

Technische documentatie



Data & Design Alliance

05/06/2026

David Hof
Nikki van der Haar
Rayan Benlakdame
Mees Elshout
Ehab Radwan



Technisch Overdrachtsrapport: Applicatie Hybride Wereld

Technische overdracht voor gemeente/Stadsarchief Rotterdam

Project	IDP Hybride Wereld - Data Design Alliance
Doelgroep	Opvolgend technisch team, beheerders en projectoverdracht
Versie	Bijgewerkte overdrachtsversie
Datum	5 juni 2026
Status	Eindkopie voor overdracht; externe contact- en deploymentgegevens nog aan te vullen

Kernsamenvatting: De applicatie is een mobiel-gerichte React/Vite webapp waarin leerlingen via Jan, een fictieve binnenvaartschipper uit 1694, een route door historisch Rotterdam volgen. De huidige hoofdroute bestaat uit Laurenskerk, Schielandshuis en Fransche Kerk. Per locatie wordt via QR, challenge/game, hartje en 3D-model een stadsfragment vrijgespeeld.

Leeswijzer

Dit rapport is bedoeld als praktische overdracht. Het beschrijft wat de applicatie doet, hoe de code technisch is opgebouwd, welke onderdelen beheer vragen en welke stappen nodig zijn om lokaal te draaien of verder te ontwikkelen.

- Lees hoofdstuk 2 en 3 voor de globale flow en architectuur.
- Lees hoofdstuk 6 en 7 voor de game-, unlock-, opslag- en analyticslogica.
- Lees hoofdstuk 8 voor installatie en overdracht aan een nieuw technisch team.
- Lees hoofdstuk 9 en 10 voor beperkingen, risico-s en doorontwikkeling.

Contents

Leeswijzer	1
1. Projectcontext en doel	3
2. Applicatieflow en gebruikersroute	4
3. Technische architectuur	5
4. Routes en belangrijkste componenten	6
5. Datamodel, content, assets en 3D-modellen.....	7
6. Game-, QR- en unlock-flow.....	8
7. Opslag, analytics en survey	9
8. Installatie, draaien en bouwen.....	10
9. Bekende beperkingen en onderhoudsrisico-s.....	11
10. Roadmap voor overdracht en doorontwikkeling.....	12
Bijlage A. Belangrijkste bestanden	13

1. Projectcontext en doel

Hybride Wereld is ontwikkeld als een interactieve, mobiel-gerichte webapplicatie voor het verkennen van historisch Rotterdam. De app combineert een verhaallijn, locatiepagina-s, QR-interactie, korte educatieve games, hartjesbeloningen en 3D-modellen van gebouwen.

De gebruiker volgt Jan, een fictieve binnenvaartschipper uit 1694. Vanuit zijn perspectief worden historische plekken in Rotterdam gekoppeld aan opdrachten voor leerlingen uit de bovenbouw van de basisschool. De route is ontworpen als speelse erfgoedervaring: ontdekken, scannen, puzzelen, vrijspelen en reflecteren.

Overdrachtsstatus: De software is in deze fase grotendeels client-side en geschikt als expo-/pilotapp. Voor structurele publieke productie zijn vooral hosting, privacy/analytics-afspraken, contentbeheer en QA-processen nog belangrijk.

2. Applicatieflow en gebruikersroute

De huidige applicatie start met een verhalende onboarding en leidt daarna naar een kaartomgeving. Vanuit de kaart bezoekt de gebruiker de route-locaties. Elke hoofdlocatie heeft een eigen challenge-flow die na afronding een hartje en het 3D-model vrijspeelt.

1. IntroStory introduceert Jan en de aanleiding van de reis.
2. Home toont de routekaart, voortgang en beschikbare locaties.
3. De gebruiker opent een locatiepagina via /location/:id.
4. Bij route-locaties moet eerst een correcte QR-code worden gescand of gedemonstreerd.
5. Daarna start de locatiechallenge: puzzelgame, timelinegame of waterbeheer-game.
6. Na winst worden localStorage keys gezet voor unlocked status en hartje.
7. De 3D-viewer toont het vrijgespeelde gebouwmodel.
8. Na voltooiing van alle drie hoofdlocaties volgt de route naar outro en survey.

Route-locatie	Challenge	Beloning/resultaat
Laenskerk	Custom puzzelgame rond het oplossen van een kaart/puzzel.	Laenskerk-hartje en /Laenskerk.glb model vrijgespeeld.
Schielandshuis	Waterbeheer-balansspel waarin de speler het waterpeil tussen kades beheert.	Schielandshuis-hartje en /Schielandshuis.glb model vrijgespeeld.
Fransche Kerk	Timelinegame rond historische volgorde en context.	Fransche Kerk-hartje en /Fransche kerk.glb model vrijgespeeld.

3. Technische architectuur

De applicatie is een Single Page Application op basis van React en Vite. De belangrijkste runtime-logica draait in de browser. Er is geen centrale productiebackend voor content, authenticatie of voortgang. De optionele analytics-server schrijft alleen lokaal events weg naar een JSON-bestand.

Laag	Techniek	Rol in de applicatie
Frontend	React 18, Vite 6, React Router	Componentgebaseerde webapp met client-side routing.
Styling/UI	Tailwind CSS, Radix UI, Framer Motion, Lucide	Mobiele interface, overlays, animaties en iconen.
3D	Three.js, @react-three/fiber, @react-three/drei	GLB-modellen renderen op locatiepagina-s, home en favorieten.
QR	html5-qrcode	Camera/QR-scan voor route-locaties en demo-scan flow.
Voortgang	localStorage	Hartjes, unlocks, tutorialstatus, vragenlimieten en routestatus.
Analytics	Express server.js + app-events.json	Optionele lokale eventopvang op poort 3001.
Survey	Supabase client	Optionele enquête na afloop via tabel survey_responses.

- De Vite-config kopieert Cesium statische bestanden naar /cesiumStatic, maar Cesium is in de huidige app geen kernlaag.
- 3D-modellen en afbeeldingen staan hoofdzakelijk in public/ en worden door Vite als statische assets geserveerd.
- De app gebruikt een mix van TypeScript (.tsx) en JavaScript/JSX (.jsx).

4. Routes en belangrijkste componenten

De router staat in `src/app/routes.ts`. De root-route gebruikt `Root` als gedeelde layout met navigatie en audio; de child-routes tonen de pagina-s.

Route	Component	Functie
/	IntroStory	Startverhaal en onboarding.
/home	Home	Kaart, routevoortgang, locatie-unlocks en route-overzicht.
/location/:id	LocationDetail	Locatiepagina met QR, game-flow, chat, 3D-model en beloning.
/favourites	Favourites	Catalogus/overzicht van vrijgespeelde 3D-modellen.
/hearts	Hearts	Hartjesverzameling.
/quests	Quests	Missie- en voortgangsoverzicht.
/search	Search	Zoekpagina voor locaties.
/saved	Saved	Opgeslagen locaties.
/settings	Settings	Karakter-/gebruikersinstellingen.
/scan	Scan	Scanpagina/placeholder naast de locatiegebonden QR-flow.
/outro	OutroStory	Afsluitend verhaal na de route.
/survey	Survey	Eindvragenlijst via Supabase.

Let op: `Root.tsx` bevat de onderste navigatie en audio-loop. `LocationID.tsx` bevat relatief veel game-, QR-, event- en unlocklogica. Bij onderhoud is dat bestand de belangrijkste plek om voorzichtig te refactoreren.

5. Datamodel, content, assets en 3D-modellen

De primaire locatiedata staat in `src/app/data/locatie.ts`. Daarin staat de `Location`-interface en de `rotterdamLocations`-array. Voor de drie hoofdlocaties worden aanvullende historische facts geladen uit tekstbestanden in `src/app/data/facts/`.

Onderdeel	Locatie in repo	Beheeropmerking
Locatiedata	<code>src/app/data/locatie.ts</code>	Naam, beschrijving, kaartpositie, historische velden en tijdlaagdata.
Historische facts	<code>src/app/data/facts/*.txt</code>	Losse regels worden willekeurig gekozen als historische context.
Gamecomponenten	<code>src/app/components/pages/*.jsx</code>	PuzzleGame, TimelineGame en SchielandWaterGame.
3D-modellen	<code>public/*.glb</code>	Gebouwmodellen: Laurenskerk, Schielandshuis en Fransche Kerk.
Jan/intro/outro assets	<code>public/characters</code> , <code>public/Intro</code> , <code>public/Outro</code>	Verhaalbeelden en karakterbeelden.
Game-assets	<code>public/Game1</code> , <code>public/Timeline</code>	Puzzel-, beloning- en timelinebeelden.

De `Location`-interface bevat velden zoals `id`, `name`, `description`, `x/y` kaartpositie, `image`, `modellImage`, `yearBuilt`, `architect`, `funFact`, historische gidsvelden en `historicalData` per jaar. De huidige hoofdroute gebruikt vooral de drie route-locaties; oudere of experimentele data kan nog in de dataset aanwezig zijn en moet bij publicatie worden opgeschoond of duidelijk gelabeld.

6. Game-, QR- en unlock-flow

Route-locaties zijn geconfigureerd in ROUTE_GAME_CONFIGS in LocationID.tsx. Deze configuratie koppelt per locatie het modelpad, de localStorage keys, quiztekst, dossierfiles, correcte code en beloningstekst.

Locatie	Modelpad	Unlock key	Heart key	Challenge
Laurenskerk	/Laurenskerk.glb	laurenskerk-unlocked	laurenskerk-heart	QR -> custom puzzelgame -> unlock.
Schielandshuis	/Schielandshuis.glb	schielandshuis-unlocked	schielandshuis-heart	QR -> waterbeheer-balansspel -> unlock.
Fransche Kerk	/Fransche kerk.glb	fransche-kerk-unlocked	fransche-kerk-heart	QR -> timelinegame -> unlock.

- QR-success zet <locationId>-qr-scanned op true en opent de volgende stap voor die locatie.
- Bij winst logt de app location_unlocked, heart_collected en route_progress_updated.
- De homepagina gebruikt de hartjesstatus om de voortgang en volgende routeplek te tonen.
- Bij volledige route kan de app naar outro en survey navigeren.

Onderhoudsrisico: De locatieflows zijn functioneel maar hardcoded. Als er meer gebouwen of speltypes bijkomen, is een apart configuratiebestand of flow-service aan te raden.

7. Opslag, analytics en survey

Gebruikersvoortgang staat in localStorage. Dit is eenvoudig en werkt goed voor demo/pilot, maar is browser- en devicegebonden. Er is geen centraal gebruikersaccount en geen server-side voortgangsopslag.

Key of bestand	Type	Betekenis
laurenskerk-unlocked / -heart	localStorage boolean-string	Vrijgespeelde status en hartje voor Laurenskerk.
schielandshuis-unlocked / -heart	localStorage boolean-string	Vrijgespeelde status en hartje voor Schielandshuis.
fransche-kerk-unlocked / -heart	localStorage boolean-string	Vrijgespeelde status en hartje voor Fransche Kerk.
latest-unlocked-location	localStorage string	Laatste vrijgespeelde routeplek voor home-progress popup.
home-last-seen-hearts	localStorage number-string	Aantal hartjes dat Home eerder heeft gezien.
home-tutorial-seen	localStorage boolean-string	Of de home tutorial al is getoond.
tijdgids-vragen-<locationId>	localStorage number-string	Maximaal 2 vragen aan de tijdgids per locatie.
app-events	localStorage JSON array	Lokale browserkopie van analytics-events.
app-events.json	JSON bestand	Lokale serverkopie als server.js draait.

Analytics-events worden via saveAppEvent eerst in browser localStorage opgeslagen en daarna geprobeerd te posten naar <http://localhost:3001/save-event>. Als de analytics-server niet draait, blijft de app doorwerken; alleen de serverkopie ontbreekt.

De survey gebruikt Supabase via `src/supabaseClient.js` en schrijft naar `survey_responses`. Voor overdracht moet de eigenaar controleren of de gebruikte Supabase-configuratie, RLS-instellingen en privacy-afspraken passen bij productiegebruik. Neem geen `service_role` of geheime sleutel op in frontendcode.

8. Installatie, draaien en bouwen

De applicatie kan lokaal worden gedraaid met Node.js. Gebruik bij voorkeur Node.js 18 of hoger. Python is alleen nodig voor data-/3D-verwerkingsscripts en niet voor het draaien van de webapp.

Stap	Commando	Resultaat
Dependencies installeren	<code>npm install</code>	Installeert Node dependencies uit package-lock.json.
Development server	<code>npm run dev</code>	Start Vite, standaard op <code>http://localhost:5173</code> .
Productiebuild	<code>npm run build</code>	Bouwt dist/ met statische productie-assets.
Build preview	<code>npm run preview</code>	Serveert de productiebuild lokaal.
Analytics server optioneel	<code>node server.js</code>	Start lokale eventserver op <code>http://localhost:3001</code> .
Analytics dashboards optioneel	<code>streamlit run analytics/dashboard.py</code>	Toont app-events.json in Streamlit, afhankelijk van Python/Streamlit-installatie.

Minimale lokale startvolgorde voor demo met analytics:

```
npm install
node server.js
npm run dev
```

Aan te vullen: Repository-URL, definitieve hosting/deploymentlocatie, technisch contactpersoon, eigenaar van Supabase-project en link naar designs/Figma zijn niet betrouwbaar uit de repo af te leiden en moeten door het projectteam worden ingevuld.

9. Bekende beperkingen en onderhoudsrisico-s

- Geen test-suite aanwezig; regressies moeten nu vooral handmatig worden getest.
- Voortgang en unlocks staan in localStorage en zijn dus niet centraal of device-overstijgend.
- QR/game-flow en veel eventlogica zitten geconcentreerd in LocationID.tsx.
- localStorage-sleutels zijn niet gecentraliseerd in een constants-bestand.
- Analytics-server is lokaal en optioneel; dit is geen productiebackend.
- Survey-configuratie staat in frontendcode en moet voor productie/privacy opnieuw worden beoordeeld.
- Sommige documentatie en oudere data/componentvarianten in de repo kunnen achterhaald zijn.
- Er is geen strikte TypeScript-config zichtbaar; de app gebruikt ook meerdere JSX-bestanden.
- 3D- en assetpaden zijn gevoelig voor exacte bestandsnamen, inclusief spaties en hoofdletters.

10. Roadmap voor overdracht en doorontwikkeling

Prioriteit	Aanbevolen actie	Waarom
Hoog	Bepaal productiehosting, domein en privacy/analyticsbeleid.	Nodig voor overdracht aan externe beheerder en publiek gebruik.
Hoog	Centraliseer route/game-config en localStorage keys.	Maakt uitbreiding naar meer gebouwen veiliger.
Hoog	Voeg smoke-tests toe voor route, QR, unlocks en build.	V voorkomt regressies in de kernflow.
Middel	Schoon verouderde docs, assets en alternatieve componenten op.	Verlaagt onderhoudslast voor nieuw team.
Middel	Maak contentbeheer losser van code, bijvoorbeeld via CMS of gestructureerde JSON.	Stadsarchief/gemeente kan content makkelijker beheren.
Middel	Documenteer QR-codes en fysieke plaatsing per locatie.	Belangrijk voor beheer op locatie.
Laag	Onderzoek PWA/offline ondersteuning.	Handig voor veldgebruik, maar niet vereist voor demo-overdracht.

Bijlage A. Belangrijkste bestanden

Bestand/map	Functie
package.json	Dependencies en npm scripts.
vite.config.ts	Vite, React, Tailwind en Cesium static copy configuratie.
src/app/routes.ts	Client-side routes.
src/app/components/pages/home.tsx	Hoofdkaart, voortgang, unlockgating en routeweergave.
src/app/components/pages/LocationID.tsx	Locatiepagina, QR, game-flow, unlocks, 3D-viewer en eventlogging.
src/app/components/pages/SchielandWaterGame.jsx	Schielandshuis waterbeheer-game.
src/app/components/pages/PuzzleGame.jsx	Laurenkerk custom game.
src/app/components/pages/TimelineGame.jsx	Fransche Kerk timelinegame.
src/utills/appEvents.ts	Eventmodel, lokale opslag en POST naar analytics-server.
server.js	Optionele lokale Express analytics-server.
src/supabaseClient.js	Supabase client voor survey; controleer productieconfig.
public/	Statische assets, GLB-modellen, Jan-beelden, intro/outro en gamebeelden.