

INNER Project - Overdracht

Documentatie

Inhoudsopgave

1. Projectoverzicht

- Projectplan
- Samenvatting van het project
- Architectuuroverzicht

2. Sprintdocumentatie

- Sprintoverzichten
- Definitie van Done (DoD)
- Retrospectives

3. Technische Documentatie

- Handleiding
- Testplan en Testresultaten
- Code Documentatie

4. Gebruikersdocumentatie

- Handleiding voor eindgebruikers
- Training en Support

5. Data en Opgeslagen Materialen

- Dataset Overzicht
- Beheerrichtlijnen

6. Overdrachtsrapport

- Samenvatting van de huidige status
- Actiepunten
- Risico's en uitdagingen

7. Aanbevolen Bijlagen

- Video of presentatie
- Feedbacklogs
- Toegangsinformatie

1. Projectoverzicht

Projectplan

Het INNER-project richt zich op het ontwikkelen van een applicatie waarmee gebruikers de kwaliteit van EV-batterijen kunnen beoordelen met behulp van röntgenbeelden. Het project volgt de Scrum-methodologie.

Bijlagen:

- [INNER-ProjectPlanning](#)

Samenvatting van het project

Doelstellingen:

- Handmatige inspectie van EV-batterijen ondersteunen.
- Gebruik maken van een ML-model om defecten te identificeren.
- Ontwikkelen van een schaalbare en gebruiksvriendelijke applicatie.

Architectuuroverzicht

Een overzicht van de technische infrastructuur, waaronder:

- **Azure Blob Storage:** Voor opslag van röntgenbeelden.
- **SQLite:** Voor metadata.
- **ML-componenten:** Voor automatische foutdetectie.

Bijlagen:

- [C4 Context Diagram Met ML](#)
- [C4 Context - Relaties Diagram Zonder ML](#)
- [UML Componenten Diagram](#)
- [UML Class Diagram Met ML](#)
- [UML Class Diagram Zonder ML](#)
- [Usecase Diagram](#)

2. Sprintdocumentatie

Sprintoverzichten

Sprint 1

In de eerste sprint is een requirementsanalyse opgesteld om een duidelijk beeld te krijgen van de functionele eisen van het systeem. Dit document is doorgestuurd naar de stakeholder voor feedback. Zodra de feedback verwerkt is en akkoord is gegeven, zullen acceptatiecriteria worden opgesteld. Deze analyse vormt de basis voor de verdere ontwikkeling van het systeem.

- **Bart:** Zorgde voor de eerste requirementsanalyse, verzamelde feedback van de stakeholders, en hielp bij het opstellen van acceptatiecriteria.
- **Piotr:** Heeft het project leren kennen, een projectplan opgesteld, en onderzoek gedaan naar machine learning. Daarnaast heeft hij een GitHub-organisatie opgezet en een CI/CD-pijplijn geïmplementeerd.
- **Stefan:** Heeft wensen en eisen van stakeholders verzameld, frameworks gekozen en de eerste ontwerpen gepresenteerd.
- **Wally:** Heeft de projectstructuur geanalyseerd, ADR-documenten opgesteld, en een CI/CD-pijplijn opgezet.

Sprint 2

Tijdens deze sprint werd gewerkt aan de feedback van de stakeholder op de eerder opgestelde requirementsanalyse. Op basis hiervan zijn acceptatiecriteria opgesteld en userstories geformuleerd. Deze userstories zijn toegevoegd aan het GitHub-projectboard om structuur te bieden in het ontwikkelproces. Daarnaast is een UI/UX Design Document opgesteld. Hierin zijn richtlijnen vastgelegd voor het ontwerp, inclusief onderzoek, wireframes en proof-of-concepts. Dit document diende als leidraad voor het verdere ontwerp en ontwikkeling.

- **Bart:** Verwerkte stakeholderfeedback, formuleerde acceptatiecriteria en userstories, en voegde deze toe aan het GitHub-projectboard. Stelde een UI/UX Design Document op als leidraad voor ontwerp en ontwikkeling.
- **Piotr:** Voerde verder onderzoek uit naar ML-modellen, ontwikkelde prototypes, en werkte aan automatische deployment naar Azure.
- **Stefan:** Onderzocht UI-mogelijkheden en best practices, en testte TIFF-bestanden in React.
- **Wally:** Richtte zich op de analyse en implementatie van Azure Blob Storage en documentatie.

Sprint 3

Deze sprint richtte zich op het maken van prototypes en het uitwerken van de gebruikersinterface. Op basis van wireframes is een eerste versie van de interface gebouwd in React. Dit prototype is gepresenteerd tijdens de sprintdemo. Feedback vanuit de demo zal in de volgende sprint worden verwerkt. Daarnaast is onderzoek gedaan naar bibliotheken voor medische beeldanalyse in React. Dit bleek complexer dan verwacht. Uiteindelijk is een Proof-of-Concept (P.O.C.) ontwikkeld met Cornerstone, waarmee notities en metingen op afbeeldingen gemaakt kunnen worden. De volgende stap is het opslaan van deze annotaties in de backend en het opnieuw laden om deze te bekijken.

- **Bart:** Werkte aan prototypes en wireframes, presenteerde deze tijdens de sprintdemo en verwerkte de ontvangen feedback. Onderzocht medische beeldanalysebibliotheken en ontwikkelde een Proof-of-Concept met annotatiemogelijkheden.
- **Piotr:** Ontwikkelde en trainde ML-modellen, creëerde demo's, en stelde een adviesdocument op.
- **Stefan:** Werkte verder aan applicatieontwerpen, navigatiestructuur, en integratie van frontend en database.
- **Wally:** Optimaliseerde Blob Storage, implementeerde een CDN, en stelde testplannen op.

Sprint 4

De vierde sprint richtte zich voornamelijk op de technische implementatie. De tijd werd besteed aan het ontwikkelen van het rapportagescherm en het bouwen van een samenhangende flow. Van het inladen van afbeeldingen tot het annoteren, genereren van rapporten en de verwerking door de ML-laag – alle onderdelen werden geïntegreerd tot een werkend geheel. Omdat het teamlid dat verantwoordelijk was voor de backend onverwacht was uitgevallen, heb ik de volledige backend opgezet. Dit omvatte het bouwen van de API's, de verbinding met de database en het verzorgen van de integratie met zowel de frontend als de ML-laag.

- **Bart:** Richtte zich op de technische implementatie van het rapportagescherm en zorgde voor een samenhangende flow tussen frontend, annotaties, en backendverwerking. Hielp met backendontwikkeling vanwege uitval van een teamlid.
- **Piotr:** Realiseerde de ML Layer-applicatie en documenteerde deze.
- **Stefan:** Ontwikkelde pagina's voor batterijbeheer en rapportage.
- **Wally:** Integreerde backend en frontend, implementeerde JWT-authenticatie, en verfijnde de Docker-omgeving.

Bijlagen:

- [INNER-Testplan en Teststrategie](#)
- [Werkinstructie voor Integratie- en Performance-Tests INNER](#)

Definitie van Done (DoD)

~~Unit test dekking van 100%.~~

- Functioneel gereviewed en goedgekeurd.
- Volledig gedocumenteerd en getest in een staging-omgeving.

3. Technische Documentatie

Handleiding

De handleiding bevat uitgebreide instructies voor installatie en configuratie, inclusief de volgende onderwerpen:

- **Docker-setup:** Richtlijnen voor het opzetten van een werkende containeromgeving.
- **Azure Blob Storage:** Gedetailleerde stappen voor het configureren en integreren met SQLite.
- **GitHub to Azure deployment:** Instructies voor het configureren van GitHub en Azure voor automatische deployment.
- **Documentatieoverzicht:** Instructies voor toegang tot projectdocumentatie en het begrijpen van de structuur.
- **Documentatieoverzicht:** Instructies voor toegang tot projectdocumentatie en het begrijpen van de structuur.

Bijlagen:

- [INNER Monorepo Setup](#)
- [INNER Monorepo Werkinstructie](#)
- [INNER-Docker Deployment Handleiding](#)
- [INNER-Dockerfiles](#)
- [Handleiding Blob INNER](#)

- [Azure Deployment](#)
 - [Reader's Guide voor INNER Blob Documentatie](#)
 - [Database structuur](#)
-

CI/CD-documentatie

De CI/CD-pijplijn waarborgt een naadloze integratie en levering via geautomatiseerde processen:

1. Automatische tests:

- Uitvoeren van unit- en integratietests bij elke commit.

2. Buildprocessen:

- Automatische builds van backend en frontend.

3. Deployment:

- (Semi-)automatische implementatie naar Azure-omgevingen.

Bijlagen:

- [INNER-Complete GitHub Actions Workflow](#)
 - [INNER-Software Architectuur en CI-CD-Documentatie](#)
 - [INNER-CI-CD Pijplijn Onderhoudshandleiding](#)
 - [Azure Deployment](#)
-

Testplan en Testresultaten

In deze sectie wordt een overzicht gegeven van uitgevoerde tests:

• **Prestaties:**

- Analyse van Azure Blob Storage en CDN-integratie.

• **Bekende problemen:**

- Technische schulden en bugs.

• **Aanbevelingen:**

- Verbeteringen voor schaalbaarheid en stabiliteit.

Bijlagen:

- [INNER-ADR-ObjectStorage](#)
 - [INNER-Testplan en Teststrategie](#)
 - [02-4 INNER Blob Adviseren](#)
 - [03-1 INNER Blob Design](#)
 - [Azure Blob Storage Testresultaten](#)
-

Code Documentatie

ADR's beschrijven de belangrijkste beslissingen en de achterliggende overwegingen voor het project:

• **Technische keuzes:**

- CI/CD-pijplijnen en frontend-backendintegratie.

Bijlagen:

- [INNER-ADR-GitHub](#)
 - [INNER-ADR-DataStorage](#)
 - [INNER-ADR-Monorepo](#)
 - [ML Layer Design](#)
 - [UI ontwerp wireframes](#)
-

4. Gebruikersdocumentatie

Handleiding voor eindgebruikers

Deze handleiding biedt richtlijnen over het gebruik van het systeem:

1. Röntgenbeelden inspecteren en manipuleren:

- Functionaliteiten zoals roteren, inzoomen, en markeren van specifieke foutgebieden zijn beschikbaar.
- Gebruikers kunnen defecten identificeren en markeren.

2. Rapportage genereren:

- Bevindingen worden samengevoegd tot een rapport, inclusief input van de gebruiker en suggesties van het ML-model.

3. AI-ondersteuning:

- Het systeem maakt gebruik van een ML-model dat automatisch defecten detecteert en kritieke problemen signaleert.

Bijlagen:

- [Handleiding Blob INNER](#)
 - [Handleiding Web applicatie](#)
-

Training en Support

Om de gebruikerservaring te optimaliseren:

- **Training:**
 - Eindpresentatie over het gebruik van de applicatie.
 - Toelichting op het genereren van rapporten.
- **Support:**
 - Beschikbaarheid van ondersteuning bij technische problemen bespreken.

Bijlagen:

- [Reader's Guide voor INNER Blob Documentatie](#)
-

Hier is de aangepaste versie van het document met een nieuwe sectie voor **Operational Costs Mapping** toegevoegd onder "Data en Opgeslagen Materialen":

5. Data en Opgeslagen Materialen

Dataset Overzicht

Een overzicht van de opgeslagen data:

- **Röntgenbeelden:** Data van ongeveer 1GB per batterij, opgeslagen in Azure Blob Storage.
- **Metadata:** Specificaties van batterijen opgeslagen in SQLite.

Bijlagen:

- [Azure Blob Storage Testresultaten](#)

Beheerrichtlijnen

1. Lifecycle management:

- Implementatie van verschillende tiers (Hot, Cool, Archive) voor kostenoptimalisatie.

2. Beveiliging:

- Toegang wordt geregeld via RBAC.
- Auditlogs voor tracking van wijzigingen.

3. Toekomstige schaalbaarheid:

- CDN-integratie voor prestatieverbetering bij groeiende datasets.

Operational Costs Mapping

1. Kosten per opslagoptie:

- **Hot Tier:** Geschikt voor data die vaak wordt opgevraagd.
- **Cool Tier:** Voor minder vaak gebruikte data, met lagere kosten maar hogere toegangskosten.
- **Archive Tier:** Voor langdurige opslag met minimale toegang.

2. Kostenoverwegingen:

- **Röntgenbeelden:**
 - Hot Tier voor recente uploads en actieve verwerking.
 - Cool of Archive Tier voor historische data.
- **Metadata:**
 - SQLite geoptimaliseerd voor snelle query-respons.

3. Kostenbeheersing:

- Monitoring van opslagkosten met Azure Cost Management.
- Automatische verplaatsing van data tussen tiers met lifecycle policies.

4. Integratie met budgetbeheer:

- Regelmatige kostenanalyse.
- Aanbevelingen voor kostenreductie gebaseerd op gebruikspatronen.

Bijlagen:

- [Azure Blob Storage Guide](#)
-

6. Overdrachtsrapport

Samenvatting van de huidige status

- Applicatie is klaar voor productie.

Actiepunten

1. Toekomstige integratie van nieuwe data:

- Continue training van het ML-model.

2. Performance optimalisatie:

- Regelmatige updates aan CDN-configuraties.
- Performance optimalisatie in de backend bij het verwerken van de .Tif bestanden

3. Uitbreiden rapportage features:

- De rapportage tool mist de feature om een gemaakt rapport op te slaan als PDF
- De rapportage tool kan worden uitgebreid met features naar behoefte

Risico's en uitdagingen

- Beheersing van opslagkosten met groeiende datasets.
- Continuïteit in AI-ondersteuning bij minder trainingsdata.

Bijlagen:

- [ML Model Advies](#)
-

7. Aanbevolen Bijlagen

1. [INNER Azure Blob Storage Inrichting](#)
2. [INNER Testplan en Resultaten](#)

3. INNER Beveiligingsrichtlijnen
4. Automatische Documentatie met Swagger UI en ReDoc
5. ML Model Onderzoek