

IIIF-beeldinfrastructuur gekoppeld aan VKC-ecosysteem, fase 3:
Crowdsourcing, IIIF-manifestenbibliotheek en IIIF-storytelling

WP3 Proefopstelling van een IIIF-manifestenbibliotheek/-catalogus met Search Engine Optimization (SEO)

Taak 7: Opzetten van een ontwerparchitectuur

Binnen deze taak werd de selectie van gewenste functionaliteiten samengevoegd in een mogelijke ontwerparchitectuur voor de IIIF-manifestenbibliotheek.

Tevens moest de ontwerparchitectuur rekening houden met het gegeven dat de manifestenbibliotheek moet aansluiten op de ecosystemen van de partners VKC, Plantin-Moretus en Openbare Bibliotheek Brugge. Ook de SEO-principes moesten hierin worden ingebracht door de gekozen aannemer (Inuits) waarbij ook gevraagd werd om zelf een voorstel te doen hoe die best kunnen gerealiseerd worden.

Deliverable 9. Ontwerparchitectuur van een IIIF-manifestenbibliotheek met SEO-optimalisatie

Medewerkers werkpakket 3

Pascal Ennaert, Rein Debrulle (VKC)

Nastasia Vanderperren (meemoo)

Kristof Selleslach, Anneleen Decraene (Museum Plantin-Moretus)

Koen Calis (Openbare Bibliotheek Brugge)

Bram Oosterlynck, Jeroen Budts (Inuits BV)



Inhoud

1. Inleiding	2
2. Overzicht componenten	2
3. Architectuur	4
3.1 Grafische voorstelling	4
3.2 Toelichting bij de belangrijkste softwarecomponenten	5
3.2.1 pyoai	5
3.2.2 Elody	5
3.2.3 Mirador	5
4. Werkwijze	6
4.1 Ophalen van manifesten bij partners	6
4.2 Persistentie van manifesten in de bibliotheek	6
4.3 Search Engine Optimization (SEO)	7



Vlaanderen
verbeelding werkt



BRUGGE



1. Inleiding

Het beschikbaar stellen van makkelijk vindbare digitale content, over de grenzen van erfgoedorganisaties heen, is een belangrijke uitdaging. Een IIF-manifestenbibliotheek moet hierop een antwoord bieden door digitale beelden en metadata, verbonden met erfgoedcollecties, vindbaar en herbruikbaar te maken voor gebruikers met minder technische vaardigheden.

In een vorige fase ([WP3 Deliverable 8](#)) werd nagegaan welke functionaliteiten/kenmerken onderdeel van zo'n IIF-manifestenbibliotheek kunnen zijn. In dit document wordt de architectuur van zo'n IIF-manifestenbibliotheek, in proefopstelling, nader toegelicht. Onder 'architectuur' wordt begrepen: een high-level technische blauwdruk voor deze IIF-manifestenbibliotheek waarin niet alleen staat aangegeven hoe de (abstracte) functionaliteiten en wensen in de praktijk gebracht worden, maar ook welke de componenten zijn waaruit de applicatie wordt opgebouwd. De focus van deze architectuur ligt op het hergebruik van bestaande open-source componenten en courante IIF-technologie.

2. Overzicht componenten

Om alle gewenste functionaliteiten te voorzien en te komen tot een werkend platform, zijn verschillende componenten nodig. Hieronder volgt een uitgezoomde olijsting van de componenten die men voor een IIF-manifestenbibliotheek vooropstelde, telkens met een korte toelichting. *In de proefopstelling werden uiteindelijk niet alle vooropgestelde componenten geïntegreerd.*

- [elody](#)

Een aaneenschakeling van microservices die samen een DAMS (Digital Assets Management System) vormen. Dit DAMS is de basiscomponent van de proefopstelling.

- [IIF Presentation Validator](#)

Een component verantwoordelijk voor de verrijking van de IIF-manifestenbibliotheek met uitsluitend geldige IIF-manifesten (zowel volgens de v2.x als de v3.0 standaard). De IIF Presentation Validator toetst immers elk IIF-manifest aan de voorgeschreven specificaties en zorgt voor foutmeldingen en waarschuwingen wanneer vereisten en aanbevelingen niet zijn gevolgd.

- [pyoai](#)

Een Python implementatie van het OAI-PMH (Open Archives Initiative Protocol for Metadata Harvesting). Deze component ligt aan de basis van een voor dit project uitgewerkte microservice die verantwoordelijk is voor het ophalen van IIF-manifesten van de verschillende projectpartners.

- [Universal Viewer](#)

Een open-source IIIF-viewer, ontwikkeld door Digirati en geadopteerd door een actieve community onder aansturing van Mnemoscene. De viewer zorgt voor de presentatie van beelden, 3D-modellen en audiovisueel materiaal, telkens met de bijhorende metadata, aan de hand van de URL van het IIIF-manifest. Deze viewer wordt in de IIIF-manifestenbibliotheek geïntegreerd. *(In de uiteindelijke opstelling bleek de Universal Viewer niet compatibel met de componenten van de front-end.)*

- [TIFY](#)

Een lichte en mobielvriendelijke IIIF-documentviewer, gebouwd met Vue.js. De viewer ondersteunt versie 2 van de IIIF Presentation API en Image API. Omdat elke viewer andere klemtonen legt, wordt ook deze in de IIIF-manifestenbibliotheek geïntegreerd. Dat TIFY enkel IIIF-manifesten versie 2 kan inlezen is een handicap. In een manifestenbibliotheek moeten meerdere viewers worden aangeboden, bij voorkeur viewers die alle manifesten, hetzij versie 2 hetzij versie 3 kunnen lezen.

- [Mirador](#)

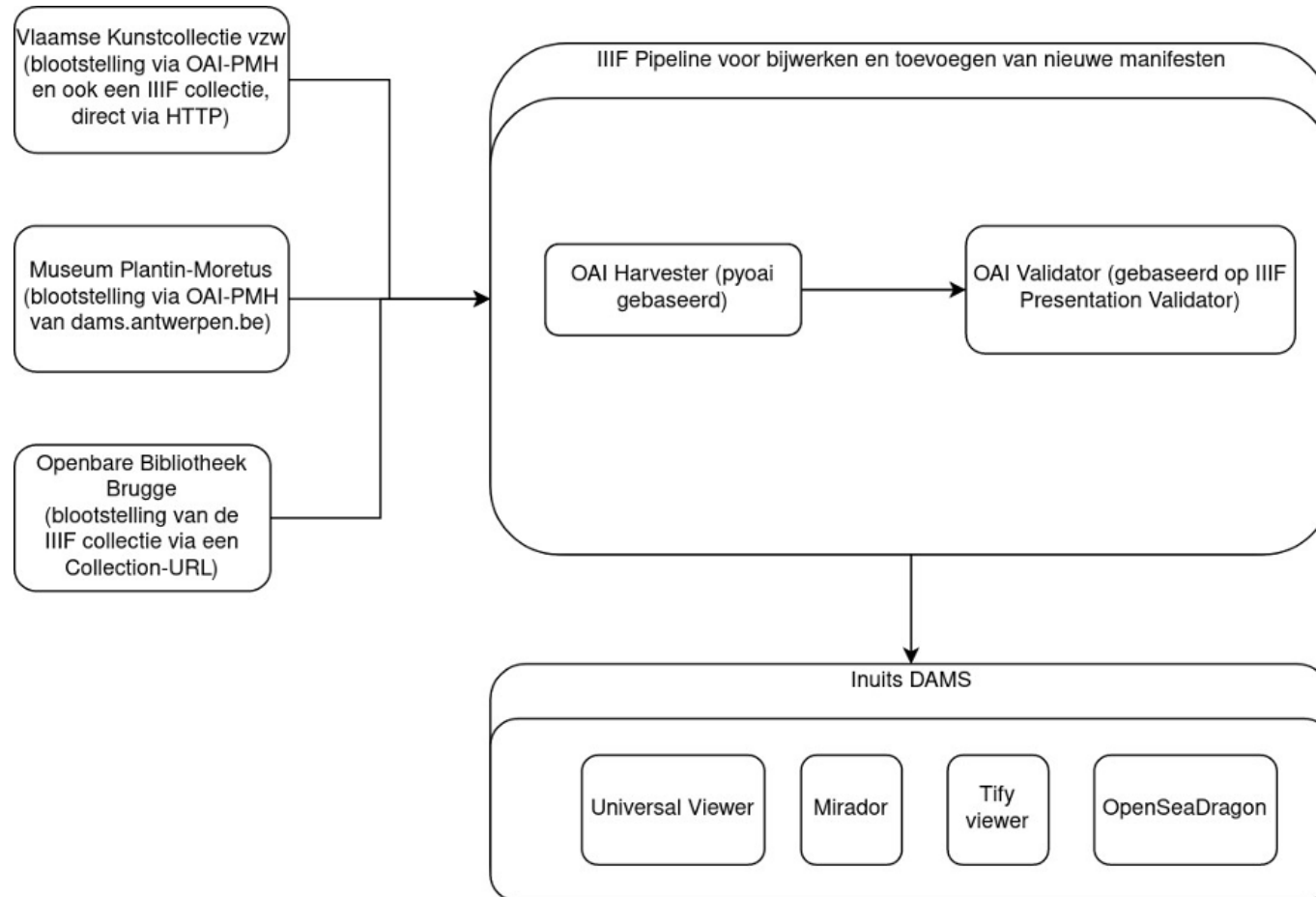
Mirador is een configureerbare, webgebaseerde IIIF-afbeeldingsviewer, waarmee afbeeldingen kunnen worden geannoteerd en vergeleken met afbeeldingen van digitale collectiehouders verspreid over de hele wereld. Net als bij de Universal Viewer is hiervoor enkel de URL van een IIIF-manifest nodig.

- [OpenSeaDragon](#)

Een open-source, webgebaseerde viewer voor zoombare afbeeldingen met hoge resolutie, geïmplementeerd in JavaScript, voor desktop en mobiel. Deze wordt nu reeds ondersteund door Inuits DAMS *(maar werd in de front-end van de proefopstelling niet zichtbaar geïmplementeerd.)*

3. Architectuur

3.1 Grafische voorstelling



3.2 Toelichting bij de belangrijkste softwarecomponenten

We houden er aan om de belangrijkste componenten nog wat uitgebreider toe te lichten.

3.2.1 pyoai

pyoai is een Python-bibliotheek die het toelaat om programma's te schrijven die een OAI-PMH interface consumeren, en de data die hieruit komt verder te verwerken, filteren, uitbreiden etc. Voor de proefopstelling ligt deze bibliotheek aan de basis van de microservice verantwoordelijk voor het ophalen van manifesten van de verschillende projectpartners.

De microservice waarvan pyoai aan de basis ligt, wordt modulair uitgewerkt met uitbreidbaarheid in het achterhoofd. In eerste instantie worden alle projectpartners ondersteund, maar indien nieuwe partners zich bij het project aanbieden, kunnen hun collecties, wanneer deze via een OAI-PMH interface worden aangeboden, toegevoegd worden zonder de implementatie van de microservice aan te passen. Als de collecties van de nieuwe partner via een niet-gestandaardiseerde interface worden aangeboden (dus niet via OAI-PMH), kunnen deze toch relatief makkelijk toegevoegd worden door een modulaire aanpassing van de architectuur van de microservice.

3.2.2 Elody

Elody (voorheen [Inuits DAMS](#)) bestaat uit een aaneenschakeling van microservices, namelijk: de [collection-api](#) (verantwoordelijk voor het opslaan van metadata, en tijdelijk bijhouden van manifesten), de [search-api](#) (verantwoordelijk voor het doorzoeken van de DAMS), de [storage-api](#) (verantwoordelijk voor het opslaan van digitaal materiaal) en de front end (grafische schil bovenop deze microservices waarmee de gebruiker interageert). Op de storage-API na worden al deze bouwstenen gebruikt in de proefopstelling. Binnen de opstelling van de IIF-manifestenbibliotheek hoeft immers bij de beheerder van de bibliotheek geen digitaal materiaal opgeslagen te worden. Het materiaal wordt rechtstreeks opgehaald via de endpoints (in het geval van de VKC de imagehub) op de DAM's van de projectpartners.

Deze component voorzag bij de start van het project reeds een groot deel van de gewenste functionaliteiten. Enkele functionaliteiten dienden echter nog verder doorontwikkeld te worden in functie van dit project. Dit gaat dan vooral over het diepgaand ondersteunen van IIF-manifesten, en alles wat daarmee verbonden is, in plaats van 'simpele' mediabestanden in het systeem. Ook worden meerdere IIF-viewers toegevoegd.

3.2.3 Mirador

Mirador wordt, naast afbeeldingsviewer, ook gebruikt om de exploratie van manifesten door gebruikers van de bibliotheek te faciliteren. Dit onder meer door een leeg (on gevuld) Mirador canvas weer te geven waar de gebruikers de in de manifestenbibliotheek geselecteerde manifesten kunnen toevoegen om deze naast elkaar te openen.

4. Werkwijze

4.1 Ophalen van manifesten bij partners

De projectpartners stellen elk hun manifesten op een verschillende manier ter beschikking: via een OAI-PMH endpoint (VKC), via een endpoint van DAMS Antwerpen (Museum Plantin-Moretus) en via een IIF-collectie (Openbare Bibliotheek Brugge).

Er diende een component te worden gebouwd, met aan de basis pyoai, voor het ophalen van de ter beschikking gestelde manifesten bij de partnerorganisaties. Deze manifesten worden bijgehouden in [Elody](#), meer specifiek in de collection-api microservice. Dit om de geavanceerde zoekfaciliteiten van Elody mogelijk te maken en om een snelle bevraging te garanderen. Als er wijzigingen gebeuren aan de manifesten bij de partners, dan worden deze na een bepaalde termijn ook getoond in de manifestenbibliotheek omdat de OAI-PMH-interface de manifesten periodiek telkens opnieuw bevroegt.. Het kan dus tijdelijk voorkomen dat bijgewerkte data in het manifest niet direct beschikbaar zijn in de manifestenbibliotheek.

4.2 Persistentie van manifesten in de bibliotheek

In tegenstelling tot de digitale beelden, die via de tool rechtstreeks uit de DAM's van de partners worden opgeroepen, worden de machineleesbare manifesten van de partners wel in hun geheel bijgehouden in de bibliotheek (en periodiek bijgewerkt, zie supra). Bovendien worden bepaalde metadata op een uniforme manier bewaard. Dit om performant manifesten te kunnen op-/doorzoeken, en om deze metadata op een uniforme wijze te kunnen weergeven in de bibliotheek.

Verder kan ook relevante metadata van buiten de IIF-manifesten worden opgeroepen. Maar hiervoor moet aan twee voorwaarden voldaan worden. Eerst en vooral moeten deze metadata via een `seeAlso`-veld in het manifest worden gekoppeld. Daarnaast moet deze via de `seeAlso`-link ter beschikking gestelde data ook beschikbaar zijn in een ondersteund LIDO/XML of "Linked Data"-formaat, zoals RDF/XML, JSON-LD, turtle, ... Wanneer dat het geval is, kunnen deze data ook zichtbaar worden ingezet in de bibliotheek. De manifesten kunnen dan ook worden opgezocht aan de hand van data gelinkt aan het manifest, maar niet direct gespecificeerd in het manifest.

Niet alle data uit een conform [seeAlso](#)-document zijn evenwel doorzoekbaar en zichtbaar. Alleen de metadata die Elody/Inuits DAMS 'herkent', worden beschikbaar gesteld. Het gaat hier dan over data conform bepaalde ontologieën zoals onder andere [Dublin Core](#).

De andere, niet conforme metadata worden evenwel ook opgeslagen in het systeem en zijn eveneens opvraagbaar. Alleen kunnen ze niet worden doorzocht in de zoekqueries.

Tot dusver is er dieper ingegaan op de JSON-representaties van de manifesten. De verwijzingen naar de effectieve afbeeldingen, opvraagbaar via de IIF image API, of de

thumbnails beschikbaar over HTTP, worden ook direct ingezet. Dit wil dus zeggen dat de afbeeldingen niet opgeslagen/gecacht worden in de bibliotheek, maar, zoals reeds gesteld, direct bevraagd worden aan de bron zoals die gespecificeerd is in de manifesten.

4.3 Search Engine Optimization (SEO)

Niet enkel de doorzoekbaarheid van de IIF-manifestenbibliotheek zelf is van belang. Ook de vindbaarheid van de bibliotheek via een zoekmachine (Google, Bing, Yahoo!) is cruciaal. Daarom gaat bij de opbouw en werking van de bibliotheek ook aandacht naar Search Engine Optimization (SEO).

Aangezien de applicatie een single-page applicatie is die vrijwel volledig wordt opgebouwd en aangestuurd via Javascript, is het vaak moeilijk voor zoekmachines om deze applicaties correct te interpreteren en indexeren.

Om de data toch makkelijk indexeerbaar te maken voor zoekmachines wordt er daarom bij iedere request gekeken naar de client die deze doet. Wanneer de webserver (NGinx) herkent dat het om een zoekmachine of sociale media site (duckduckgo, google, facebook, x...) gaat, geeft de webserver niet de standaard applicatie terug, maar een geoptimaliseerde pagina voor zoekmachines.

Op deze pagina staat rechtstreeks de metadata in machine-interpreteerbare vorm, zonder dat er Javascript-code voor moet uitgevoerd worden. Deze pagina is ook voorzien van Open Graph Meta Tags zodat sociale media een correcte preview (met afbeelding) kunnen tonen.

Verder is het ook mogelijk om alle data als Linked Open Data (LOD) op te vragen, bijvoorbeeld in RDF-formaat. Dit zijn formaten die specifiek zijn voorzien om machineleesbaar te zijn.