eerste interactie “Resultaat brengen” zijn:

PATCH /associations/{associationId}

In het **verzoek** “2: Resultaat brengen” (zie Figuur 3.3.2) wordt de betreffende deelname geïdentificeerd door associationId als padparameter (zie {associationId}). In het verzoekbericht zelf worden de gegevens van het individuele **resultaat** meegegeven in gegeven result (object Result) binnen **object Association**.

Let op, dit endpoint en operatie ligt in het koppelvak bij TPL.

Het antwoordbericht “3: Verwerkingsresultaat resultaat” (zie Figuur 3.3.2) bevat bij succesvolle verwerking een positieve terugmelding.

Operatie en endpoint binnen OOAPI van de tweede interactie “Document ophalen” zijn:

GET /documents/{documentId}

In het verzoek “4: Document opvragen” (zie Figuur 3.3.2) wordt het specifieke document geïdentificeerd door documentId als padparameter.

Het **antwoordbericht** “5: Document” bevat bij succesvolle aanroep het document in een **bestand**.

Let op, dit endpoint en operatie ligt in het koppelvak bij TAS.

Interacties “Eerst aanwezigheid en later resultaat brengen”

In TPL is het belangrijk om de aanwezigheid van een student bij een toets/examen tijdig te weten, bijvoorbeeld om een nieuwe deelname te plannen. Zeker in de gevallen dat het voorlopige of definitieve resultaat van de student nog enige tijd kan, kan TAS alvast de gegevens over de aanwezigheid van een student per deelname overdragen. In een latere overdracht wordt dan het resultaat gebracht.

Het brengen van eerste de aanwezigheid en later het resultaat bij TPL gaat allebei in dezelfde interacties als in voorgaande Figuur 3.3.2.

OOAPI operaties en endpoints

Operatie en endpoint binnen OOAPl van de interactie “Aanwezigheid brengen” zijn eveneens:

PATCH /associations/{associationId}

In het **verzoek “2: Resultaat brengen”** (zie Figuur 3.3.2) wordt de betreffende deelname geïdentificeerd door associationId als padparameter (zie {associationId}). In het verzoekbericht zelf worden de gegevens van de individuele aanwezigheid (**aanwezigheidsinfo**) meegegeven in gegeven attendance binnen **object Association**. Voor het brengen van het voorlopige of definitieve resultaat (scoreresultaat) worden de alle resultaatgegevens meegegeven in gegeven result binnen object Association.

Let op, dit endpoint en operatie ligt in het koppelvak bij TPL.

Het antwoordbericht “3: Verwerkingsresultaat resultaat” (zie Figuur 3.3.2) bevat bij succesvolle verwerking een positieve terugmelding.

Operatie en endpoint binnen OOAPl van de interactie “Aanwezigheid ophalen” zijn:

GET /documents/{documentId}

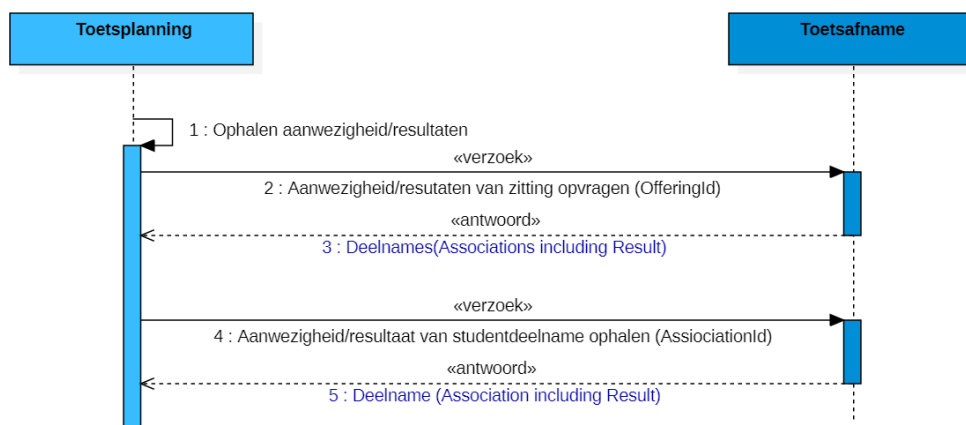
In het verzoek “4: Document opvragen” (zie Figuur 3.3.2) wordt het specifieke document geïdentificeerd door documentId als padparameter.

Het **antwoordbericht “5: Document”** bevat bij succesvolle aanroep het document in een **bestand**.

Let op, dit endpoint en operatie ligt in het koppelvak bij TAS.

Interacties “Aanwezigheid en resultaten ophalen”

In TPL is het op bepaalde momenten belangrijk om de aanwezigheid en eventueel de studentresultaten van een zitting te weten, bijvoorbeeld om een nieuwe deelname of zitting te plannen. Alle resultaten van de studentdeelnames van een zitting of per individuele studentdeelnamen kunnen door TPL bij TAS worden opgehaald. De interacties zien er als volgt uit:



Figuur 3.3.3: Aanwezigheid/resultaten ophalen interactiemodel

OOAPI operaties en endpoints

Operatie en endpoint binnen OOAPl van de eerste interactie “Aanwezigheid/resultaat ophalen” zijn:

GET /offerings/{offeringId}/associations

In verzoek “2: Aanwezigheid/resultaat van zitting ophalen” (zie Figuur 3.3.3) wordt de betreffende zitting geïdentificeerd door offeringId als padparameter (zie {offeringId}).

Het **antwoordbericht “3: Deelnames”** bevat de gegevens van de individuele deelnames in een lijst van Deelnames (**object Association**).

Let op, dit endpoint en operatie ligt in het koppelvak bij TAS.

Operatie en endpoint binnen OOAPl van de tweede interactie “Aanwezigheid/resultaat ophalen” zijn:

GET /associations/{associationId}

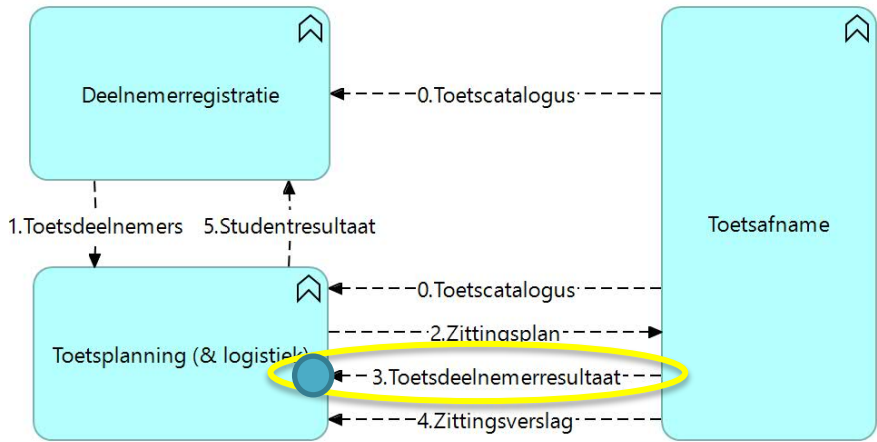
In het verzoek “4: Aanwezigheid/resultaat van studentdeelname ophalen” (zie Figuur 3.3.3) wordt de betreffende deelname geïdentificeerd door associationId als padparameter (zie {associationId}).

Het **antwoordbericht “5: Deelname”** bevat bij succesvolle aanroep het **object Association**.

Let op, dit endpoint en operatie ligt in het koppelvlak bij TAS.

3.3.3 Koppelvlak bij Toetsplanning

Het koppelvlak bij Toetsplanning (TPL) voor informatiestroom Toetsdeelnemerresultaat is in onderstaand figuur met blauwe cirkel gemarkeerd.



Figuur 3.3.4: Koppelvlak bij Toetsplanning voor Toetsdeelnemerresultaat

Operatie & Endpoint	Verzoekbericht	Antwoordbericht
PATCH /associations/{associationId}	Padparameter: associationId=uuid Inhoud Toetsdeelnemerresultaat (Association), zie Figuur 3.3.5	Statuscode

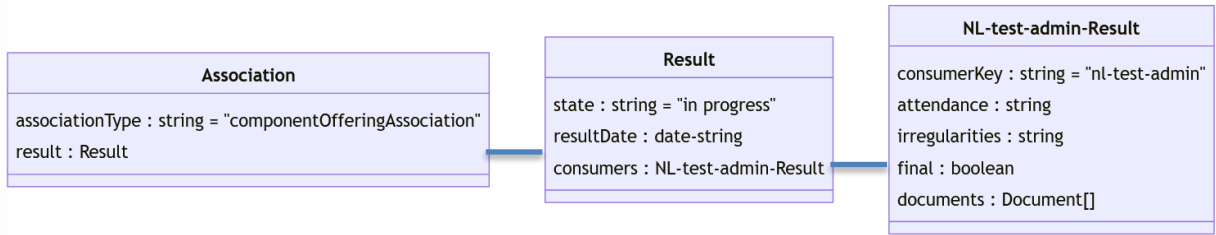
Tabel 3.3.1: Endpoints en operaties in koppelvlak bij Toetsplanning voor Toetsdeelnemerresultaat

De aanvrager (verzoeker/initiator) van deze endpoint-operatie in bovenstaande tabel is TAS.

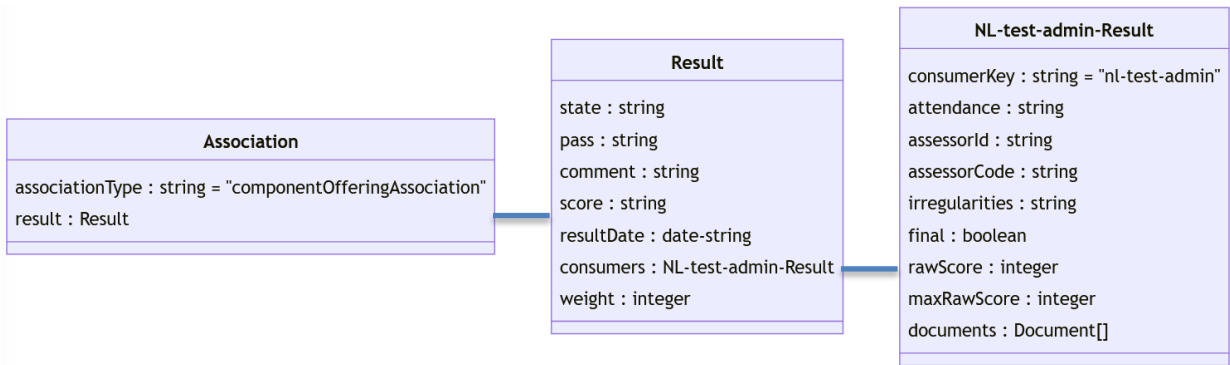
Het verzoekbericht van `PATCH /associations/{associationId}` bevat hier de gegevens m.b.t. de aanwezigheid en eventueel het resultaat van een deelname aan de zitting door een student of een medewerker (afnameleider, surveillant of beoordelaar) in object **Association**.

In het geval van aanwezigheidsinfo zal binnen het resultaat (object Result) het gegeven attendance, eventueel met gegeven final en bijgaande document(en), zeker worden gebruikt (zie onderstaand Figuur 3.3.5).

Aanvullende eis: In het verzoekbericht zijn in deze context (3. Toetsdeelnemerresultaat) gegevens **result** (object **Result**) en **associationType** binnen object **Association** verplicht; alle overige gegevens binnen Association zijn overbodig want deze zullen door TPL worden genegeerd (zie onderstaand Figuur 3.3.6).



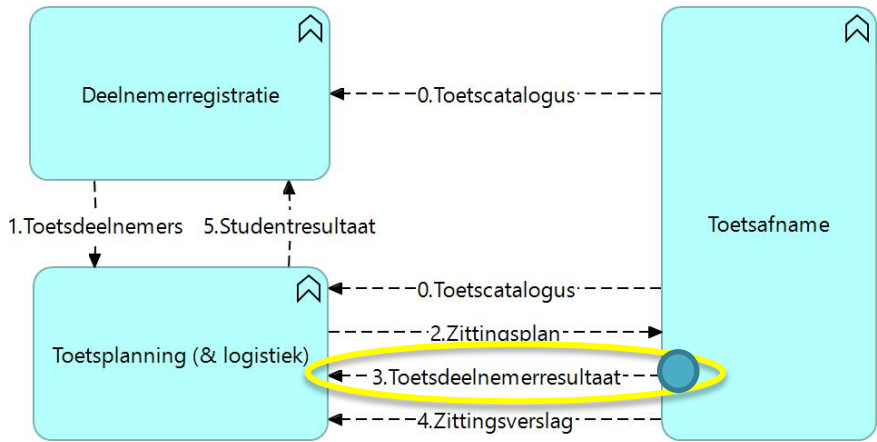
Figuur 3.3.5: Association gegevensmodel t.b.v. aanwezigheidsinfo in 3. Toetsdeelnemerresultaat



Figuur 3.3.6: Association gegevensmodel t.b.v. scoreresultaat in 3. Toetsdeelnemerresultaat

3.3.4 Koppelvlak bij Toetsafname

Het koppelvlak bij Toetsafname (TAS) voor informatiestroom Toetsdeelnemerresultaat is in onderstaand figuur met blauwe cirkel gemarkeerd.



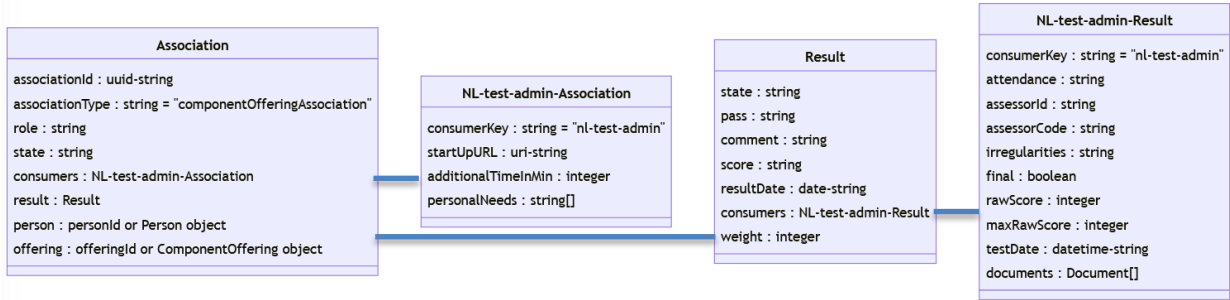
Figuur 3.3.7: Koppelvlak bij Toetsafname voor Toetsdeelnemerresultaat

Operatie & Endpoint	Verzoekbericht	Antwoordbericht
GET /associations/{associationId}	Padparameter: associationId=uuid	Inhoud Deelname (Association), zie Figuur 3.3.8
GET /documents/{documentId}	Padparameter: documentId=uuid	Bijlage Bestand
GET /offerings/{offeringId}/associations	Padparameter: offeringId=uuid	Inhoud Lijst van Deelnames (Association), zie Figuur 3.3.8

Tabel 3.3.2: Endpoints en operaties in koppelvlak bij Toetsafname in 2. Toetsdeelnemerresultaat

De aanvrager (verzoeker/initiator) van alle endpoint-operaties in bovenstaande tabel is TPL.

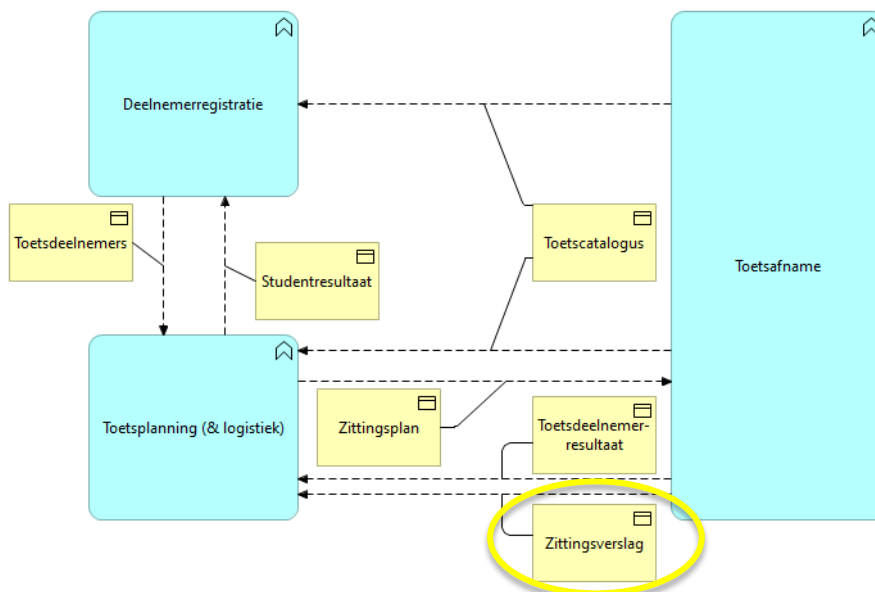
Het antwoordbericht van `GET /associations/{associationId}` en `GET /offerings/{offeringId}/associations` bevat hier de gegevens m.b.t. de Deelname van een student of een medewerker (afnameleider, surveillant of beoordelaar) in object Association (zie onderstaand Figuur 3.8).



Figuur 3.3.8: Association gegevensmodel t.b.v. resultaten ophalen in 2. Toetsdeelnemerresultaat

3.4 Zittingsverslag informatiestroom

De informatiestroom Zittingsverslag (flow 4) betreft het informatieobject Zittingsverslag dat in onderstaand figuur met gele ovaal is gemarkeerd.



Figuur 3.4.1: Informatieobject Zittingsverslag in het scenario

Ten behoeve van de processen “Vaststellen van resultaat” en “Vastleggen van resultaat” (zie Figuur 2.B in Hoofdstuk 2) heeft **Toetsplanning (& logistiek)** behoefte aan informatie over het verloop van de zitting. Dit informatieobject Zittingsverslag wordt beschikbaar gesteld in **Toetsafname** en overgedragen in informatiestroom 4. **Zittingsverslag**.

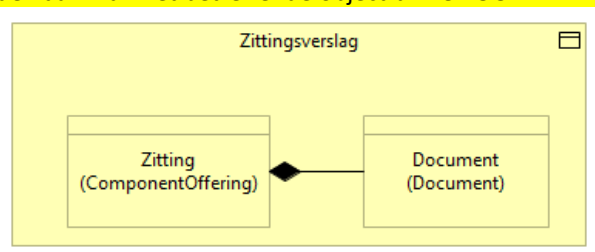
Gegevensverstrekker van het Zittingsverslag is de Toetsafname (TAS). De gegevensafnemer is de Toetsplanning (TPL). De informatie in het Zittingsverslag gaat van TAS naar TPL.

In de overdracht van het Zittingsverslag neemt TAS het initiatief zodra de informatie over het verslag gedeeltelijk of geheel bekend is en gereed is voor verzending. TPL zorgt ervoor dat het Zittingsverslag correct wordt verwerkt.

Voordat de gegevensuitwisseling kan plaatsvinden zijn TPL en TAS al technisch met elkaar verbonden, m.b.t. base-URL, authenticatie, etc. en hebben eerder het zittingsplan met de deelnames onderling uitgewisseld.

3.4.1 Informatieobject Zittingsverslag

In het volgende figuur staan de informatieobjecten binnen het informatieobject Zittingsverslag, met tussen haakjes de naam van het betreffende object binnen OOAPI.



Figuur 3.4.1B: Informatieobject Zittingsverslag in detail

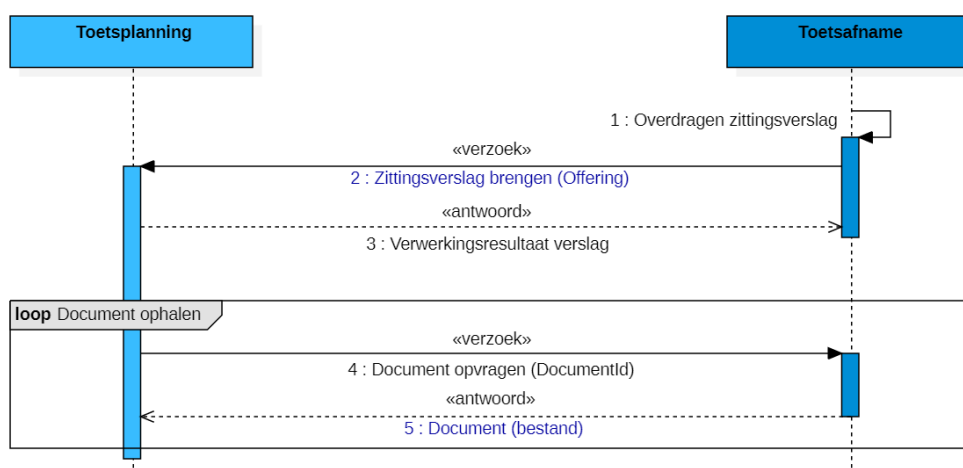
Het **informatieobject Zittingsverslag** omvat de informatie over de aanwezigheid en procesverbaal bij de afname van de toets (of examen) van de betrokken studenten in het informatieobject Zitting. Eventueel vormen gerelateerde documenten ook onderdeel van Zitting.

3.4.2 Interacties voor Zittingsverslag

Het algemene transactiepatroon is “Push” zodat de verslaggegevens door Toetsafname (TAS) bij Toetsplanning (TPL) worden gebracht. In de verslaggegevens zitten verwijzingen naar bijlagedocumenten die door TPL kunnen worden opgehaald.

Interacties “Zittingsverslag brengen”

In TAS moet het groepsresultaat van de zitting kunnen worden overgedragen aan TPL. Het brengen van het Zittingsverslag bij TPL gaat in stappen: eerst de gestructureerde gegevens over het verslag en vervolgens kunnen de bijbehorende documenten worden opgehaald door TPL bij TAS. De interacties zien er als volgt uit:



Figuur 3.4.2: Zittingsverslag brengen interactiemodel

OOAPI operaties en endpoints

Operatie en endpoint binnen OOAPI van de eerste interactie in “Zittingsverslag brengen” zijn:

PATCH /offerings/{offeringId}

In verzoek “2: Zittingsverslag brengen” (zie Figuur 3.4.2) wordt de betreffende zitting geïdentificeerd door offeringId als padparameter (zie {offeringId}). In het verzoekbericht zelf worden de gegevens van het resultaat van de zitting meegegeven in een **object ComponentOffering**. Let op, dit endpoint en operatie ligt in het koppelvlak bij TPL.

Aanvullende eis: In het verzoekbericht zijn in deze situatie (Zittingsverslag brengen) de volgende gegevens binnen het gegevensblok consumers van **object ComponentOffering** verplicht:

- Gegevensveld **consumerKey**,
- Gegevensveld **irregularities**
- Gegevensgroep **documents** met gegevensvelden **documentId**, **documentType** en **documentName**

Alle overige attributen binnen Offering zijn overbodig want deze zullen door TPL worden genegeerd.

Het antwoordbericht “3: Verwerkingsresultaat verslag” (zie Figuur 3.4.2) bevat bij succesvolle verwerking een positieve terugmelding.

Operatie en endpoint binnen OOAPI van de tweede interactie “Document ophalen” zijn:

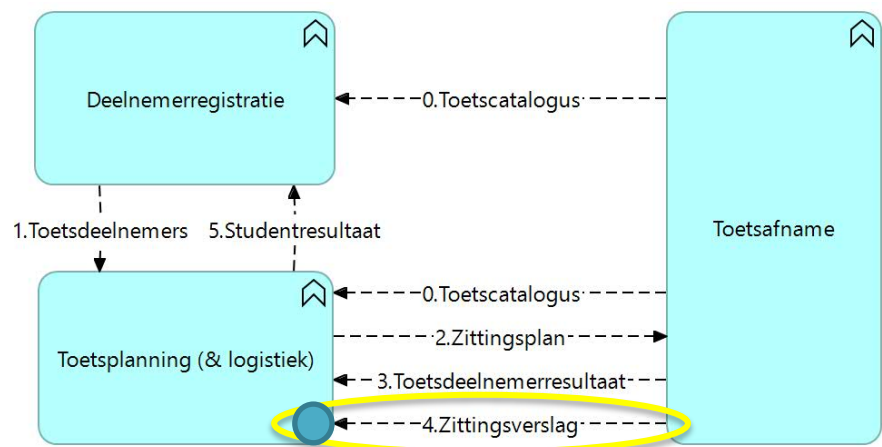
GET /documents/{documentId}

In verzoek “4: Document opvragen” (zie Figuur 3.4.2) wordt het specifieke document geïdentificeerd door documentId als padparameter.

Het antwoordbericht “5: Document” bevat bij succesvolle aanroep het gevraagde document in een **bestand**. Let op, dit endpoint en operatie ligt in het koppelvlak bij TAS.

3.4.3 Koppelvlak bij Toetsplanning

Het koppelvlak bij Toetsplanning (TPL) voor informatiestroom Zittingsverslag is in onderstaand Figuur 3.4.3 met blauwe cirkel gemarkeerd.



Figuur 3.4.3: Koppelvlak bij Toetsplanning voor Zittingsverslag

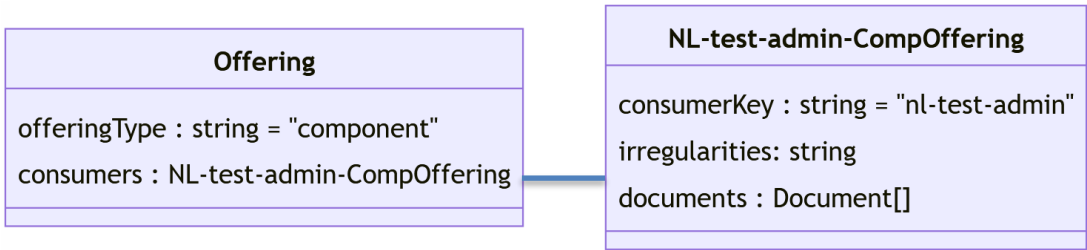
Operatie & Endpoint	Verzoekbericht	Antwoordbericht
PATCH /offerings/{offeringId}	Padparameter: offeringId=uuid Inhoud Zittingsverslag (ComponentOffering), zie Figuur 3.4.4	Statuscode

Tabel 3.4.1: Endpoints en operaties in koppelvlak bij Toetsplanning voor Zittingsverslag

De aanvrager (verzoeker/initiator) van de endpoint-operatie in bovenstaande tabel is TAS.

Het verzoekbericht van PATCH /associations/{associationId} bevat hier de gegevens m.b.t. het Zittingsverslag in object ComponentOffering.

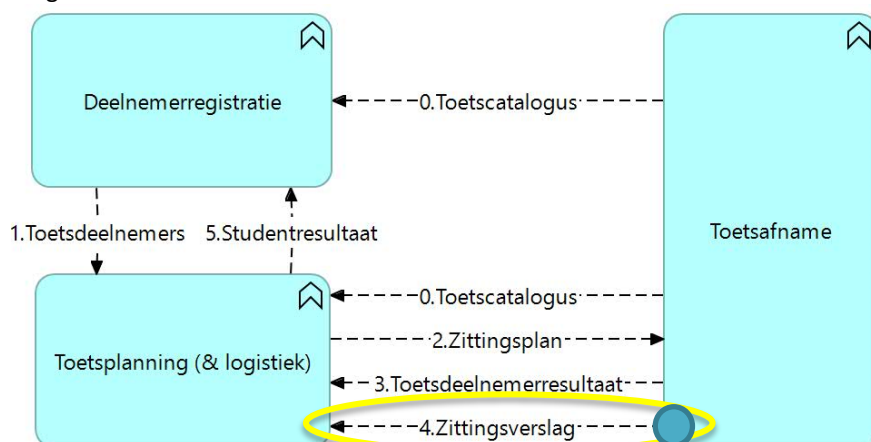
Aanvullende eis: In het verzoekbericht zijn in deze context (3. Zittingsverslag) gegevens **irregularities** (met eventueel **documents** via **consumers**) en **associationType** binnen object **ComponentOffering** verplicht; alle overige gegevens binnen Offering zijn overbodig want deze zullen door TPL worden genegeerd (zie onderstaand Figuur 3.4.4).



Figuur 3.4.4: Offering (Zitting) gegevensmodel in 4. Zittingsverslag

3.4.4 Koppelvlak bij Toetsafname

Het koppelvlak bij Toetsafname (TAS) voor informatiestroom Zittingsverslag is in onderstaand figuur met blauwe cirkel gemarkeerd.



Figuur 3.4.5: Koppelvlak bij Toetsafname voor Zittingsverslag

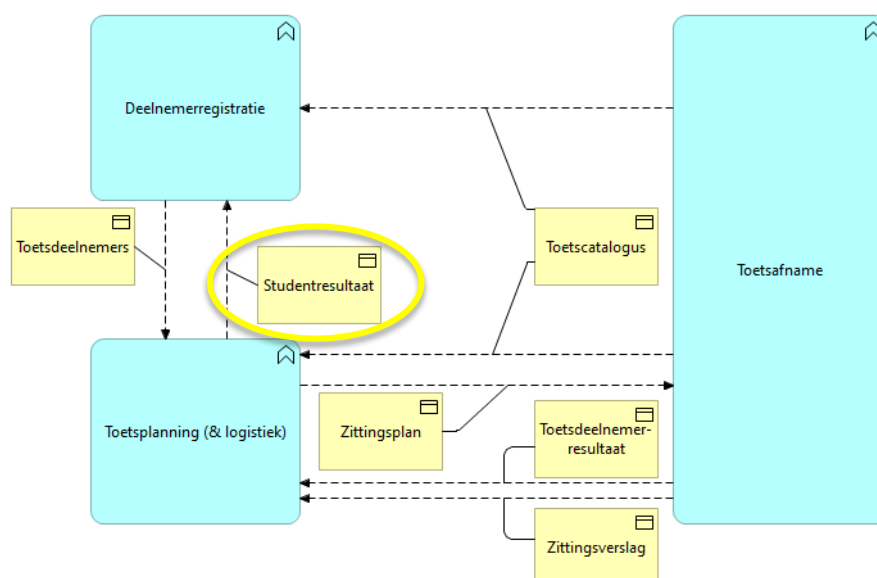
Operatie & Endpoint	Verzoekbericht	Antwoordbericht
GET /documents/{documentId}	Padparameter: documentId=uuid	Bijlage Bestand

Tabel 3.4.2: Endpoints en operaties in koppelvlak bij Toetsafname voor Zittingsverslag

De aanvrager (verzoeker/initiator) van de endpoint-operatie in bovenstaande tabel is TPL.

3.5 Studentresultaat informatiestroom

De informatiestroom Studentresultaat (flow 5) betreft het informatieobject Studentresultaat dat in onderstaand figuur met gele ovaal is gemarkeerd.



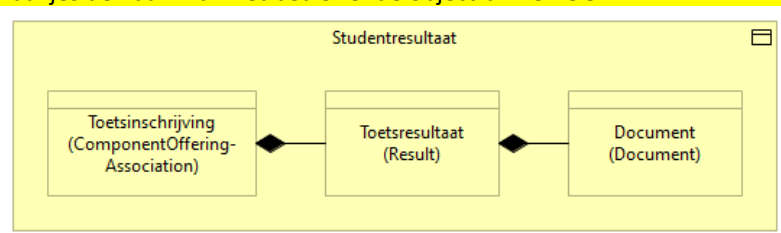
Figuur 3.5.1: Informatieobject Studentresultaat in het scenario

Ten behoeve van het proces “Publiceren van resultaat” (zie Figuur 2.B in Hoofdstuk 2) heeft **Deelnemerregistratie** of SIS (DR/SIS) behoefte aan informatie over het studentresultaat van de toets (of het examen). Dit informatieobject Studentresultaat wordt beschikbaar gesteld in **Toetsplanning (& logistiek)** en overgedragen in informatiestroom 5. **Studentresultaat**.

Gegevensverstrekker van de Studentresultaat met informatie over het resultaat van een student behaald in de toets/examen het systeem/applicatie van de mbo-instelling met de Deelnemerregistratie-functionaliteit. De gegevensafnemer zijn de systemen met de Toetsplanning-functionaliteit. De informatie over het Studentresultaat gaat van Toetsplanning (TPL) naar DR/SIS.

3.5.1 Informatieobject Studentresultaat

In het volgende figuur staan de informatieobjecten binnen het informatieobject Studentresultaat, met tussen haakjes de naam van het betreffende object binnen OOAPI.



Figuur 3.1.1B: Informatieobject Studentresultaat in detail

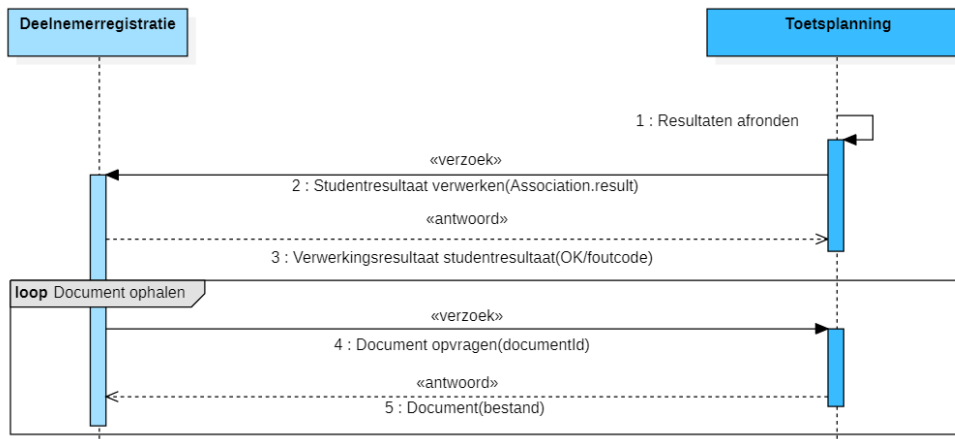
Het **informatieobject Studentresultaat** omvat de individuele informatie over de aanwezigheid, behaalde score en eventueel bijbehorende documenten als resultaat van de toetsafname (of examenafname) van de student in informatieobject Toetsresultaat. Het Toetsresultaat is onderdeel van de Toetsinschrijving en daarmee wordt dit resultaat indirect gekoppeld aan de betreffende Toets. Eventueel vormen gerelateerde documenten ook onderdeel van Toetsresultaat.

3.5.2 Interacties bij Studentresultaat

Het algemene transactiepatroon is “Push” zodat de toetsresultaten van de student bij DR/SIS worden gebracht. In dit studentresultaat zit verwijzingen naar bijlagedocumenten die door DR/SIS bij TPL kunnen worden opgehaald.

Interacties “Studentresultaat brengen”

In TPL moet het studentresultaat van de toetsafname kunnen worden overgedragen aan DR/SIS. Het brengen van het Studentresultaat bij DR/SIS gaat eventueel in stappen: eerst de gestructureerde resultaatgegevens en vervolgens kunnen de bijbehorende documenten worden opgehaald door DR/SIS bij TPL. De interacties zien er als volgt uit:



Figuur 3.5.2: Studentresultaat brengen interactiemodel

OOAPI operaties en endpoints

Operatie en endpoint binnen OOAPI van de eerste interactie “Student-toetsresultaat brengen” zijn:

PATCH /associations/{associationId}

In **verzoek “2: Studentresultaat verwerken”** (zie Figuur 3.5.2) wordt de betreffende inschrijving geïdentificeerd door **associationId** als padparameter (zie {associationId}). In het verzoekbericht zelf worden de gegevens van het studentresultaat meegegeven in een **object Association**. Let op, dit endpoint en operatie ligt in het koppelvak bij DR/SIS.

Het antwoordbericht “3: Verwerkingsresultaat studentresultaat” (zie Figuur 3.5.2) bevat bij succesvolle verwerking een positieve terugmelding.

In geval er op basis van een toetsinschrijving resultaten van meerdere pogingen worden overgedragen moet TPL een nieuwe toetsinschrijving maken en opsturen (ook wel toetsgelegenheid genoemd) waarbij deze toetsgelegenheid wordt gekoppeld aan de originele toetsinschrijving van de heenweg (flow 1). Operatie en endpoint binnen OOAPI van de deze interactie “Student-pogingresultaat brengen” zijn:

PUT /associations/{associationId}

In **verzoek “2: Studentresultaat verwerken”** (zie Figuur 3.5.2) wordt de betreffende inschrijving geïdentificeerd door een nieuwe **associationId** als padparameter (zie {associationId}) bij de eerste aanlevering.

Vervolgaanleveringen kunnen middels de PUT (of eventueel PATCH) worden uitgevoerd. In het verzoekbericht zelf worden de gegevens van het studentresultaat meegegeven in een object **Association**, inclusief verwijzing naar de originele toetsinschrijving van de heenweg (flow 1) middels gegevensveld **orgAssociationId** in het object **Association**. Let op, dit endpoint en operatie ligt in het koppelvak bij DR/SIS.

Het antwoordbericht 3 (zie Figuur 3.5.2) bevat bij succesvolle verwerking een positieve terugmelding.

Operatie en endpoint binnen OOAPI van de tweede interactie “Document ophalen” zijn:

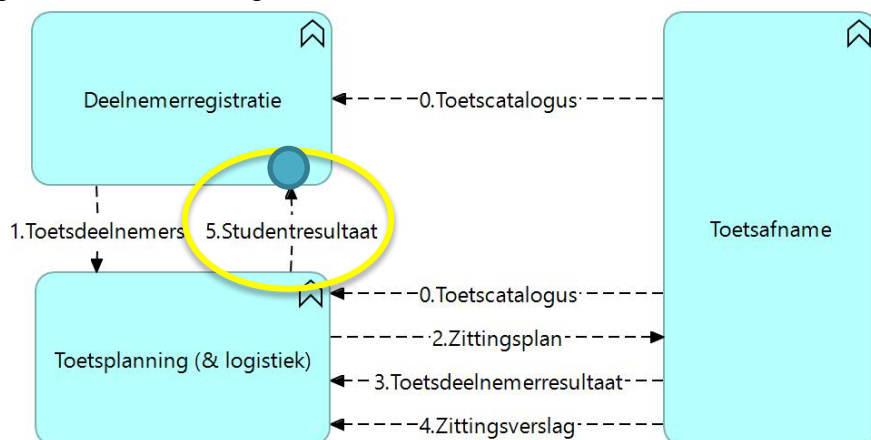
GET /documents/{documentId}

In verzoek “4: Document opvragen” (zie Figuur 3.5.2) wordt het specifieke document geïdentificeerd door **documentId** als padparameter.

Het **antwoordbericht “5: Document”** bevat bij succesvolle aanroep het gevraagde document in een **bestand**. Let op, dit endpoint en operatie ligt in het koppelvak bij TPL.

3.5.3 Koppelvlak bij Deelnemerregistratie

Het koppelvlak bij Deelnemerregistratie of SIS (DR/SIS) voor informatiestroom Studentresultaat is in onderstaand figuur met blauwe cirkel gemarkeerd.



Figuur 3.5.3: Koppelvlak bij Deelnemerregistratie voor Studentresultaat

Operatie & Endpoint	Verzoekbericht	Antwoordbericht
PATCH /associations/{associationId}	Padparameter: associationId=uuid Inhoud Studentresultaat (Association), zie Figuur 3.5.4 en 3.5.5	Statuscode
PUT /associations/{associationId}	Padparameter: associationId=uuid Inhoud Studentresultaat (Association), zie Figuur 3.5.6	Statuscode

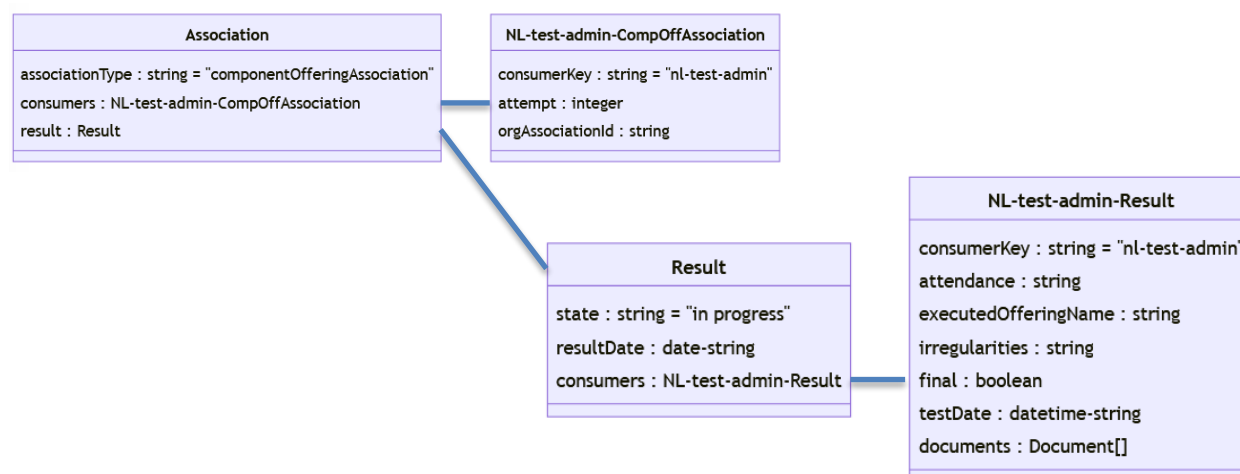
Tabel 3.5.1: Endpoints en operaties in koppelvlak bij Deelnemerregistratie voor Studentresultaat

De aanvrager (verzoeker/initiator) van alle endpoint-operaties in bovenstaande tabel is TPL.

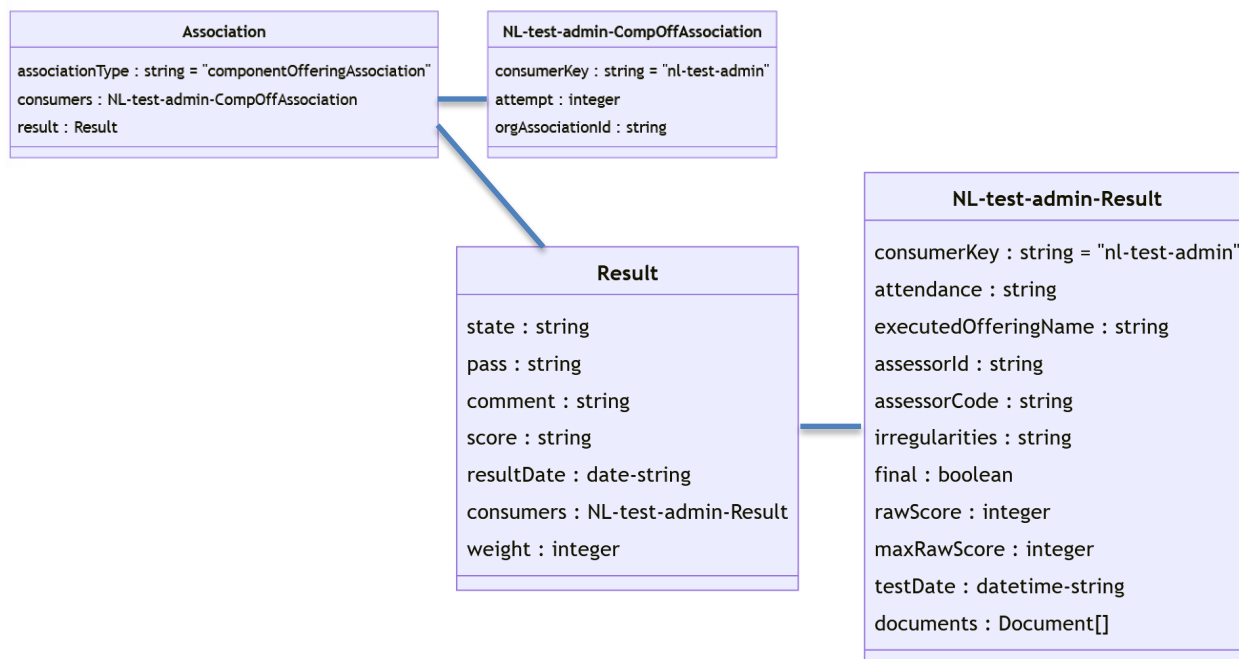
Het verzoekbericht van **PATCH /associations/{associationId}** bevat hier de gegevens m.b.t. de aanwezigheid en eventueel het studentresultaat van een deelname aan de zitting in object **Association**.

In het geval van aanwezigheidsinfo zal binnen de toetsinschrijving (object Association) de gegevensvelden **orgAssociationId** en **attempt** en binnen het resultaat (object Result) de gegevensvelden **attendance**, **executedOfferingName**, **final** en **testDate** zeker worden gebruikt, eventueel met gegevens irregularities en bijgaande documenten (zie onderstaand Figuur 3.5.4).

Aanvullende eis: In het verzoekbericht zijn in deze context (5. Studentresultaat) gegevens **result** (object **Result**), **orgAssociationId** en **attempt** (via **consumers**) en **associationType** binnen object **Association** verplicht; alle overige gegevens binnen Association zijn overbodig want deze zullen door DR/SIS worden genegeerd (zie onderstaand Figuur 3.5.4).



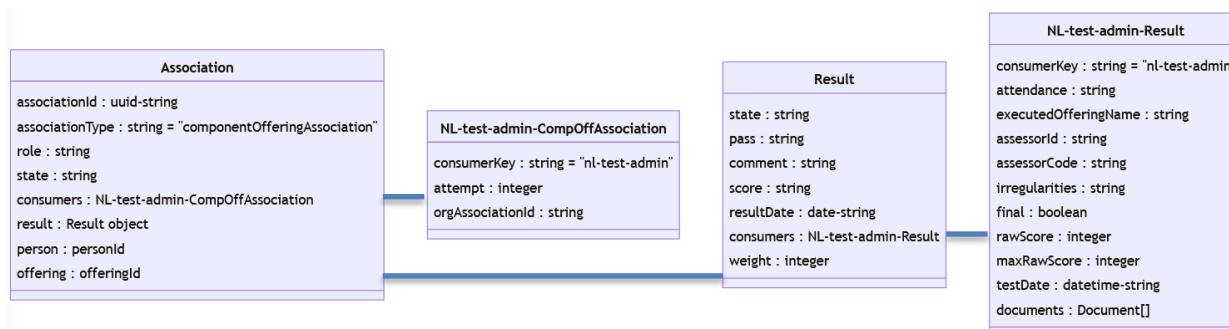
Figuur 3.5.4: Association gegevensmodel t.b.v. aanwezigheidsinfo in 5. Studentresultaat



Figuur 3.5.5: Association gegevensmodel t.b.v. scoreresultaat in 5. Studentresultaat

Het verzoekbericht van **PUT /associations/{associationId}** bevat hier de gegevens m.b.t. een nieuwe toetsinschrijving in **object Association** van het type **ComponentOfferingAssociation** (Toetsinschrijving) **inclusief associationId** om het resultaat van een poging bij een bestaande toetsinschrijving terug te geven. In deze situatie (Informatiestroom 5) wordt **gegeven result binnen object Association** gebruikt om het resultaat van de poging over te dragen en gegevens attempt voor het pogingnummer en orgAssociationId voor verwijzing naar originele toetsinschrijving (zie onderstaand Figuur 3.5.6).

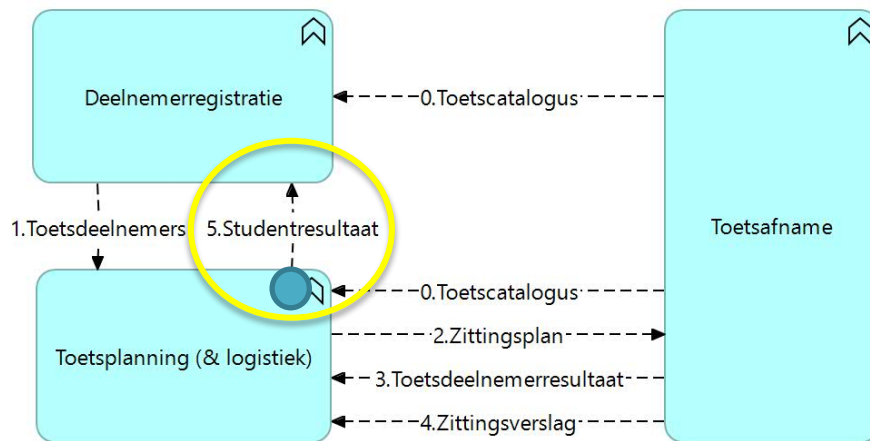
Aanvullende eis: Deze endpoint operatie PUT /associations/{associationId} wordt alleen gebruikt bij het aanleveren van het toetsresultaat van een extra poging door nieuwe toetsinschrijving; toetsresultaten bij bestaande toetsinschrijvingen worden middels PATCH aangeleverd.



Figuur 3.5.6: ComponentOfferingAssociation (Toetsinschrijving) gegevensmodel in 5. Studentresultaat

3.5.4 Koppelvlak bij Toetsplanning

Het koppelvlak bij Toetsplanning (TPL) voor informatiestroom 5. Studentresultaat is in onderstaand figuur met blauwe cirkel gemarkeerd.



Figuur 3.5.7: Koppelvlak bij Toetsplanning voor Studentresultaat

Operatie & Endpoint	Verzoekbericht	Antwoordbericht
GET /documents/{documentId}	Padparameter: documentId=uuid	Bijlage Bestand

Tabel 3.5.2: Endpoints en operaties in koppelvlak bij Toetsplanning voor Studentresultaat

De aanvrager (verzoeker/initiator) van de endpoint-operatie in bovenstaande tabel is DR/SIS.

4 Gegevensmodel

De uitwerking van het gegevensmodel in dit hoofdstuk beperkt zich tot de gegevensobjecten van de gegevensberichten in de informatiestromen in voorgaand hoofdstuk 3. Hierbij wordt uitgegaan van de informatieobjecten in de informatiestromen. Vanuit deze informatieobjecten wordt de relatie gelegd naar de gegevensobjecten in het gegevensmodel.

In voorgaande hoofdstuk 3 zijn de volgende informatieobjecten genoemd, met per informatieobject tussen haakjes het betreffende Engelstalige gegevensobject uit OOAPI:

- **Toets** (Component) en **Organisatie** (Organization) in Toetscatalogus (informatiestroom 0)
- **Persoon** (Person), **Toetsinschrijving** (ComponentOfferingAssociation), **Planbare toets** (ComponentOffering), **Deelinschrijving** (CourseOfferingAssociation), **Opleidingsdeel** (CourseOffering), **Opleidingsinschrijving** (ProgramOfferingAssociation), **Opleiding** (ProgramOffering), **Opleidingsprogramma** (Program), **Organisatie** (Organization), **Toetsgroep** (Group) en **Groepslidmaatschap** (MembershipItem) in Toetsdeelnemers (informatiestroom 1).
- **Persoon** (Person), **Toetsdeelname** (ComponentOfferingAssociation), **Zitting** (ComponentOffering) en **Organisatie** (Organization) in Zittingsplan (informatiestroom 2).
- **Toetsdeelname** (ComponentOfferingAssociation) met **Toetsresultaat** (Result) en **Document** (Document) in Toetsdeelnemerresultaat (informatiestroom 3).
- **Zitting** (ComponentOffering) met **Document** (Document) in Zittingsverslag (informatiestroom 4).
- **Toetsinschrijving** (ComponentOfferingAssociation) met **Toetsresultaat** (Result), **Pogingresultaat** (ComponentOfferingAssociation) en **Document** (Document) in Studentresultaat (informatiestroom 5).

Leeswijzer

In de tabellen die de gegevensmodellen beschrijven komen de volgende coderingen in kolom "Type" voor:

Code	Typenaam	Betekenis voor gegevenselement
Boolean	true/false	Gegevensveld met als waarde true of false (ja of nee).
Datum	date (string)	Gegevensveld met als waarde een datum volgens ISO 8601 Dates , d.w.z. volgens patroon CCYY-MM-DD (bijv. 2023-07-16).
Datumtijd	date-time (string)	Gegevensveld met als waarde een datum en tijdstip volgens ISO 8601 , d.w.z. volgens patroon CCYY-MM-DDThh:mm:ssTZD (bijv. 2023-07-16T19:20:30+01:00).
Document	object	De gegevensgroep Document met de velden documentId, documentType en documentName (zie elders in deze paragraaf).
Enum	enum (string)	Gegevensveld met waarde uit opsomming van waarden (waardelijst / enumeratie).
Getal	integer	Gegevensveld met als waarde een geheel getal (zoals voor aantal).
Groep	object	Gegevensgroep met deelelementen (groepen en/of velden) als onderdeel van Groep of Object.
IDkey	ID-key(string)	Gegevensveld met tekstwaarde voor identificatie van een gegevensentiteit.
IDref	ID-ref (string)	Gegevensveld met tekstwaarde voor verwijzing naar een gegevensentiteit.
Object	object	Gegevensgroep met deelelementen (groepen en/of velden) als zelfstandig onderdeel van het model.
TaalTekst	object	De gegevensgroep LanguageTypedString met de velden language en value (zie elders in deze paragraaf). Aanvullende eis Alle waarden voor language anders dan "nl-NL" moeten genegeerd.
Tekst	string	Een gegevensveld met als waarde een tekststring.
Tijdsduur	timeperiod (string)	Een gegevensveld met als waarde een tijdperiode volgens ISO 8601 Durations , d.w.z. volgens patroon PTmM (bijv. PT40M voor 40 minuten).
URI	uri (string)	Een gegevensveld met URI-waarde.
UUID	uuid (string)	Een gegevensveld met waarde volgens UUID (zie IETF RFC 4122).

Tabel 4.0.A: Leeswijzer: Codering in kolom Type

In de tabellen die de gegevensmodellen beschrijven komen de volgende coderingen in kolom “#” voor om te duiden of het gegeven verplicht of optioneel is en of het één of meer keren kan voorkomen:

Code	Betekenis
?	Optioneel enkelvoudig gegevenselement, komt nul of één keer voor.
1	Verplicht enkelvoudig gegevenselement, komt altijd voor exact één keer voor.
*	Optioneel meervoudig gegevenselement, komt nul of meer keren voor.
+	Verplicht meervoudig gegevenselement, komt één of meer keren voor.

Tabel 4.0.B: Leeswijzer: Codering in kolom #

Let op, in de definitie van de gegevensobjecten zijn genoemde gegevens binnen een object bij aanlevering (PUT of POST) of aflevering (GET) verplicht of optioneel zoals aangegeven in kolom #; de doorgestreepte gegevens (met 0 in kolom #) mogen worden gebruikt maar zullen bij ontvangst worden genegeerd.

Bij aanlevering van een wijziging (PATCH) mogen ongewijzigde verplichte gegevens ontbreken, m.u.v. offeringType binnen Offering en associationType binnen Association. Het weghalen van een optioneel enkelvoudig gegeven kan door de waarde null mee te geven; deze waarde null is dus hiervoor in PATCH toegestaan, ook wanneer dit element niet nullable is (d.w.z. nullable=false). Het weghalen van een optioneel meervoudig gegeven (lijst of array) kan door een lege lijst mee te geven; de lege lijst is dus hiervoor ook toegestaan al is de lijst gedefinieerd als “non-empty” (d.w.z. minItems=1).

Algemene gegevensgroepen

De volgende gegevenselementen worden op diverse plaatsen in het gegevensmodel als gegevensgroep (gegevensblok) toegepast:

IdentifierEntry	LanguageTypedString
codeType : string	language : string
code : string	value : string

Figuur 4.0.D: Algemene gegevensgroepen in het gegevensmodel

Het gegevenselement **IdentifierEntry** is een OOAPI-object en gegevensgroep omvat de gegevens over identifier in de vorm van een code met codetyping. De deelelementen van IdentifierEntry zijn weergegeven in onderstaande tabel.

Gegeven	Type	#	Betekenis en waardebereik
codeType	Enum	1	Typering van de code zoals meegegeven in veld code. Waarde in bestaande enumeratie CodeType . Zijn o.a.: • componentCode = De code voor een component (toets) • offeringCode = De code voor een offering (zitting)
code	Tekst	1	Identificerende code van de gegevensentiteit.

Tabel 4.0.C: Gegevensdefinities van IdentifierEntry

Het gegevenselement **LanguageTypedString** is een OOAPI-object en gegevensgroep en omvat een talige tekst, inclusief taalaanduiding. De deelelementen van LanguageTypedString (zie ook TaalTekst) zijn weergegeven in onderstaande tabel.

Gegeven	Type	#	Betekenis en waardebereik
language	Enum	1	De taalaanduiding van de tekst in veld value. Waarde is altijd “nl-NL”.
value	Tekst	1	De tekst waarvoor de taalaanduiding geldt..

Tabel 4.0.D: Gegevensdefinities van LanguageTypedString

Specifieke gegevensobjecten

De specifieke objecten in dit gegevensmodel en de uit OOAPI afgeleide definitie/beschrijving (zie [hier](#)) zijn:

- Component (Toets): “A Component is a template for the most concrete learning activities the OOAPI describes. Examples include workgroups, lectures and tests.”, zie paragraaf 4.1

- **ComponentOffering** (Planbare toets / Zitting): “A ComponentOffering is the concrete offering of a Component in time. Persons can be associated with a ComponentOffering through a ComponentOfferingAssociation.”, zie paragraaf 4.2.
- **ComponentOfferingAssociation** (Toetsinschrijving/-deelname): “This offeringAssociation provides the information regarding the association between an offering of type component and a person (e.g. student or employee)”. Een belangrijk onderdeel van Association is Result: “Result of a unit of study participation, Study Activity or test”, zie paragraaf 4.3.
- **CourseOffering** (Opleidingsdeel): “A CourseOffering is the concrete offering of a Course in time. Persons can be associated with a CourseOffering through a CourseOfferingAssociation.”, zie paragraaf 4.4.
- **CourseOfferingAssociation** (Deelinschrijving): “This offeringAssociation provides the information regarding the association between an offering of type course and a person (e.g. student)”, zie paragraaf 4.5.
- **Document** (Document): “The information about a available document.”, zie paragraaf 4.6 (Nieuw object).
- **Group** (Toetsgroep): “A group is a group of students that is related to organization, persons and componentOffering.”, zie paragraaf 4.7.
- **MembershipItem** (Groepslidmaatschap): “A membershipitem is a group of membership data of a student that is member of a group.”, zie paragraaf 4.8 (Nieuw object).
- **Organization** (Organisatie): “The organization that is responsible for the execution and recognition of education.”, zie paragraaf 4.9.
- **Person** (Persoon): “A person that has a relationship with this institution”, zie paragraaf 4.10.
- **Program** (Opleidingsprogramma): “A Program describes a coherent collection of courses leading to a certain outcome. Programs can be repeatedly offered to learners through ProgramOfferings..”, zie paragraaf 4.11.
- **ProgramOffering** (Opleiding): “A ProgramOffering is the concrete offering of a Program in time. Persons can be associated with a ProgramOffering through a ProgramOfferingAssociation.”, zie paragraaf 4.12.
- **ProgramOfferingAssociation** (Opleidingsinschrijving): “This offeringAssociation provides the information regarding the association between an offering of type program and a person (e.g. student)”, zie paragraaf 4.13.

De BIV-classificatie (Beschikbaarheid, Integriteit en Vertrouwelijkheid) van deze gegevensobjecten zijn als volgt ingeschat (zie [onderstaande](#) tabel):

Gegevensobject	Beschikbaarheid	Integriteit	Vertrouwelijkheid
Association	Medium/Low	Medium/High	High/Medium
Result as part of Association	Low	High	Medium
Component	Medium	High	Low
Document	Low	High	Medium
Group	Low	Medium	Low
MembershipItem	Low	Medium	Medium
Offering (ProgramOffering)	Medium	High	Low
Offering (CourseOffering)	Medium	High	Publicly available
Offering (ComponentOffering)	Medium	Medium	Low
Organization	Low	Low	Low
Person	Medium	High	High
Program	Medium	High	Publicly available

Tabel 4.0.E: BIV-classificatie van de gegevensobjecten

De Eigenaar (aanmaken en vernietigen), Bewerker (verwerken en/of mutaties aanbrengen) en Afnemer (lezen en/of opslaan) van deze gegevensobjecten zijn als volgt (zie onderstaande tabel).

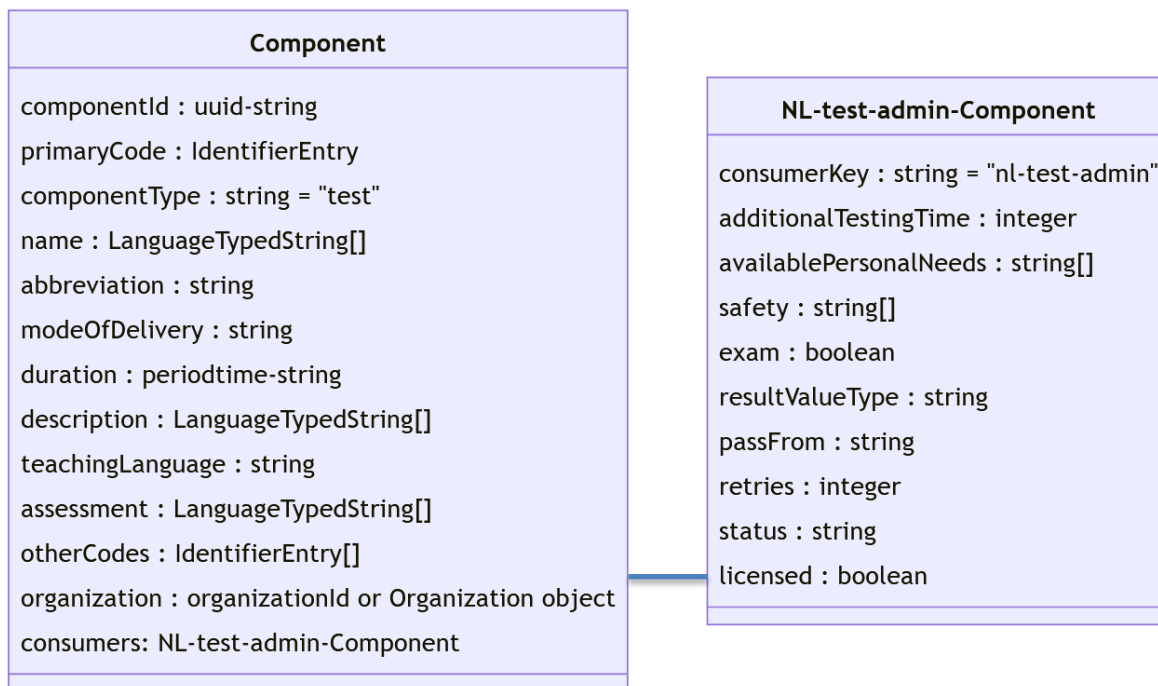
Gegevensobject	Eigenaar	Bewerker	Afnemer
Component (Toets)	Toetsafname	n.v.t.	Deelnemerregistratie, Toetsplanning
ComponentOffering (Planbare toets)	Deelnemerregistratie	n.v.t.	Toetsplanning
ComponentOffering (Zitting)	Toetsplanning	Toetsafname (Offering.consumers)	Toetsafname
CourseOffering (Opleidingsdeel)	Deelnemerregistratie	n.v.t.	Toetsplanning
CourseOfferingAssociation (Deelinschrijving)	Deelnemerregistratie	n.v.t.	Toetsplanning

ComponentOfferingAssociation (Toetsinschrijving)	Deelnemerregistratie	Toetsplanning (Association.result)	Toetsplanning
ComponentOfferingAssociation (Deelname)	Toetsplanning	Toetsafname (Association.result)	Toetsafname
Document (Document)	Toetsafname	n.v.t.	Toetsplanning, Deelnemerreg.
Group (Toetsgroep)	Deelnemerregistratie	n.v.t.	Toetsplanning
MembershipItem (Groepslidmaatschap)	Deelnemerregistratie	n.v.t.	Toetsplanning
Organization (Organisatie)	Toetsafname (in Component) Deelnemerregistratie (elders)	n.v.t. n.v.t.	Toetsplanning, Deelnemerreg. Toetsplanning, Toetsafname
Person (Persoon)	Deelnemerregistratie	n.v.t.	Toetsplanning, Toetsafname
Program (Opleidingsprogramma)	Deelnemerregistratie	n.v.t.	Toetsplanning
ProgramOffering (Opleiding)	Deelnemerregistratie	n.v.t.	Toetsplanning
ProgramOfferingAssociation (Opleidingsinschrijving)	Deelnemerregistratie	n.v.t.	Toetsplanning

Tabel 4.0.F: Eigenaar, Bewerker en Afnemer van de gegevensobjecten

Hierbij is de Eigenaar de bron van de gegevens. De Eigenaar maakt een gegevensobject aan en verwijdert dit object indien nodig. De eigenaar is dus in staat om alle gegevenswaarden van attributen te zetten (in te stellen) en te wijzigen. Eventueel worden wijzigingen voor een specifiek deel van de gegevens binnen het gegevensobject aangedragen door de Bewerker; de Eigenaar bepaalt wat het doet met de aangedragen gegevenswijziging: toevoegen, overschrijven of negeren. De Afnemer neemt de gegevens van het gegevensobject af voor het uitvoeren van de eigen functie(s).

4.1 Component (Toets) object



Figuur 4.1.A: Component gegevensmodel

Het gegevensobject Component (Toets) is een [OOAPI-object en -resource van het type Component](#) en omvat de gegevens m.b.t. een toets. De gegevenselementen van Component voor deze uitwisselingen zijn weergegeven in onderstaande tabel.

Gegeven	Type	#	Betekenis en waardebereik
componentId	IDkey	1	Identificer van de Component (Toets) in de vorm van een UUID, uniek in deze uitwisselingen. De identifier van deze toets of dit examen wordt gemaakt door Toetsafname.
primaryCode	IdentifierEntry	1	Gevegensgroep (object IdentifierEntry) om getypeerde primaire code van toets/examen mee te geven. Dit is de code die door de toetsleverancier als unieke, vaak menselijk leesbare aanduiding wordt gebruikt. Wanneer geen primaire code beschikbaar is voor deze toets/examen dan kan de waarde van componentId worden gebruikt.
• codeType	Enum	1	Typering van de code/identifier in gegevensveld 'code'. Aanvullende eis Gebruik hier altijd de waarde "componentCode" om aan te geven dat dit de identifier volgens de toetsleverancier is.
• code	Tekst	1	Unieke, leesbare aanduiding van de code/identifier volgens typering in gegevensveld 'codeType'.
componentType	Enum	1	Typering van toets/examen. Aanvullende eis Waarde is altijd "test" (voor tentamen, toets of examen). (niet gebruikt: lecture, practical, tutorial, consultation, project, workshop, excursion, independent study, external, skills training)
name	TaalTekst	*	Naam van toets/examen zoals de toetsleverancier heeft bepaald. Toelichting: gezien de duiding en herkenbaarheid wordt aanbevolen deze naam niet leeg te laten.
abbreviation	Tekst	1	Afkorting van toets/examen. Waarde heeft maximale lengte van 256 karakters.
modeOfDelivery	Enum	?	Aanduiding van aflevervormen/-wijzen van toets/examen. Aanvullende eis De toegestane waarde uit bestaande enumeratie zijn: • distance-learning = online, op afstand (kan ook thuis zijn) • online = online, op een bepaalde plek • situated= offline, op een bepaalde plek (kan ook ergens in een bedrijf zijn) (niet gebruikte waarden zijn: "on campus" en "hybrid")
duration	Tijdsduur	?	Duur van toets/examen. Waarde in de vorm van ISO 8601.
description	TaalTekst	*	Omschrijving van toets/examen zoals de toetsleverancier heeft bepaald.
teachingLanguage	Enum	1	Voertaal waarin toets/examen wordt afgenomen. Waarde is 3-letter code volgens ISO 639-2, zoals "nld" voor Nederlands. Aanvullende eis Gegeven niet gebruikt maar verplicht; waarde is altijd "nld".
learningOutcomes	TaalTekst	0	
enrollment	TaalTekst	0	
resources	Tekst	0	
assessment	TaalTekst	?	Beschrijving van de wijze waarop wordt geëxamineerd.
addresses	Groep	0	
otherCodes	IdentifierEntry	*	Lijst van objecten IdentifierEntry om getypeerde andere codes van toets/examen mee te geven, naast de primaire code in primaryCode.
• codeType	Enum	1	Typering van de code/identifier in gegevensveld 'code'.

			Aanvullende eis Gebruik hier in veld 'codeType' eventueel de waarden "testSystem" of "testProvider" om aan te geven dat dit de identifier van het systeem resp. de maker van de toets/examen typeert. <i>Let op, zolang deze waarden nog niet onderdeel zijn van de OOAPI-specificatie zouden deze waarden vooraf moeten worden gegaan door "x-", dus bijvoorbeeld "x-testSystem".</i>
• code	Tekst	1	Unieke, leesbare code/identifier volgens typering in gegevensveld 'codeType'.
course	IDref/Object	0	
organization	IDref/Object	?	De organisatie (toetsleverancier) waarop de Component (Toets) betrekking heeft middels verwijzing (organizationId) of specificatie (Organization object, zie paragraaf 4.9).
consumers	Groep	1	Aanvullende gegevens over Component (Toets) in object NL-test-admin-Component. Verplicht voor iedere Toets. Aanvullende eis Attributen gebruiken zoals onderstaand gedefinieerde gegevensgroep.
• consumerKey	IDref	1	Unieke typering van de aanvullende gegevens. Hier altijd de waarde "nl-test-admin" gebruiken.
• additionalTestingTime	Getal	?	De extra tijd in minuten die kan worden toegestaan voor een individuele student met behoefte aan extra tijd. Aanvullende eis Als waarde groter of gelijk aan 10.000 dan is de extra tijd onbeperkt. <i>Toelichting Dit gegeven is hernoemd naar het betreffende gegeven van 1EdTech PnP (Personal needs and Preferences). Het gegeven is een inperking van de drie opties in PnP: extra tijd (in minuten), extra tijd (in percentage) of onbeperkt.</i>
• availablePersonalNeeds	Enum	*	Lijst van bij toetsafname toegestane persoonlijke hulpmiddelen. Waarde uit nieuwe enumeratie HulpmiddelType (zie Tabel 4.1.B).
• safety	Enum	*	Lijst van veiligheidsmaatregelen die zijn voorzien door de toetsaanbieder. Waarde uit nieuwe enumeratie (zie Tabel 4.1.C).
• exam	Boolean	?	Aanduiding of dit een toets of een examen betreft.
• resultValueType	Enum	?	Aanduiding van te verwachten resultaattype. Waarde uit bestaande enumeratie ResultValueType (zie Tabel 4.1.D). <i>Toelichting inventarisatie leerde dat naast de resultaattypen uit de bestaande enumeratie ook de volgende soorten in gebruik zijn: "0-10" en "0.0-10.0" en "OVG" voor Onvoldoende/Voldoende/Goed). Eventueel kan dit in de toekomst generiek worden gemaakt met minimumwaarde, maximumwaarde, stapgrootte en cesuur.</i>
• passFrom	Tekst	?	Resultaatwaarde vanaf welke (en hoger) de student geslaagd is.
• retries	Getal	?	Aantal pogingen dat kan worden toegestaan voor een individuele student.
• status	Enum	?	Status van toets/examen; geeft de mogelijkheid toets/examen terug te trekken. Waarde uit nieuwe enumeratie TestStatusType (zie Tabel 4.1.E).
• licensed	Boolean	?	Geeft aan of de opvragende organisatie (school of organisatorische eenheid binnen de school) een licentie heeft voor gebruik van Toets.
ext	Groep	0	

Tabel 4.1.A: Gevensdefinities van Toets (object Component)

Waarde	Betekenis
extraTime	Student krijgt extra tijd voor de toetsafname.
spoken	Student krijgt de toetsafname uitgesproken.
spell-checker-on-screen	Student krijgt bij digitale toetsafname de beschikking over een spellingscontrole.

Tabel 4.1.B: Enumeratie HulpmiddelType (nieuw).

Waarde	Betekenis
securedComputer	Speciale, beveiligde computer
fixedLocation	Afname op een specifieke plek
surveillance	Toezicht tijdens de afname

Tabel 4.1.C: Enumeratie SafetyType (nieuw).

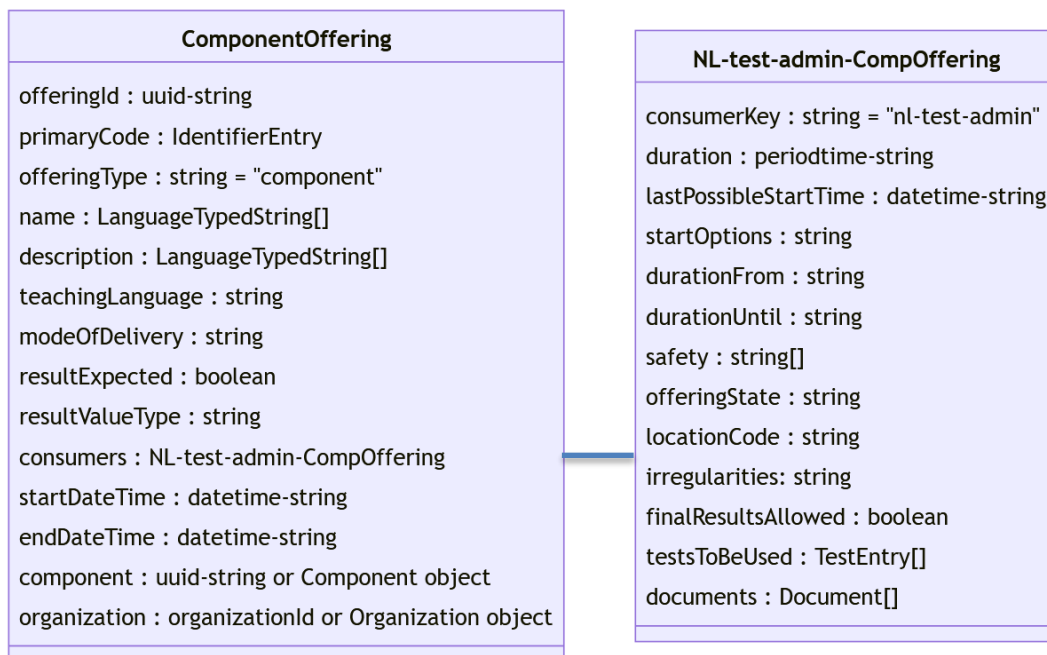
Waarde	Betekenis
pass-or-fail	Resultaatwaarde is conform slaging waarde uit waardelijst {passed, failed} voor Geslaagd respectievelijk Niet-geslaagd
insufficient-satisfactory-good	Resultaatwaarde is waarde uit waardelijst {insufficient, satisfactory, good} voor beoordeling Onvoldoende, Voldoende respectievelijk Goed, ook wel bekend onder OVG (nieuwe waarde)
US letter	Resultaatwaarde is een letter uit A t/m D en F voor fail (eventueel met + of -) zoals in VS wordt toegepast, zie hier
UK letter	Resultaatwaarde is een letter uit A t/m G en U voor ungraded/unclassified (eventueel met + of -) zoals in VK toegepast, zie hier
DE grade	Resultaatwaarde is waarde zoals in Duitsland wordt toegepast (nieuwe waarde)
0-100	Resultaatwaarde is geheel getal uit interval (0,100)
0-10	Resultaatwaarde is geheel getal uit interval (0,10)
0.0-10.0	Resultaatwaarde is decimaal getal uit interval (1.0,10.0) (nieuwe waarde)
referenceLevelRKTR	Resultaatwaarde is referentieniveau uit Referentiekader Taal en Rekenen uit waardelijst {1F, 1S, 2F, 2S, 3F, 3S, 4F, 4S} en additionele waarden zoals "Op weg naar 1F".
referenceLevelERK	Resultaatwaarde is referentieniveau uit Europees Referentiekader voor de talen uit waardelijst {A1, A2, B1, B2, C1, C2}.

Tabel 4.1.D: Enumeratie ResultValueType.

Waarde	Betekenis
active	Status van toets (examen) is actief
inactive	Status van toets (examen) is nog niet-actief (in ontwikkeling) of niet meer actief (teruggetrokken)

Tabel 4.1.E: Enumeratie TestStatusType (nieuw).

4.2 ComponentOffering (Planbare toets & Zitting) object



Figuur 4.2.A: ComponentOffering gegevensmodel

Het gegevensobject ComponentOffering is een [OOAPI-object en -resource Offering van het type component](#) en omvat de gegevens m.b.t. de Planbare toets of de Zitting (ofwel geplande toets). Deze Offering heeft een directe verwijzing naar het betreffende Component (Toets).

De gegevenselementen van Offering voor deze uitwisselingen zijn weergegeven in onderstaande tabel.

Gegeven	Type	#	Betekenis en waardebereik
offeringId	IDkey	1	Identificer van de ComponentOffering (Planbare toets of Zitting) in de vorm van een UUID, uniek in deze uitwisselingen. Deze identificer van Planbare toets wordt gemaakt door Deelnemerregistratie; deze identificer van Zitting wordt gemaakt door Toetsplanning (& logistiek).
primaryCode	IdentifierEntry	1	Gegevensgroep (object IdentifierEntry) om getypeerde primaire code van de offering mee te geven. Dit is voor de zitting de code die door de afnameleider als unieke, vaak menselijk leesbare aanduiding wordt gebruikt. Wanneer geen primaire code beschikbaar is voor deze offering dan kan de waarde van offeringId worden gebruikt. <i>Let op, deze primaire code hoeft niet uniek te zijn, de afnameleider moet 'm herkennen.</i>
• codeType	Enum	1	Typering van de code/identificer in gegevensveld 'code'. Aanvullende eis Gebruik in veld codeType altijd de waarde "offeringCode" om aan te geven dat dit de identificer van de offering is.
• code	Tekst	1	Unieke, leesbare code/identificer volgens typering in gegevensveld 'codeType'.
offeringType	Enum	1	Typering van de offering. Aanvullende eis Waarde is altijd "component" (niet gebruikte waarden zijn program en course).
academicSession	IdRef/Object	0	
name	TaalTekst	*	Naam van offering. Toelichting: gezien de duiding en herkenbaarheid wordt aanbevolen deze naam niet leeg te laten.
abbreviation	Tekst	0	
description	TaalTekst	+	Omschrijving van offering.
teachingLanguage	Enum	1	Voertaal waarin deze offering wordt afgenomen. Waarde is 3-letter code volgens ISO 639-2, zoals "nld" voor Nederlands. Aanvullende eis Gegeven niet gebruikt maar verplicht; waarde is altijd "nld".
modeOfDelivery	Enum	*	Aanduiding van aflevervormen van offering. Aanvullende eis Toegestane waarde uit bestaande enumeratie zijn: • distance-learning = online, op afstand (kan ook thuis zijn) • online = online, op een bepaalde plek • situated = offline, op een bepaalde plek (kan ook ergens in een bedrijf zijn) (niet gebruikte waarden: "on campus" en "hybrid")
maxNumberStudents	Tekst	0	
enrolledNumberStudents	Tekst	0	
pendingNumberStudents	Tekst	0	
minNumberStudents	Tekst	0	
resultExpected	Boolean	1	Geeft aan of er voor deze offering een resultaat wordt verwacht. Aanvullende eis Waarde is hier altijd true.

resultValueType	Enum	?	Het te verwachten resultaattype. Waarde uit enumeratie ResultValueType (zie Tabel 4.1.D). Aanvullende eis: alleen gebruikt voor <i>ComponentOffering</i> als Planbare toets in flow 1. Toelichting: Het attribuut resultValueType is binnen Toets (Component) gedefinieerd. Het is daarom niet nodig dit attribuut in ComponentOffering als Zitting in flow 2 te gebruiken.
link	Tekst	0	
otherCodes	IdentifierEntry	0	
consumers	Groep	1	Aanvullende gegevens in object NL-test-admin-CompOffering. Verplicht voor iedere offering. Aanvullende eis Attributen gebruiken zoals onderstaande gegevensgroep.
• consumerKey	IDref	1	Typering van de aanvullende gegevens. Hier altijd de waarde “nl-test-admin” gebruiken.
• duration	Tijdsduur	?	Duur van de zitting. Waarde geeft het aantal minuten aan volgens ISO 8601 durations, bijvoorbeeld “PT40M” voor 40 minuten.
• lastPossibleStartDateTime	Datumtijd	?	Het laatst mogelijke moment van starten van de zitting. Waarde volgens ISO 8601 date-time, bijvoorbeeld “2020-12-15T08:50:00.000Z”.
• startOptions	Enum	?	Indicatie van het soort van starten van de zitting dat is toegestaan. Waarde uit nieuwe enumeratie: ▪ individualStart = Student kan individueel starten in toegestane tijdsperiode ▪ triggeredStart = Student kan starten na vrijgave door afnameleider
• durationFrom	Enum	?	Indicatie om het startmoment van de afname te bepalen. Waarde uit nieuwe enumeratie: ▪ startDateTime = Afnameduur wordt gemeten vanaf startmoment van afname in ‘startDateTime’ ▪ individualStartDateTime = De afnameduur wordt gemeten vanaf het individuele startmoment ▪ triggeredStartDateTime = De afnameduur wordt gemeten vanaf het moment van vrijgave
• durationUntil	Enum	?	Indicatie om de lengte van de afname te bepalen. Waarde uit nieuwe enumeratie: ▪ testDuration = De afnameduur is conform de duur in ‘duration’ ▪ endDateTime = De afname wordt beëindigd zodra eindmoment in ‘endDateTime’ is bereikt
• safety	Enum	*	Lijst van bij toetsafname geldende veiligheidsmaatregelen. Waarde uit nieuwe enumeratie (zie Tabel 4.1.C): ▪ securedComputer = Speciale, beveiligde computer ▪ fixedLocation = Afname op een specifieke plek ▪ surveillance = Toezicht tijdens de afname
• offeringState	Enum	1	Status van de zitting; geeft de mogelijkheid zitting terug te trekken. Waarde uit nieuwe enumeratie: ▪ active = Status van de zitting (offering) is actief ▪ canceled = Status van het zitting (offering) is nog niet-actief (in ontwikkeling) of niet meer actief (teruggetrokken) Toelichting Deze eigenschap offeringState ontbreekt in Offering zelf en is daarom toegevoegd in deze consumers gegevensgroep.
• locationCode	Tekst	?	Aanduiding van locatie/ruimte/lokaal waar de toetsafname plaatsvindt.
• irregularities	Tekst	?	Tekstuele beschrijving van onregelmatigheden die tijdens de toetsafname hebben plaatsgevonden. Aanvullende eis: alleen gebruikt voor zittingsresultaat in flow 4.
• finalResultsAllowed	Boolean	?	Geeft aan of ook een vastgestelde resultaat mag worden aangeleverd; default op true. Door deze waarde op false te zetten kan worden aangegeven dat een vastgesteld resultaat niet is toegestaan.
• testsToBeUsed	Groep	*	Lijst van items van type TestEntry (is gegevensgroep met combinatie van testProvider en componentId) Aanvullende eis: alleen gebruikt voor Inschrijving in flow 1.
• documents	Document	*	Lijst van documenten (object Document) betreffende het resultaat van de zitting middels objecten Document, zie paragraaf 4.6). Aanvullende eis: alleen gebruikt voor zittingsresultaat in flow 4.
ext	Groep	0	
startDateTime	Datumtijd	1	Moment (datum en tijdstip) van start van toetsafname.
endDateTime	Datumtijd	1	Moment (datum en tijdstip) van einde van toetsafname.
enrollStartDate	Datum	0	
enrollEndDate	Datum	0	
resultWeight	Integer	0	
addresses	Groep	0	
priceInformation	Groep	0	
room	Groep	0	
component	IDref/Object	?	Toets/examen waarop deze offering (Planbare toets of Zitting) betrekking heeft middels verwijzing (componentId) of specificatie (Component object) Opmerking: wanneer geen endpoint beschikbaar is voor het ophalen van de informatie over het component dan is het belangrijk om het gegevensobject mee te geven, en niet de verwijzing.
courseOffering	IDref/Object	0	
organization	IDref/Object	?	Organisatie van offering middels verwijzing (organizationId) of specificatie (Organization object, zie paragraaf 4.9).

Tabel 4.2.A: Gegevensdefinities van ComponentOffering (Planbare toets of Zitting)

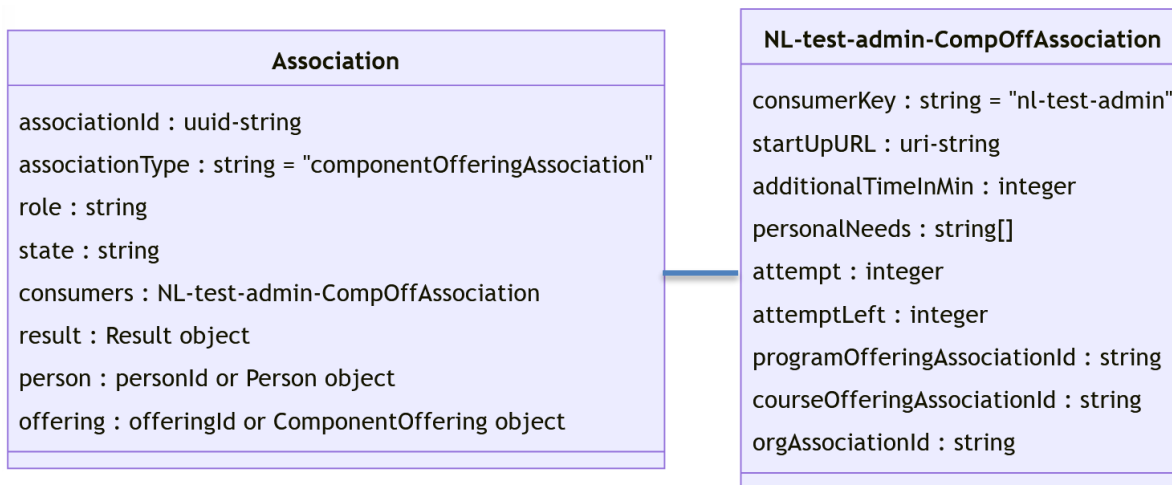
Werkingsregel Testmomenten

Om de samenhang tussen de verschillende gegevensvelden aan te geven is onderstaande tabel van scenario's "Testmomenten" gepresenteerd met gebruik van de betreffende gegevensvelden.

Scenario	startDateTime	entryDateTime	endDateTime	duration	startOptions	durationFrom	durationUntil
Testmoment starts at 9:00 and ends at 10:00. Candidates can start test at any moment during test moment	2022-11-15 09:00		2022-11-15 10:00	40 minutes (+ 10 minutes for extra time)	individualStart	individualStartDateTime	endDateTime
Testmoment starts at 9:00 and ends at 10:00. Candidates can start until 09:15	2022-11-15 09:00	2022-11-15 09:15	2022-11-15 10:00	40 minutes (+ 10 minutes for extra time)	individualStart	individualStartDateTime	endDateTime
Testmoment starts at 9:00 and ends at 10:00. Candidates can start until 09:30 and can always finish their test	2022-11-15 09:00	2022-11-15 09:30	2022-11-15 10:00	40 minutes (+ 10 minutes for extra time)	individualStart	individualStartDateTime	testDuration
Testmoment starts at 9:00. Candidates can start later. Testduration from start of testmoment	2022-11-15 09:00	2022-11-15 10:00		40 minutes (+ 10 minutes for extra time)	individualStart	startDateTime	
Testmoment starts at 9:00. Candidates can start when supervisor releases testmoment. Testduration is from releasement	2022-11-15 09:00		2022-11-15 10:00	40 minutes (+ 10 minutes for extra time)	triggeredStart	triggeredStartDateTime	testDuration
Practice can be started by candidates until one hour before end of the practice period	2022-11-15 08:00	2022-11-15 21:00	2022-11-15 22:00	40 minutes (+ 10 minutes for extra time)	individualStart	individualStartDateTime	testDuration

Tabel 4.2.C: Scenario's voor testmomenten.

4.3 ComponentOfferingAssociation (Toetsinschrijving & Deelname) object



Figuur 4.3.A: ComponentOfferingAssociation gegevensmodel

Dit gegevensobject ComponentOfferingAssociation is een [OOAPI-object en -resource Association van het type componentOfferingAssociation](#) en omvat de gegevens m.b.t. een inschrijving van de student of medewerker (afnameleider, surveillant of beoordelaar) aan de planbare toets (Toetsinschrijving) of deelname door een student of een medewerker aan de zitting (Deelname). Deze ComponentOfferingAssociation heeft een directe verwijzing naar de betreffende Offering (Zitting) en bevat een verwijzing naar (of gegevens over) de betreffende Person (Persoon). Dit object omvat tevens de resultaatgegevens van de deelname, dus het resultaat dat behaald is door betreffende persoon tijdens de zitting.

De gegevenselementen van ComponentOfferingAssociation zijn weergegeven in onderstaande tabel.

Gegeven	Type	#	Betekenis en waardebereik
associationId	IDkey	1	Identificer van ComponentOfferingAssociation (Toetsinschrijving of Deelname) in de vorm van een UUID, uniek in deze uitwisselingen.
associationType	Enum	1	Typing van inschrijving/deelname; waarde is hier altijd "componentOfferingAssociation".
role	Enum	1	Typing van rol van deelnemer in deze deelname.
state	Enum	1	Status van inschrijving/deelname.
remoteState	Enum	0	
consumers	Groep	1	Aanvullende gegevens over inschrijving/deelname in NL-test-admin-CompOffAssociation. Verplicht voor iedere studentdeelname.
• consumerKey	IDref	1	Aanvullende eis: Attributen gebruiken zoals onderstaand gedefinieerde gegevensgroep.
• startUpURL	URL	0	Typing van de aanvullende gegevens. Hier altijd de waarde "nl-test-admin" gebruiken.
• additionalTimeInMin	Getal	?	Dit attribuut wordt sinds versie 0.9.3 van deze spec niet meer gebruikt.
• personalNeeds	Enum	*	Extra tijd voor betreffende student in toetsafname van betreffende toetsinschrijving/deelname; waarde is getal dat het aantal extra minuten aangeeft.
• attempt	Getal	?	Aanvullende eis: Als waarde groter of gelijk aan 10.000 dan is de extra tijd onbeperkt.
• attemptLeft	Getal	?	Lijst van bij toetsafname benodigde hulpmiddelen voor de bij deelname betrokken student. Waarde uit nieuwe enumeratie HulpmiddelType (zie Tabel 4.1.B).
• programOffering-AssociationId	Tekst	?	Pogingnummer van de poging die gaat worden ingepland (flow 1) of van de poging die is uitgevoerd (flow 5).
• courseOffering-AssociationId	Tekst	?	Aanvullende eis: alleen gebruikt voor Inschrijving in flow 1 en flow 5.
• orgAssociationId	Tekst	?	Aantal pogingen over.
• result	Groep	?	Aanvullende eis: alleen gebruikt voor Inschrijving in flow 1.
• person	IDref/Object	1	Verwijzing naar de inschrijving van de opleiding waartoe deze planbare toets behoort.
			Aanvullende eis: alleen gebruikt voor Inschrijving in flow 1.
			Verwijzing naar de inschrijving van het opleidingsdeel waartoe deze planbare toets behoort.
			Aanvullende eis: alleen gebruikt voor Inschrijving in flow 1.
			Verwijzing naar de originele planbare toets waartoe deze toetsinschrijving behoort. Dit gegeven is bedoeld om o.b.v. 1 planbare toets meerdere resultaten te kunnen terugsturen in flow 5.
			Aanvullende eis: alleen gebruikt voor overdracht van studentresultaat in flow 5.
ext	Groep	0	
result	Groep	?	Resultaatgegevens over inschrijving/deelname aan de toets, zoals aanwezigheid en behaalde score, zie Result (Resultaat) object verderop in deze paragraaf (zie Figuur 4.1.B en Tabel 4.1.E).
person	IDref/Object	1	Aanvullende eis: Dit gegeven moet worden gebruikt voor toetsresultaat in flow 3 en flow 5.
			Persoon die als deelnemer betrokken is bij inschrijving/deelname: óf de ID van de persoon (personId) óf de gegevensgroep van type Person over de persoon, zie Person (Persoon) object .
			Aanvullende eis: Hier de personId gebruiken (en niet het gegevensobject), tenzij specifiek om gevraagd.

offering	IDref/Object	1	Offering (Planbare toets of Zitting) van deze inschrijving/deelname middels verwijzing (offeringId) of specificatie (object ComponentOffering).
issuer	Groep	0	Aanvullende eis Hier de offeringId gebruiken (en niet het gegevensobject), tenzij specifiek om gevraagd.

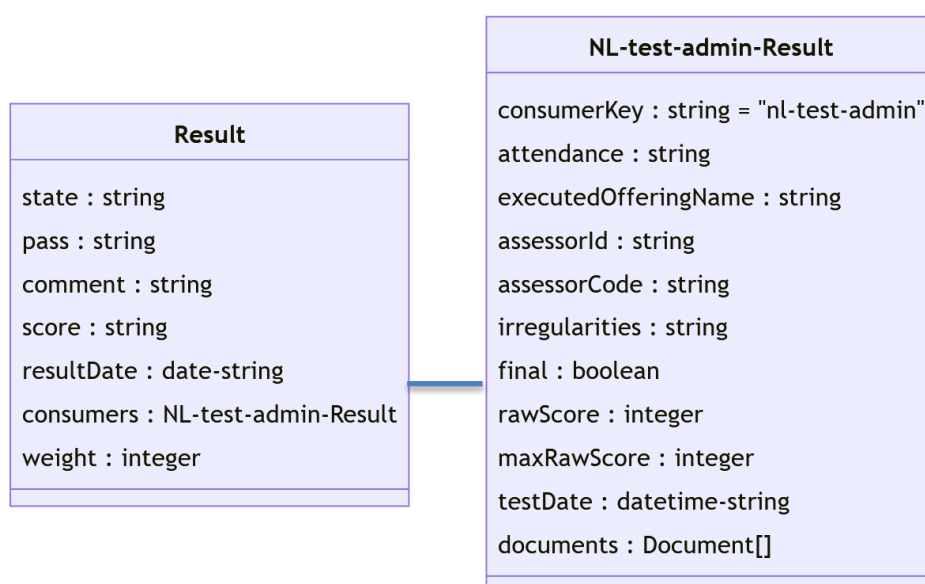
Tabel 4.3.A: Gegevensdefinities van ComponentOfferingAssociation (Toetsinschrijving & Deelname)

Waarde	Betekenis
student	Student
coordinator	Coördinator / afnameleider
invigilator	Surveillant (nieuwe waarde)
assessor	Beoordelaar (nieuwe waarde)
lecturer	
teaching assistant	
guest	Iedere rol anders dan bovenstaande rollen (alleen voor medewerker)

Tabel 4.3.B: Enumeratie RoleType.

Waarde	Betekenis
associated	Status van de inschrijving/deelname is actief (gekoppeld).
canceled	Status van de inschrijving/deelname is niet meer actief omdat het is teruggetrokken/geannuleerd.
pending	
denied	
queued	
finished	Status van de inschrijving/deelname is niet meer actief omdat het is beëindigd/afgesloten.

Tabel 4.3.C: Enumeratie AssociationStateType.



Figuur 4.3.B: Result gegevensmodel

Het gegevensobject Result is een OOAPI-object van het type **ComponentResult** en omvat de gegevens m.b.t. een resultaat van de betreffende inschrijving of deelname. In deze context betreft dit het resultaat van de Inschrijving van een student (Person) bij een Planbare toets (ComponentOffering) of het resultaat van de Deelname (Association) van een student (Person) bij een bepaalde Zitting (ComponentOffering). De gegevenselementen van Result (Resultaat) zijn weergegeven in onderstaande tabel.

Gegeven	Type	#	Betekenis en waardebereik
state	Enum	1	Status van resultaat. Waarden in enumeratie zijn "in progress", "postponed", "completed", "queued". Aanvullende eis Het resultaat kan worden verstuurd met status "in progress" (of "postponed" en "queued"), maar moet worden verstuurd wanneer de status wijzigt in "completed". Het resultaat van de status "completed" kan worden vervangen door een ander resultaat (bijvoorbeeld wanneer een opgave is vervallen), gewoonlijk ook in status "completed".
pass	Enum	?	Status van slaging van resultaat. Waarden in bestaande enumeratie zijn: • unknown = Onbekend • passed = Geslaagd • failed = Niet geslaagd
comment	Tekst	?	Commentaar op of opmerking over resultaat.
score	Tekst	?	Behaalde score (resultaatwaarde) voor betreffende deelname aan de toets; waarde moet voldoen aan het resultaattype in de offering (zitting). <i>Toelichting</i> aanvankelijk was er enige onduidelijkheid over de betekenis: score of resultaat; door de toevoeging van rawScore en maxRawScore in consumers is dit opgehelderd. Verder is het op deze wijze lastig om verschillende scores/resultaten over dezelfde toetsafname over te dragen, bijvoorbeeld het behaalde cijfer en de percentielscore. Een andere moeilijkheid is wanneer er cijfers/resultaten zijn voor toetsonderdelen of domeinen.

resultDate	Datum	1	Datum waarop het behaalde resultaat is bepaald/afgerond/gepubliceerd.
consumers	Groep	1	Aanvullende gegevens over resultaat in object NL-test-admin-Result. Verplicht voor ieder resultaat. Aanvullende eis Attribueet consumers gebruiken hoewel dit gegeven binnen Result ontbreekt in OOAPI! Aanvullende gegevens over het resultaat. Verplicht voor ieder resultaat. Aanvullende eis Attributen gebruiken zoals onderstaand gedefinieerde gegevensgroep.
• consumerKey	IDref	1	Unieke typering van de aanvullende gegevens. Hier altijd de waarde "nl-test-admin" gebruiken.
• attendance	Enum	1	Indicatie over de aanwezigheid van de student bij de afname. Waarde uit nieuwe enumeratie AttendanceType (zie Tabel 4.3.E).
• executed- OfferingName	Tekst	1	Naam van de zitting waarbij het resultaat is behaald. Aanvullende eis: Verplicht te gebruiken in de overdacht van studentresultaat in flow 5.
• assessorId	IDref	?	Identificer van de persoon die de beoordeling heeft gedaan.
• assessorCode	Tekst	?	Code van de persoon die de beoordeling heeft gedaan.
• irregularities	Tekst	?	Tekstuele informatie over afname, zoals onregelmatigheden die zijn opgetreden.
• final	Boolean	?	Geeft aan of resultaat vastgesteld is door de examencommissie. <i>Toelichting</i> De waarde bij het ontbreken van dit gegeven is false (d.w.z. niet vastgesteld)! Het vaststellen door de examencommissie kan in Toetsafname plaatsvinden maar kan ook op het moment dat de toetsdeelnemerresultaten zijn overgedragen aan Toetsplanning, of zelfs verder naar Deelnemerregistratie.
• rawScore	Getal	?	Het aantal behaalde scorepunten door een persoon waarmee het resultaat kan worden berekend/bepaald. Deze ruwe score geeft aanvullend inzicht in de prestatie van de persoon. De maximum ruwe score (zie maxRawScore) geeft context aan de ruwe score.
• maxRawScore	Getal	?	Het maximum aantal scorepunten dat een persoon kan behalen. Deze maximum ruwe score geeft context aan de behaalde ruwe score (zie rawScore).
• testDate	Datumtijd	?	Het moment (datum en tijdstip) wanneer de toetsafname heeft plaatsgevonden. Hierbij wordt de slotdag/slotmoment van de toetsafname genomen of het moment waarop het werk is ingeleverd/afgerond. Waarde volgens ISO 8601 date-time, bijvoorbeeld "2020-12-15T08:50:00.000Z". <i>Toelichting</i> Dit gegeven is vooral belangrijk in de informatiestroom 5 zodat het werkelijke moment van afname bij het resultaat kan worden vastgelegd in Deelnemerregistratie.
• documents	Document	*	Lijst van documenten betreffende het resultaat van de individuele deelname middels objecten Document, zie paragraaf 4.6).
ext	Tekst	0	
weight	Getal	1	Gewicht als totaal van deze toets in het Opleidingsdeel (Course). Waarde in interval 0 t/m 100. <i>Toelichting</i> Dit gegeven wordt niet gebruikt maar is verplicht, daarom altijd waarde 100!

Tabel 4.3.D: Gegevensdefinities van Result (Resultaat)

Waarde	Betekenis
notKnown	Informatie over aanwezigheid is niet bekend, zoals wanneer de laatste mogelijke starttijd van de afname nog niet is gepasseerd of wanneer er nog geen aanwezigheidsgegevens zijn ingevoerd of geregistreerd.
notPresent	Niet aanwezig bij afname.
notStarted	Aanwezig maar niet gestart
notFinished	Aanwezig en gestart maar niet afgerond/ingeleverd.
present	Aanwezig en afgerond/ingeleverd.

Tabel 4.3.E: Enumeratie AttendanceType (nieuw).

4.4 CourseOffering (Opleidingsdeel) object

CourseOffering
offeringId : uuid-string primaryCode : IdentifierEntry offeringType : string = "course" name : LanguageTypedString[] description : LanguageTypedString[] teachingLanguage : string resultExpected : boolean startDate : date-string endDate : date-string programOffering : offeringId or ProgramOffering object organization : organizationId or Organization object

Figuur 4.4.A: CourseOffering gegevensmodel

Het gegevensobject CourseOffering (Opleidingsdeel) is een [OOAPI-object en -resource Offering van het type course](#) en omvat de gegevens m.b.t. een deel van de opleiding. De gegevenselementen van CourseOffering voor deze uitwisselingen zijn weergegeven in onderstaande tabel.

Gegeven	Type	#	Betekenis en waardebereik
offeringId	IDkey	1	Identificer van het CourseOffering (Opleidingsdeel) in de vorm van een UUID, uniek in deze uitwisselingen. Deze identifier van opleidingsdeel wordt gemaakt door Deelnemerregistratie.
primaryCode	IdentifierEntry	1	Gegevensgroep (object IdentifierEntry) om getypeerde primaire code van het opleidingsdeel mee te geven. Wanneer geen primaire code beschikbaar is voor dit opleidingsdeel dan kan de waarde van offeringId worden gebruikt.
• codeType	Enum	1	Typering van de code/identifier in gegevensveld 'code'. Aanvullende eis Gebruik in veld codeType altijd de waarde "offeringCode" om aan te geven dat dit de identifier van de CourseOffering (Opleidingsdeel) is.
• code	Tekst	1	Unieke, leesbare code/identifier volgens typering in gegevensveld 'codeType'.
offeringType	Enum	1	Typering van de offering. Aanvullende eis Waarde is altijd "course" (niet gebruikte waarden zijn program en component).
academicSession	IdRef/Object	0	
name	TaalTekst	+	Naam van opleidingsdeel. <i>Toelichting: gezien de duiding en herkenbaarheid wordt aanbevolen deze naam niet leeg te laten.</i>
abbreviation	Tekst	0	
description	TaalTekst	+	Omschrijving van opleidingsdeel.
teachingLanguage	Enum	1	Voertaal waarin dit opleidingsdeel wordt gevolgd. Waarde is 3-letter code volgens ISO 639-2, zoals "nld" voor Nederlands. Aanvullende eis Gegeven niet gebruikt maar verplicht; waarde is altijd "nld".
modeOfDelivery	Enum	2	
maxNumberStudents	Getal	0	
enrolledNumberStudents	Getal	0	
pendingNumberStudents	Getal	0	
minNumberStudents	Getal	0	
resultExpected	Boolean	1	Geeft aan of er voor dit opleidingsdeel resultaat wordt verwacht. Aanvullende eis Waarde is hier altijd false.
resultValueType	Enum	0	
link	Tekst	0	
etherCodes	IdentifierEntry	0	
consumers	Groep	0	
ext	Groep	0	
startDate	Datum	1	Startdatum van opleidingsdeel.
endDate	Datum	1	Einddatum van opleidingsdeel.
enrollStartDate	Datum	0	
enrollEndDate	Datum	0	
flexibleEntryPeriodStart	Datum	0	
flexibleEntryPeriodEnd	Datum	0	
addresses	Groep	0	

priceInformation	Greep	0	
course	IDref/Object	0	
programOffering	IDref/Object	?	ProgramOffering (Opleiding) waarop deze CourseOffering (Opleidingsdeel) betrekking heeft middels verwijzing (offeringId) of specificatie (Offering object).
organization	IDref/Object	?	Organisatie van deze CourseOffering (Opleidingsdeel) middels verwijzing (organizationId) of specificatie (Organization object, zie paragraaf 4.9). Opmerking: wanneer geen endpoint beschikbaar is voor het ophalen van de informatie over de organisatie dan is het belangrijk om het gegevensobject mee te geven, en niet de verwijzing.

Tabel 4.4.A: Gegevensdefinities van CourseOffering (Opleidingsdeel)

4.5 CourseOfferingAssociation (Deelinschrijving) object

Association
associationId : uuid-string associationType : string = "courseOfferingAssociation" role : string state : string person : personId or Person object offering : offeringId or CourseOffering object

Figuur 4.5.A: CourseOfferingAssociation gegevensmodel

Dit gegevensobject CourseOfferingAssociation (Deelinschrijving) is een [OOAPI-object en -resource Association van het type courseOfferingAssociation](#) en omvat de gegevens m.b.t. een inschrijving voor het opleidingsdeel door een student. Deze Association bevat een verwijzing naar het betreffende object Person (persoon).

De gegevenselementen van CourseOfferingAssociation zijn weergegeven in onderstaande tabel.

Gegeven	Type	#	Betekenis en waardebereik
associationId	IDkey	1	Identifier van CourseOfferingAssociation (Deelinschrijving) in de vorm van een UUID, uniek in deze uitwisselingen. Deze identifier van deelinschrijving wordt gemaakt door Deelnemerregistratie.
associationType	Enum	1	Typering van inschrijving; waarde is hier altijd "courseOfferingAssociation" voor deelinschrijving.
role	Enum	1	Typering van rol van de persoon in deze deelinschrijving. Aanvullende eis Hier alleen waarde "student" gebruiken uit enumeratie RoleType (zie Tabel 4.3.B).
state	Enum	1	Status van deelinschrijving. Aanvullende eis Beperkte waarden uit enumeratie AssociationStateType (zie Tabel 4.3.C).
remoteState	Enum	0	
consumers	Groep	0	
ext	Groep	0	
result	Groep	0	
person	IDref/Object	1	Persoon die deze inschrijving betreft middels verwijzing (personId) of specificatie (Person object). Aanvullende eis Dit attribuut over bij deze inschrijving betrokken persoon is verplicht!
offering	IDref/Object	1	Opleidingsdeel die deze inschrijving betreft middels verwijzing (offeringId) of specificatie (CourseOffering object). Aanvullende eis Dit attribuut over het aangeboden opleidingsdeel is verplicht voor deze inschrijving!
issuer	Groep	0	

Tabel 4.5.A: Gegevensdefinities van CourseOfferingAssociation (Deelinschrijving)

4.6 Document (Document) object

Document
documentId : string
documentType : string
documentName : string

Figuur 4.6.A: Document (Document) gegevensmodel

Het gegevensobject **Document** is nieuw en bestaat nog niet in OOAPI. Dit object omvat de gegevens over een document. De gegevenselementen van Document zijn weergegeven in onderstaande tabel.

Gegeven	Type	#	Betekenis en waardebereik
documentId	IDref	1	Unieke identifier van document, deze ID moet worden gebruikt bij de opvraging van het document..
documentType	Enum	1	Typering van document. Waarde uit nieuwe enumeratie DocumentType (zie Tabel 4.6.B).
documentName	Tekst	1	Naam van document.

Tabel 4.6.A: Gegevensdefinities van Document (nieuw)

Let op, gezien de definitie van Document (Document) in bovenstaande Tabel 4.6.A zijn genoemde gegevens altijd verplicht zoals aangegeven in kolom #.

Waarde	Betekenis
assessmentForm	Beoordelingsformulier.
assessmentFormWithAnswers	Antwoordenvol met beoordelingsopmerkingen.
assessmentModel	Beoordelingsmodel/-voorschrift.
other	Elke document dat niet onder de andere typering past.

Tabel 4.6.B: Enumeratie DocumentType (nieuw).

4.7 Group (Toetsgroep) object

Group
groupId : uuid-string primaryCode : IdentifierEntry groupType : string name : string description : LanguageTypedString[] startDate : date-string endDate : date-string personCount : integer otherCodes : IdentifierEntry[] organization : organizationId or Organization object

Figuur 4.7.A: Group (Toetsgroep) gegevensmodel

Het gegevensobject Group (Toetsgroep) is een [OOAPI-object en -resource Group](#) en omvat de gegevens m.b.t. een groep studenten en medewerkers rondom een toets. De gegevens-elementen van Group voor deze uitwisselingen zijn weergegeven in onderstaande tabel.

Gegeven	Type	#	Betekenis en waardebereik
groupId	IDkey	1	Identificer van de Group (Toetsgroep of groep) in de vorm van een UUID, uniek in deze uitwisselingen. De identifier van deze groep wordt gemaakt door Deelnemerregistratie.
primaryCode	IdentifierEntry	1	Gegevensgroep (object IdentifierEntry) om getypeerde primaire code van groep mee te geven. Wanneer geen primaire code beschikbaar is dan kan de waarde van groupId worden gebruikt.
• codeType	Enum	1	Typing van de code/identifier in gegeven 'code'. Aanvullende eis Gebruik in veld codeType hier altijd de waarde "groupCode" om aan te geven dat dit de identifier van de Group (Toetsgroep) is:
• code	Tekst	1	Unieke, leesbare code/identifier volgens typing in gegevensveld 'codeType'.
groupType	Enum	1	Typing van groep;. Waarde uit waardelijst: {learning group, class, team}.
name	TaalTekst	+	Naam van groep. <i>Toelichting: gezien de duiding en herkenbaarheid wordt aanbevolen deze naam niet leeg te laten.</i>
description	TaalTekst	*	Omschrijving van groep.
startDate	Datum	?	Startdatum van groep.
endDate	Datum	?	Einddatum van groep.
personCount	Getal	?	Aantal personen in de groep.
otherCodes	IdentifierEntry	*	Lijst van objecten IdentifierEntry om getypeerde andere codes van de groep mee te geven, naast de primaire code in primaryCode.
• codeType	Enum	1	Typing van de code/identifier in gegevensveld 'code'.
• code	Tekst	1	Unieke, leesbare code/identifier volgens typing in gegevensveld 'codeType'.
consumers	Groep	0	
organization	IDref/Object	?	Organisatie van deze Group (Toetsgroep) middels verwijzing (organizationId) of specificatie (Organization object, zie paragraaf 4.9). Opmerking: wanneer geen endpoint beschikbaar is voor het ophalen van de informatie over de organisatie dan is het belangrijk om het gegevensobject mee te geven, en niet de verwijzing.
ext	Groep	0	

Tabel 4.7A: Gegevensdefinities van Group (Toetsgroep)

4.8 MembershipItem (Groepslidmaatschap) object

MembershipItem
personId : uuid-string
startDateTime : datetime-string
endDateTime : datetime-string
state : string
role : string

Figuur 4.8.A: MembershipItem gegevensmodel

Dit gegevensobject MembershipItem is een nieuw gedefinieerd OOAPI-object en omvat de gegevens m.b.t. het lidmaatschap van de student of medewerker (afnameleider, surveillant of beoordelaar) aan de Toetsgroep (Gruop). Deze MembershipItem heeft een directe verwijzing naar de betreffende Person (Persoon). De gegevenselementen van MembershipItem zijn weergegeven in onderstaande tabel.

Gegeven	Type	#	Betekenis en waardebereik
personId	IDref	1	Verwijzing naar de persoon die betrokken is bij dit groepslidmaatschap.
startDateTime	Datumtijd	1	Moment (datum en tijdstip) van start van groepslidmaatschap.
endDateTime	Datumtijd	?	Moment (datum en tijdstip) van einde van groepslidmaatschap.
state	Enum	?	Status van groepslidmaatschap. Aanvullende eis Beperkte waarden uit enumeratie MembershipStateType (zie Tabel 4.8.B).
role	Enum	?	Typering van rol van persoon in de Toetsgroep middels dit groepslidmaatschap. Aanvullende eis Beperkte waarden uit enumeratie RoleType (zie Tabel 4.3.B).
consumers	Groep	0	
ext	Groep	0	

Tabel 4.8.A: Gegevensdefinities van MembershipItem (Groepslidmaatschap)

Waarde	Betekenis
active	Status van de groepslidmaatschap is actief (gekoppeld).
canceled	Status van de groepslidmaatschap is niet meer actief omdat het is teruggetrokken/geannuleerd.

Tabel 4.8.B: Enumeratie MembershipStateType.

4.9 Organization (Organisatie) object

Organization
organizationId : uuid-string primaryCode : IdentifierEntry organizationType : string name : string shortname : string description : LanguageTypedString[] addresses : Address object[] link : uri-string logo : uri-string otherCodes : IdentifierEntry[] parent : organizationId or Organization object children : organizationId[] or Organization object[]

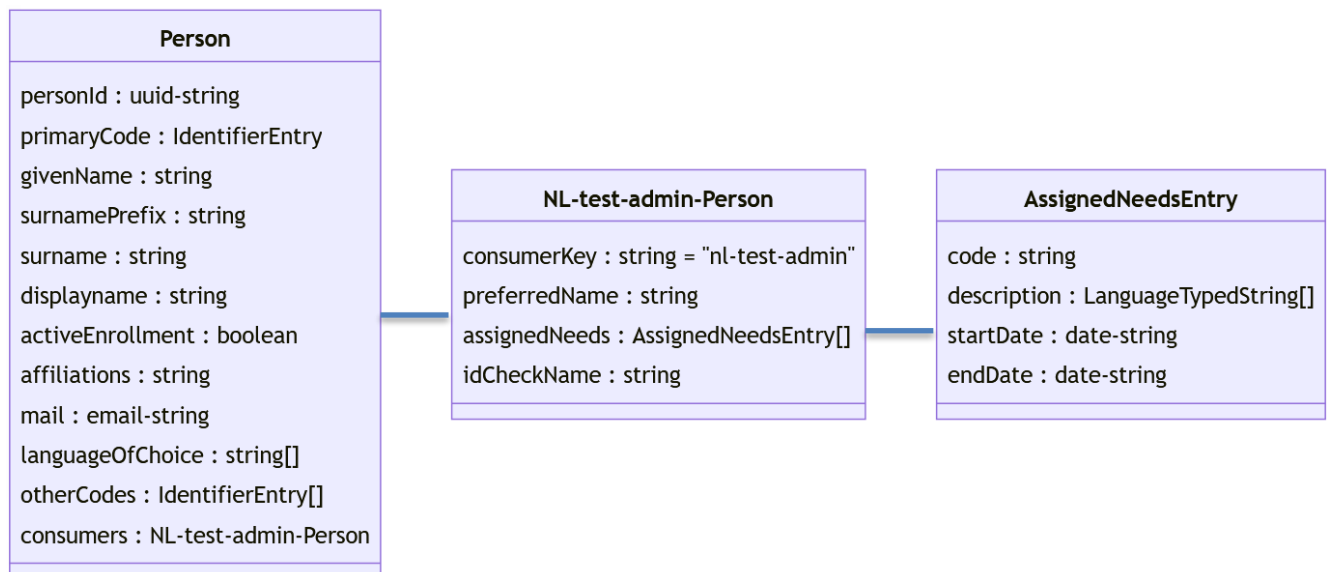
Figuur 4.9.A: Organization (Organisatie) gegevensmodel

Het gegevensobject Organization (Organisatie) is een [OOAPI-object en -resource Organization](#) en omvat de gegevens m.b.t. een organisatie. De gegevenselementen van Organization voor deze uitwisselingen zijn weergegeven in onderstaande tabel.

Gegeven	Type	#	Betekenis en waardebereik
organizationId	IDkey	1	Identifier van de Organization (Organisatie) in de vorm van een UUID, uniek in deze uitwisselingen.
primaryCode	IdentifierEntry	1	Gegevensgroep (object IdentifierEntry) om getypeerde primaire code van organisatie mee te geven. Wanneer geen primaire code beschikbaar is dan kan de waarde van organizationId worden gebruikt.
• codeType	Enum	1	Typering van de code/identifier in gegeven 'code'.
• code	Tekst	1	Unieke, leesbare code/identifier volgens typering in gegevensveld 'codeType'.
organizationType	Enum	1	Typering van organisatie. Waarde uit: {root, institute, department, faculty, branch, academy, school}.
name	TaalTekst	1	Naam van organisatie.
shortname	Tekst	1	Korte naam van organisatie. Waarde heeft maximale lengte van 256 karakters.
description	TaalTekst	*	Omschrijving van organisatie.
addresses	Groep	*	Adressen van organisatie.
link	Tekst	?	URL van website van organisatie.
logo	Tekst	?	Logo van organisatie.
otherCodes	IdentifierEntry	*	Lijst van objecten IdentifierEntry om getypeerde andere codes van organisatie mee te geven, naast de primaire code in primaryCode.
parent	IDref/Object	?	De organisatie-eenheid die de ouder (parent) is voor deze organisatie.
children	IDref/Object	*	De organisatie-eenheid waarvoor deze organisatie de ouder (parent) is.
consumers	Groep	0	
ext	Groep	0	

Tabel 4.9.A: Gegevensdefinities van Organization (Organisatie)

4.10 Person (Persoon) object



Figuur 4.10.A: Person gegevensmodel

Het gegevensobject Person (Persoon) is een [OOAPI-object en -resource Person](#) en omvat de gegevens m.b.t. een persoon. In deze context betreft de student (de persoon die deelneemt aan een zitting) of medewerker (de persoon in de rol van afnameleider, surveillant of beoordelaar). Let op, deze medewerker is niet altijd een persoon die in dienst is van de onderwijsinstelling.

De gegevenselementen van Person (Persoon) zijn weergegeven in onderstaande tabel.

Gegeven	Type	#	Betekenis en waardebereik
<code>personId</code>	IDkey	1	Identificer van Person (Persoon) in de vorm van een UUID, uniek in deze uitwisselingen. Deze identificer van persoon wordt gemaakt door Deelnemerregistratie.
<code>primaryCode</code>	IdentifierEntry	1	Gegevensgroep (object IdentifierEntry) om getypeerde primaire code van de persoon mee te geven. Dit is de code die door de school als unieke, vaak menselijk leesbare aanduiding wordt gebruikt. Wanneer geen primaire code beschikbaar is voor deze persoon dan kan de waarde van <code>personId</code> worden gebruikt.
• <code>codeType</code>	Enum	1	Typing van de code/identificer in gegevensveld 'code'. <i>Toelichting</i> Aanbevolen wordt hier voor studenten het studentnummer (<code>studentNumber</code>) en voor medewerkers het personeelsnummer (<code>organizationId</code>) of eventueel e-mailadres (<code>emailAddress</code>) te gebruiken.
• <code>code</code>	Tekst	1	Unieke, leesbare code/identificer volgens typing in gegevensveld 'codeType'.
<code>givenName</code>	Tekst	1	De voornaam of voornamen van persoon. Waarde heeft maximale lengte van 256 karakters. <i>Toelichting</i> Dit veld bevat een aaneenrijging van alle voornamen van de persoon (student) gescheiden door spatie. Als de voornamen niet beschikbaar zijn dan bij voorkeur dit veld gevuld met de eerste voornaam (zoals vermeld op het legitimatiebewijs), anders eventueel de roepnaam.
<code>preferredName</code>	Tekst	0	Dit attribuut wordt sinds versie 0.9.3 niet meer gebruikt (is verplaatst naar <code>consumer</code> binnen dit object).
<code>surnamePrefix</code>	Tekst	?	Voorvoegsel voor achternaam (surname) van persoon. Deze wordt gebruikt om in de applicaties te tonen.
<code>surname</code>	Tekst	1	Achternaam van persoon, zoals bij voorkeur aangesproken. Deze wordt gebruikt om in de applicaties te tonen. Waarde heeft maximale lengte van 256 karakters.
<code>displayName</code>	Tekst	1	Weergavenaam van persoon die wordt getoond op het computerscherm bij digitale toetsafname. Dit betreft de voorkeursbenaming (achternaam en roepnaam) in de attributen <code>surname</code> , <code>preferredName</code> en <code>surnamePrefix</code> . Waarde heeft maximale lengte van 256 karakters. <i>Toelichting</i> Dit veld bevat een aaneenrijging van de voorkeur-naamgegevens van de persoon in de volgorde achternaam en roepnaam (eventueel aangevuld met voorvoegsel), gescheiden door komma en spatie (tussen roepnaam en voorvoegsel alleen spatie), bijvoorbeeld "Graaf, Jacob van der".
<code>initials</code>	Tekst	0	
<code>activeEnrollment</code>	Boolean	1	Geeft aan of student een actieve onderwijsinschrijving heeft (studiegeld betaald); voor medewerker geeft dit aan of deze persoon in actieve dienst is. De waarde is hier altijd True.
<code>dateOfBirth</code>	Datum	0	
<code>cityOfBirth</code>	Tekst	0	
<code>countryOfBirth</code>	Tekst	0	
<code>nationality</code>	Tekst	0	
<code>dateOfNationality</code>	Datum	0	
<code>affiliations</code>	Enum	*	De relaties die persoon heeft met de organisaties; waarde is uit enumeratie "student" (voor student) en "employee" (voor medewerker) of "guest" (anders).
<code>mail</code>	Tekst	1	Het primaire e-mailadres van persoon. <i>Let op, dit gegeven wordt niet gebruikt maar is verplicht.</i>
<code>secondaryMail</code>	Tekst	0	
<code>telephoneNumber</code>	Tekst	0	
<code>mobileNumber</code>	Tekst	0	
<code>photoSocial</code>	URI	0	

photoOfficial	URI	0	
gender	Enum	0	
titlePrefix	Tekst	0	
titleSuffix	Tekst	0	
office	Tekst	0	
address	Groep	0	
ICENAME	Tekst	0	
ICEPHONE_NUMBER	Tekst	0	
ICERELATION	Enum	0	
languageOfChoice	Enum	*	Voorkeurstaal van persoon. Waarde volgens RFC3066, bijvoorbeeld "nl-NL" voor Nederlands.
otherCodes	IdentifierEntry	*	Lijst van gegevensgroepen (object IdentifierEntry) om getypeerde andere codes van de persoon mee te geven, naast de primaire code in primaryCode.
• codeType	Enum	1	Typering van de code/identificatie in gegevensveld 'code'. Aanvullende eis Gebruik hier in veld codeType de waarden "studentNumber", "username", "accountid", of "eckid" om aan te geven dat dit de identificatie van deze persoon typeert.
• code	Tekst	1	Unieke, leesbare code/identificatie volgens typering in gegevensveld 'codeType'.
consumers	Groep	1	Aanvullende gegevens over persoon in NL-test-admin-Person. alleen bedoeld voor een student. Aanvullende eis Attributen gebruiken zoals onderstaand gedefinieerde gegevensgroep.
• consumerKey	IDref	1	Typering van de aanvullende gegevens. Hier altijd de waarde "nl-test-admin" gebruiken.
• preferredName	Tekst	1	Roepnaam van persoon, zoals de persoon bij voorkeur wordt aangeroept. Deze wordt gebruikt om in de applicaties te tonen. Waarde heeft maximale lengte van 256 karakters. <i>Let op, dit veld is verplaatst binnen object Person!</i>
• personalNeeds	Enum	0	Dit attribuut wordt sinds versie 0.9.3 van deze specs niet meer gebruikt (is vervangen door assignedNeeds).
• assignedNeeds	Groep	*	Lijst van toegekende hulpmiddelen (inclusief extra tijd) voor de student gebaseerd op diens behoefte. Deze toegekende hulpmiddelen worden later vertaald naar persoonlijke hulpmiddelen voor een specifieke deelname. Voorbeelden zijn (code): "ExtraTimeOnlyMath25%" (=Extra tijd van 25% van de totale tijd van een toets waarin rekenen voorkomt), "ExtraTimeOnlyMath30Min" (=Extra tijd van 30 minuten bij een toets waarin rekenen voorkomt of "ExtraTimeDigitalTests25%" (=Extra tijd van 25% van de totale tijd van een digitale toets). Aanvullende eis: alleen gebruikt voor persoonsgegevens van een student in flow 1.
• code	Tekst	1	Door mens leesbare code, identificatie of korte aanduiding van het toegekende hulpmiddel.
• description	TaalTekst	+	Omschrijving van het toegekende hulpmiddel.
• startDate	Datum	?	Datum van de dag waarop de toekenning van het hulpmiddel begint.
• endDate	Datum	?	Datum van de dag waarop de toekenning van het hulpmiddel eindigt.
• idCheckName	Tekst	1	De achternaam en voornamen van de student, zoals vermeld op het legitimatiebewijs. Aanvullende eis Dit veld bevat een aaneenrijging van de naamgegevens van de persoon zoals op legitimatiebewijs in de volgorde achternaam en voornamen (eventueel aangevuld met voorvoegsel), gescheiden door komma en spatie (tussen voornamen en voorvoegsel alleen spatie), bijvoorbeeld "Graaf, Jacob van der". Als de naam geheim of niet beschikbaar is dan waarde "secret" respectievelijk "notAvailable" vermelden.
ext	Groep	0	

Tabel 4.10.A: Gegevensdefinities van Person (Persoon)

4.11 Program (Opleidingsprogramma) object

Program
programId : uuid-string primaryCode : IdentifierEntry programType : string name : LanguageTypedString[] abbreviation : string description : LanguageTypedString[] teachingLanguage : string modeOfStudy : string duration: string firstStartDate : date-string levelOfQualification : string otherCodes : IdentifierEntry[] validFrom : date-string validTo : date-string

Figuur 4.11.A: Program (Opleidingsprogramma) gegevensmodel

Het gegevensobject Program (Opleidingsprogramma) is een [OOAPI-object en -resource Program](#) en omvat de gegevens m.b.t. het opleidingsprogramma van een opleiding.

De gegevenselementen van de Program zijn weergegeven in onderstaande tabel.

Gegeven	Type	#	Betekenis en waardebereik
programId	IDkey	1	Identifier van het Program (Opleidingsprogramma) in de vorm van een UUID, uniek in deze uitwisselingen. Deze identifier van opleidingsprogramma wordt gemaakt door Deelnemerregistratie.
primaryCode	IdentifierEntry	1	Gegevensgroep (object IdentifierEntry) om getypeerde primaire code van het opleidingsprogramma mee te geven. Wanneer geen primaire code beschikbaar is voor dit opleidingsprogramma dan kan de waarde van programId worden gebruikt.
• codeType	Enum	1	Typering van de code/identifier in gegevensveld 'code'. Gebruik hier in veld codeType de nieuwe waarde "nationalEducationCode" voor de RIO opleidingscode (voorheen crebocode). <i>Let op, deze waarde "nationalEducationCode" is geen waarde van enumeratie codeType binnen object Program (en zonder prefix "x-") en wijkt hiermee af van OOAPI v5!</i>
• code	Tekst	1	Unieke, leesbare code/identifier volgens typering in gegevensveld 'codeType'.
programType	Enum	1	Typering van het opleidingsprogramma. Aanvullende eis Waarde is altijd "program" (niet gebruikte waarden zijn minor, honours, specialization, track).
name	TaalTekst	+	Naam van opleidingsprogramma. <i>Toelichting: gezien de duiding en herkenbaarheid wordt aanbevolen deze naam niet leeg te laten.</i>
abbreviation	Tekst	1	Afkorting van opleidingsprogramma. Waarde heeft maximale lengte van 256 karakters.
description	TaalTekst	+	Omschrijving van opleidingsprogramma.
teachingLanguage	Enum	1	Voertaal waarin opleidingsprogramma wordt gevolgd. Waarde is 3-letter code volgens ISO 639-2, zoals "nld" voor Nederlands. Aanvullende eis Gegeven niet gebruikt maar verplicht; waarde is altijd "nld".
studyLoad	Groep	0	
qualificationAwarded	Tekst	0	
modeOfStudy	Enum	?	Opleidingsmode. Waarde uit bestaande enumeratie ModeOfStudyType (zie Tabel 4.11.B).
modeOfDelivery	Tekst	0	
duration	Tijdsduur	?	Geeft de duur van het opleidingsprogramma aan. Waarde is conform ISO 8601 duration.
firstStartDate	Datum	1	Eerste startdatum van opleidingsprogramma.
levelOfQualification	Enum	?	MBO-opleidingsniveau van dit opleidingsprogramma volgens NLQF-niveaus (en EQF-niveaus). Waarde voor MBO uit tekstuele enumeratie {1,2,3,4} (originele enumeratie is {1, 2, 3, 4, 4+, 5, 6, 7, 8}).
level	Tekst	0	
sector	Tekst	0	
fieldOfStudy	Tekst	0	
enrollment	TaalTekst	0	
resources	Tekst	0	

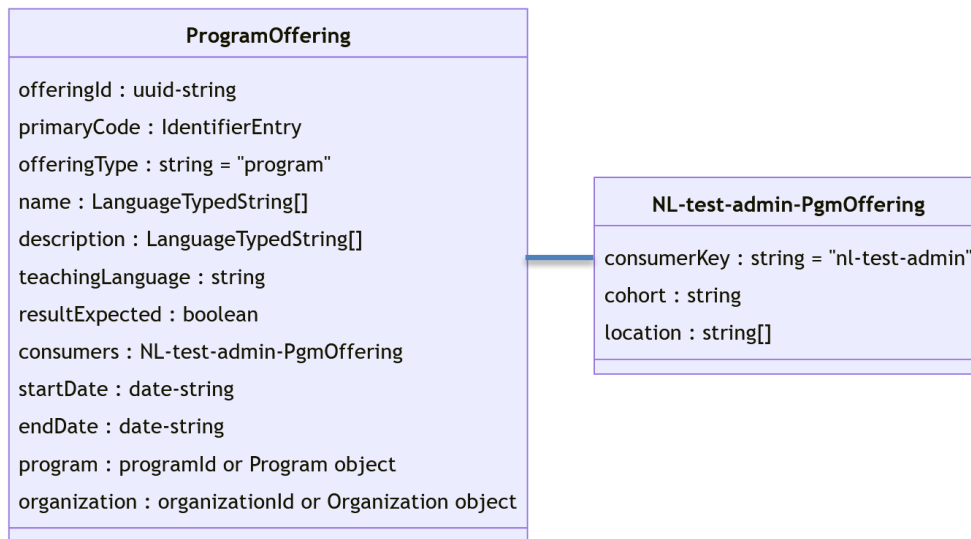
learningOutcomes	TaalTekst	0	
assessment	TaalTekst	0	
admissionRequirements	TaalTekst	0	
qualificationRequirements	TaalTekst	0	
link	Tekst	0	
educationSpecification	IDref/Object	0	
otherCodes	IdentifierEntry	*	Lijst van gegevensgroepen (object IdentifierEntry) om getypeerde andere codes van het opleidingsprogramma mee te geven, naast de primaire code in primaryCode.
• codeType	Enum	1	Typering van de code/identificer in gegevensveld 'code'. Gebruik hier in veld codeType de nieuwe waarde "nationalEducationCode" voor de RIO opleidingscode (voorheen crebocode). <i>Let op, deze waarde "nationalEducationCode" is geen waarde van enumeratie codeType binnen object Program (en zonder prefix "x-") en wijkt hiermee af van OOAPl v5!</i>
• code	Tekst	1	Unieke, leesbare code/identificer volgens typering in gegevensveld 'codeType'.
addresses	Groep	0	
parent	IDref/Object	0	
children	IDref/Object	0	
coordinators	IDref/Object	0	
organization	IDref/Object	0	
consumers	Groep	0	
ext	Groep	0	
validFrom	Datum	1	De eerste dag dat dit programma geldig is.
validTo	Datum	1	De dag dat dit programma ongeldig wordt.

Tabel 4.11.A: Gegevensdefinities van Program (Opleidingsprogramma)

Waarde	Betekenis
full-time	fulltime
part-time	parttime
dual training	duaal
selfpaced	eigen tempo
extraneous	extraneus student: examendeelnemer (student mag alleen aan de toetsen deelnemen / beperkt collegegeld)

Tabel 4.11.B: Enumeratie ModeOfStudyType.

4.12 ProgramOffering (Opleiding) object



Figuur 4.12.A: ProgramOffering (Opleiding) gegevensmodel

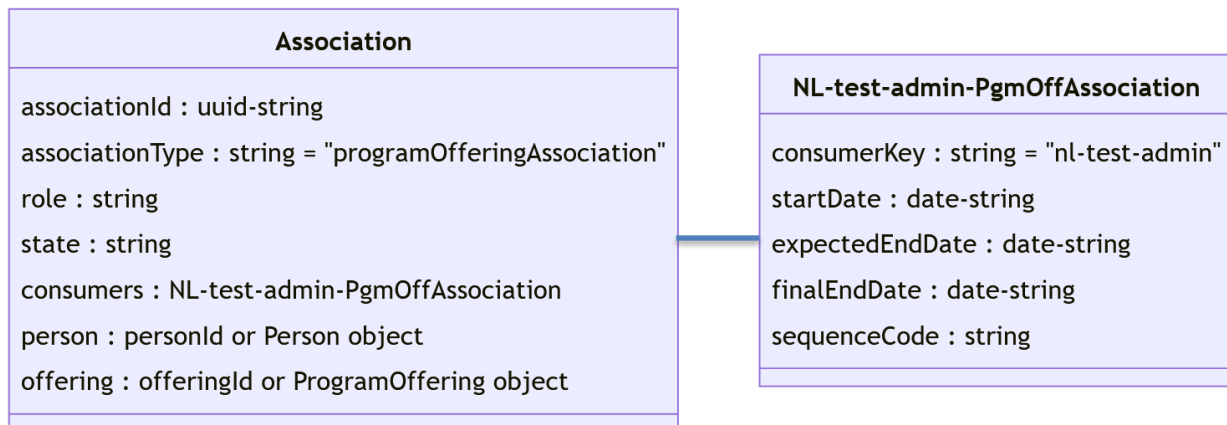
Het gegevensobject ProgramOffering (Opleiding) is een [OOAPI-object en -resource Offering van het type program](#) en omvat de gegevens m.b.t. een opleiding. Deze Offering heeft een directe verwijzing naar het betreffende Program (Opleidingsprogramma/Kwalificatie). De gegevens-elementen van ProgramOffering zijn weergegeven in onderstaande tabel.

Gegeven	Type	#	Betekenis en waardebereik
<code>offeringId</code>	IDkey	1	Identificer van de ProgramOffering (Opleiding) in de vorm van een UUID, uniek in deze uitwisselingen. Deze identifier van opleiding wordt gemaakt door Deelnemerregistratie.
<code>primaryCode</code>	IdentifierEntry	1	Gegevensgroep (object IdentifierEntry) om getypeerde primaire code van de opleiding mee te geven. Wanneer geen primaire code beschikbaar is voor deze opleiding dan kan de waarde van <code>offeringId</code> worden gebruikt.
• <code>codeType</code>	Enum	1	Typing van de code/identifier in gegevensveld 'code'. Aanvullende eis Gebruik hier altijd de waarde "offeringCode" om aan te geven dat dit de identifier van de ProgramOffering (opleiding) is.
• <code>code</code>	Tekst	1	Unieke, leesbare code/identifier volgens typing in gegevensveld 'codeType'.
<code>offeringType</code>	Enum	1	Typing van de offering Aanvullende eis Waarde is altijd "program" (niet gebruikte waarden zijn course en component).
<code>academicSession</code>	IDref/Object	0	
<code>name</code>	TaalTekst	+	Naam van opleiding. <i>Toelichting: gezien de duiding en herkenbaarheid wordt aanbevolen deze naam niet leeg te laten.</i>
<code>abbreviation</code>	Tekst	0	
<code>description</code>	TaalTekst	+	Omschrijving van opleiding.
<code>teachingLanguage</code>	Enum	1	Voertaal waarin deze opleiding wordt gevolgd. Waarde is 3-letter code volgens ISO 639-2, zoals "nld" voor Nederlands. Aanvullende eis Gegeven niet gebruikt maar verplicht; waarde is altijd "nld".
<code>modeOfDelivery</code>	Enum	0	
<code>maxNumberStudents</code>	Getal	0	
<code>enrolledNumberStudents</code>	Getal	0	
<code>pendingNumberStudents</code>	Getal	0	
<code>minNumberStudents</code>	Getal	0	
<code>resultExpected</code>	Boolean	1	Geeft aan of er voor deze opleiding resultaat wordt verwacht. Aanvullende eis Waarde is hier altijd false.
<code>resultValueType</code>	Enum	0	
<code>link</code>	Tekst	0	
<code>otherCodes</code>	IdentifierEntry	0	
<code>consumers</code>	Groep	1	Aanvullende gegevens in object NL-test-admin-PgmOffering. Verplicht voor iedere opleiding. Aanvullende eis Attributen gebruiken zoals onderstaand gedefinieerde gegevensgroep.
• <code>consumerKey</code>	IDref	1	Typing van de aanvullende gegevens. Hier altijd de waarde "nl-test-admin" gebruiken.
• <code>cohort</code>	Tekst	1	Welk OER wordt toegepast (kan keuze voor toetsmiddel bepalen).
• <code>location</code>	Tekst	1	Onderwijslocatie om de campus of vestiging van de school te duiden, bijvoorbeeld "Campus Groningen" of "WINS".
<code>ext</code>	Groep	0	
<code>startDate</code>	Datum	1	Startdatum van opleiding.
<code>endDate</code>	Datum	1	Einddatum van opleiding.
<code>enrollStartDate</code>	Datum	0	
<code>enrollEndDate</code>	Datum	0	

flexibleEntryPeriodStart	Datum	0	
flexibleEntryPeriodEnd	Datum	0	
addresses	Groep	0	
priceInformation	Groep	0	
room	Groep	0	
program	IDref/Object	1	Opleidingsprogramma van de opleiding middels verwijzing (programId) of specificatie (Program object). Toelichting Dit attribuut over bij deze opleiding betrokken opleidingsprogramma wordt aanbevolen! Opmerking: wanneer geen endpoint beschikbaar is voor het ophalen van de informatie over het opleidingsprogramma dan is het belangrijk het gegevensobject mee te geven, en niet de verwijzing.
organization	IDref/Object	?	Organisatie van deze ProgramOffering (Opleiding) middels verwijzing (organizationId) of specificatie (Organization object, zie paragraaf 4.9) Opmerking: wanneer geen endpoint beschikbaar is voor het ophalen van de informatie over de organisatie dan is het belangrijk om het gegevensobject mee te geven, en niet de verwijzing.

Tabel 4.12.A: Gegevensdefinities van ProgramOffering (Opleiding)

4.13 ProgramOfferingAssociation (Opleidingsinschrijving) object



Figuur 4.13.A: ProgramOfferingAssociation gegevensmodel

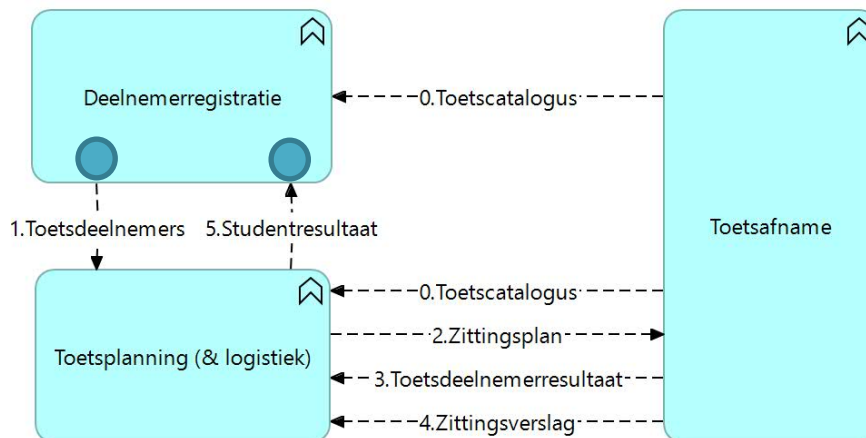
Dit gegevensobject ProgramOfferingAssociation (Opleidingsinschrijving) is een [OOAPI-object en -resource Association van het type programOfferingAssociation](#) en omvat de gegevens m.b.t. een inschrijving voor de opleiding door een student. Deze Association bevat een verwijzing naar het betreffende object Person (persoon). De gegevenselementen van ProgramOfferingAssociation zijn weergegeven in onderstaande tabel.

Gegeven	Type	#	Betekenis en waardebereik
associationId	IDkey	1	Identificer van ProgramOfferingAssociation (Opleidingsinschrijving) in de vorm van een UUID, uniek in deze uitwisselingen. Deze identificer van opleidingsinschrijving wordt gemaakt door Deelnemerregistratie
associationType	Enum	1	Typering van inschrijving; waarde is hier altijd "programOfferingAssociation" voor opleidingsinschrijving.
role	Enum	1	Typering van rol van de persoon in deze opleidingsinschrijving. Aanvullende eis Hier alleen waarde "Student" gebruiken uit enumeratie RoleType (zie Tabel 4.3.B).
state	Enum	1	Status van opleidingsinschrijving. Aanvullende eis Beperkte waarden uit enumeratie AssociationStateType (zie Tabel 4.1.C).
remoteState	Enum	0	
consumers	Groep	1	Aanvullende gegevens in object NL-test-admin-PgmOffAssociation. Verplicht voor iedere opleidingsinschrijving. Aanvullende eis Attributen gebruiken zoals onderstaand gedefinieerde gegevensgroep.
• consumerKey	IDref	1	Typering van de aanvullende gegevens. Hier altijd de waarde "nl-test-admin" gebruiken.
• startDate	Datum	?	Begindatum van toetsafnameperiode
• expectedEndDate	Datum	?	Verwachte einddatum van toetsafnameperiode
• finalEndDate	Datum	?	Definitieve einddatum van toetsafnameperiode
• sequenceCode	Tekst	?	De transactiecode (of volgnummer) van de opleidingsinschrijving, bijvoorbeeld loopbaannummer of nummer van opleidingsblad. Bedoeld te gebruiken in de overdracht van opleidingsinschrijving in flow 1.
ext	Groep	0	
result	Groep	0	
person	IDref/Object	1	Persoon die deze opleidingsinschrijving betreft middels verwijzing (personId) of specificatie (Person object). Aanvullende eis Hier de ID van de persoon gebruiken (en niet het gegevensobject). Aanvullende eis Dit attribuut over bij deze inschrijving betrokken persoon is verplicht!
offering	IDref/Object	1	Opleiding die deze inschrijving betreft middels verwijzing (offeringId) of specificatie (ProgramOffering object). Aanvullende eis Hier de ID van de opleiding gebruiken (en niet het gegevensobject). Aanvullende eis Dit attribuut over de aangeboden opleiding is verplicht voor deze inschrijving!
issuer	Groep	0	

Tabel 4.13.A: Gegevensdefinities van ProgramOfferingAssociation (Opleidingsinschrijving)

5 Koppelvlakken

5.1 Deelnemerregistratie koppelvlak



Figuur 5.1.A: Koppelvlak van Deelnemerregistratie

Endpoint	Operaties	Informatiestroom	Aanvrager
/associations/{associationId}	GET	1. Toetsdeelnemers: Inschrijving	Toetsplanning
/associations/{associationId}	PUT, PATCH	5. Studentresultaat	Toetsplanning
/groups?q={zoektekst}	GET	1. Toetsdeelnemers: Toetsgroepen	Toetsplanning
/groups/{groupId}/persons	GET	1. Toetsdeelnemers: Personen	Toetsplanning
/offerings/{offeringId}	GET	1. Toetsdeelnemers: Planbare toets / Opleiding / Opleidingsdeel	Toetsplanning
/offerings/{offeringId}/associations/	GET	1. Toetsdeelnemers: Inschrijvingen	Toetsplanning
/persons/{personId}	GET	1. Toetsdeelnemers: Persoon	Toetsplanning

Tabel 5.1.A: Endpoints en operaties van Deelnemerregistratie

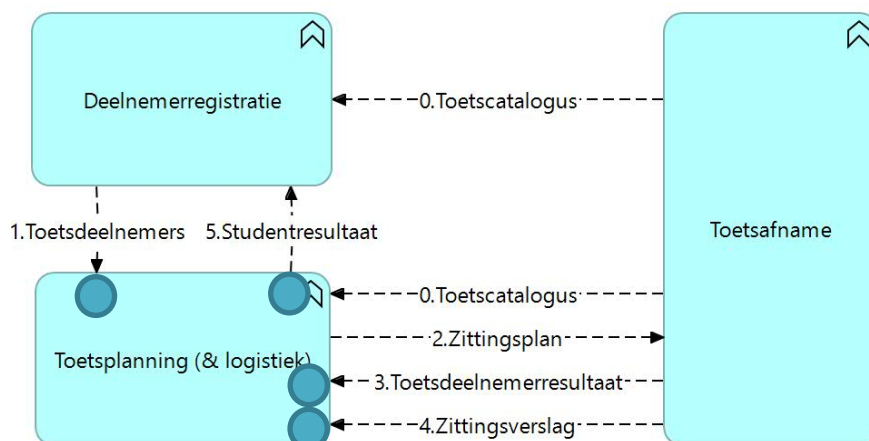
Let op, bij de GET-operaties zijn de volgende foutcodes naast 200 (OK) voorzien: 400 (Bad request), 401 (Unauthorized), 403 (Forbidden), 404 (Not found), 405 (Method not allowed), 429 (Too many requests) en 500 (Internal server error)

Let op, bij de PUT-operaties zijn de volgende foutcodes naast 200 (OK) voorzien: 400 (Bad request), 401 (Unauthorized), 403 (Forbidden), 404 (Not found), 405 (Method not allowed), 429 (Too many requests) en 500 (Internal server error)

Let op, bij de PATCH-operaties is de volgende foutcode naast 200 (OK) voorzien: 400 (Bad request).

Afgesproken is om bij de GET en PUT operaties de gegevensobjecten gelijkvormig te houden, dat wil zeggen dat in het antwoordbericht van de GET operatie en het verzoek van de PUT operatie het object het attribuut id bevat. Met als voorwaarde dat bij de PUT operatie het id in het object gelijk is aan de id als padparameter; als het id afwijkt van de padparameter dan mag het verzoek worden geweigerd middels foutmelding. *Bijvoorbeeld, het Association object in het antwoordbericht van GET /associations/{associationId} en in het verzoekbericht van PUT /associations/{associationId} bevatten allebei het attribuut associationId. In de PATCH operatie wordt het id van het object nooit gebruikt omdat het id nooit mag worden veranderd.*

5.2 Toetsplanning koppelvak



Figuur 5.2.A: Koppelvak van Toetsplanning (& logistiek)

Endpoint	Operaties	Informatiestroom	Aanvrager
/associations/{associationId}	PUT, PATCH	1. Toetsdeelnemers: Inschrijving	Deelnemerregistratie
/associations/{associationId}	PATCH	3. Toetsdeelnemerresultaat	Toetsafname
/offerings/{offeringId}	PUT, PATCH	1. Toetsdeelnemers: Planbare toets / Opleiding	Deelnemerregistratie
/offerings/{offeringId}	PATCH	4. Zittingsverslag	Toetsafname
/persons/{personId}	PUT	1. Toetsdeelnemers: Persoon	Deelnemerregistratie

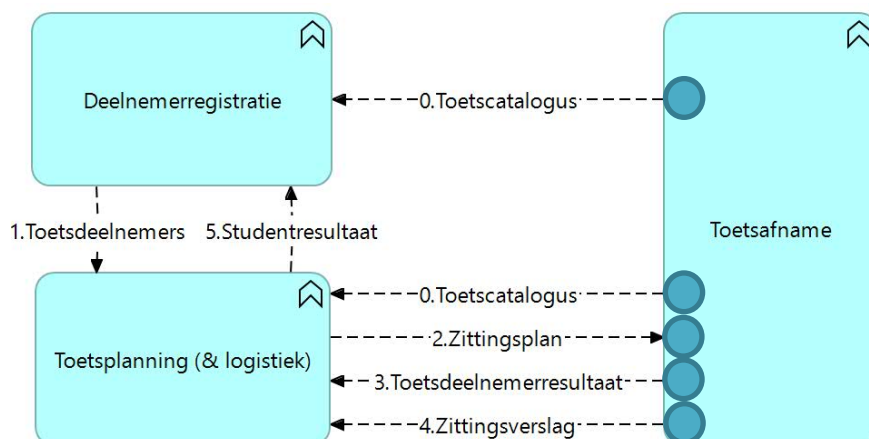
Tabel 5.2.A: Endpoints en operaties van Toetsplanning (& logistiek)

Let op, bij de PUT-operaties zijn de volgende foutcodes naast 200 (OK) voorzien: 400 (Bad request), 401 (Unauthorized), 403 (Forbidden), 404 (Not found), 405 (Method not allowed), 429 (Too many requests) en 500 (Internal server error)

Let op, bij de PATCH-operaties is de volgende foutcode naast 200 (OK) voorzien: 400 (Bad request).

Afgesproken is om bij de GET en PUT operaties de gegevensobjecten gelijkvormig te houden, dat wil zeggen dat in het antwoordbericht van de GET operatie en het verzoek van de PUT operatie het object het attribuut id bevat. Met als voorwaarde dat bij de PUT operatie het id in het object gelijk is aan de id als padparameter; als het id afwijkt van de padparameter dan mag het verzoek worden geweigerd middels foutmelding. *Bijvoorbeeld, het Association object in het antwoordbericht van GET /associations/{associationId} en in het verzoekbericht van PUT /associations/{associationId} bevatten allebei het attribuut associationId. In de PATCH operatie wordt het id van het object nooit gebruikt omdat het id nooit mag worden veranderd.*

5.3 Toetsafname koppelvlak



Figuur 5.3.A: Koppelvlak van Toetsafname

Endpoint	Operaties	Informatiestroom (v)	Aanvrager
/associations/{associationId}	PUT, PATCH	2. Zittingsplan: Deelname	Toetsplanning
/associations/{associationId}	GET	3. Toetsdeelnemerresultaat	Toetsplanning
/associations/{associationId}/url	GET	2. Zittingsplan: Deelname-url	Toetsplanning
/components	GET	0. Toetscatalogus	Deelnemerregistratie Toetsplanning
/components/{componentId}	GET	0. Toetscatalogus	Deelnemerregistratie Toetsplanning
/documents/{documentId}	GET	3. Toetsdeelnemerresultaat 4. Zittingsverslag	Toetsplanning
/offerings/{offeringId}	GET, PUT, PATCH	2. Zittingsplan: Zitting	Toetsplanning
/offerings/{offeringId}/associations/	GET	2. Zittingsplan: Deelnames 3. Toetsdeelnemerresultaat	Toetsplanning

Tabel 5.3.A: Endpoints en operaties van Toetsafname

Let op, bij de GET-operaties zijn de volgende foutcodes naast 200 (OK) voorzien: 400 (Bad request), 401 (Unauthorized), 403 (Forbidden), 404 (Not found), 405 (Method not allowed), 429 (Too many requests) en 500 (Internal server error)

Let op, bij de PUT-operaties zijn de volgende foutcodes naast 200 (OK) voorzien: 400 (Bad request), 401 (Unauthorized), 403 (Forbidden), 404 (Not found), 405 (Method not allowed), 429 (Too many requests) en 500 (Internal server error)

Let op, bij de PATCH-operaties is de volgende foutcode naast 200 (OK) voorzien: 400 (Bad request).

Afgesproken is om bij de GET en PUT operaties de gegevensobjecten gelijkvormig te houden, dat wil zeggen dat in het antwoordbericht van de GET operatie en het verzoek van de PUT operatie het object het attribuut id bevat. Met als voorwaarde dat bij de PUT operatie het id in het object gelijk is aan de id als padparameter; als het id afwijkt van de padparameter dan mag het verzoek worden geweigerd middels foutmelding.

Bijvoorbeeld, het Association object in het antwoordbericht van GET /associations/{associationId} en in het verzoekbericht van PUT /associations/{associationId} bevatten allebei het attribuut associationId.

In de PATCH operatie wordt het id van het object nooit gebruikt omdat het id nooit mag worden veranderd.