

Software Design Document (SDD)

Eindopdracht Gemeente Almere – Software Development Project



Luuk Steltenpool, Jesper Meuzelaar, Ryan Dürr

Projectleider: Hans Pieters

Gemeente Almere

7-5-2025, versie 1

Voorwoord

Voeg hier (op het eind) een management samenvatting toe.

Versie's

Versie	Auteur	Datum	Veranderingen
1.1	Ryan	7-5-2025	Start van SDD

Inhoudsopgave

Voorwoord	2
Versie's	2
1. Inleiding.....	5
1.1. Doel	5
1.2. Project definitie	5
1.3. Product Visie	5
1.4. Scope	5
1.5. Stakeholders	6
2. Gebruikerseisen	6
3. Architectuuroverzicht	6
3.1. Technische eisen	6
3.2. Algemene architectuur	7
3.3. Technologieën en frameworks.....	7
4. Gedetailleerd Ontwerp	7
4.1. Interface-ontwerp	7
4.2. Datastromen.....	7
4.3. Datamodel	8
4.4. Module-overzicht.....	8
5. Implementatie Details (optioneel)	8
5.1. Programmeertaal en coding standaarden.....	8
5.2. Ontwerpprincipes	8
5.3. Beveiliging en authenticatie.....	8
6. Testing en Validatie.....	8
6.1. Teststrategie.....	8
6.2. Prestatie-eisen	8
6.3. Foutafhandeling en logging (CI/CD).....	8
7. Deployments en Onderhoud (optioneel).....	8
7.1. Deployment strategie.....	8
7.2. Monitoring en onderhoud.....	9
7.3. Schaalbaarheid	9
8. Globale planning.....	9
9. Bronnen	9
10. Bijlagen	9

1. Inleiding

1.1. Doel

Beschrijf het doel van dit document en voor wie het bedoeld is.

De Lief & Leed Pot ondersteunt medewerkers bij bijzondere gelegenheden zoals jubilea, ziekte of pensionering. Momenteel verloopt het proces handmatig, wat inefficiënt is. Dit project beoogt het proces te automatiseren, inclusief:

- Digitale aanvraagverwerking
- Automatische controle op basis van leeftijd en dienstjaren
- Integratie met Mollie voor uitbetalingen

Het doel van dit document is om dit project te documenteren zodat de opdrachtgever een duidelijk beeld heeft van het product.

1.2. Project definitie

Dit project is opgezet in samenwerking met de personeelsvereniging van Gemeente Almere om het proces rondom de Lief & Leed Potje te digitaliseren. Medewerkers kunnen maandelijks vrijwillig bijdragen aan deze pot, waarvan bij bijzondere gebeurtenissen een geldbedrag beschikbaar wordt gesteld. Dit project gaat ervoor zorgen dat de aanvraag van het geldbedrag automatisch verloopt, waardoor de administratieve lasten afnemen en efficiëntie toeneemt.

1.3. Product Visie

Hoe gaat het product er uit zien en wat zal de werking zijn? Waarom wordt de software ontwikkeld en welke waarde biedt de software als het af is?

1.4. Scope

Beschrijf de grenzen van het project, de belangrijkste functies en de context waarin het systeem zal opereren.

In scope:

- Digitaal aanvraagportaal
- Automatische controles (leeftijd, dienstjaren)
- Handmatige beoordeling bij ziekte
- Integratie van betaling via rekeningnummer.

Buiten scope:

- Wijzigingen in contributiestructuur
- Integratie met externe HR-systemen (m.u.v. geboortedata en dienstjaren)

1.5. Stakeholders

- Medewerkers Gemeente Almere
 - HR-afdeling
 - IT-afdeling
 - Financiële administratie
 - Mollie (betalingsverwerker)
-

2. Gebruikerseisen

Beschrijf hier wat het systeem moet doen, inclusief de vereiste functies, gebruikersinteracties en verwachte outputs. Wij noemen dit ook Functional Requirements (FR's).

F1	Gebruikers moeten aanvragen kunnen indienen via een portal
F2	Een automatisch systeem wat automatisch controles voor leeftijd en dienst jubilea verwerkt
F3	Invoeren type gebeurtenis (verjaardag, ziekte, etc..)
F4	Mogelijkheid voor handmatige controle van aanvragen Bijvoorbeeld: ziektegevallen
F5	Betaling moet mogelijk zijn via Mollie
F6	Een rapportage module voor inzicht in uitbetalingen en aanvragen
F7	Er moet een apart beheerder dashboard beschikbaar zijn voor HR
F8	Validatie op basis van geboortedata en dienstjaren
F9	Ziekte aanvragen markeren voor handmatige controle
F10	Mogelijkheid tot opmerkingen plaatsen bij de aanvraag

3. Architectuuroverzicht

3.1. Technische eisen

Lijst en beschrijf de belangrijkste technische eisen en hun verantwoordelijkheden. Wij noemen dit ook wel Non-Functional Requirements (NFR's).

Nf1	Systeem moet beschikbaar zijn via een beveiligde web omgeving	Must	Privacy gevoelige gegevens
NF2	Alleen geautoriseerde medewerkers hebben toegang tot het HR-dashboard	Must	Voor avg compliance
NF3	Gebruikersinterface moet intuïtief zijn	Should	Bevordert Acceptatie
NF4	Systeem moet schaalbaar zijn voor toekomstig gebruik	could	Voor toekomstigheid
NF5	Gegevensopslag moet privacy correct zijn	Must	Het betreft privacygevoelige gegevens
NF6	Minimale laadtijd van paginas: < 2 seconden	Must	Bevordert acceptatie
NF7	Data moet worden opgeslagen in een MySQL database	Must	Maakt de data-opslag makkelijk schaalbaar

3.2. Algemene architectuur

Beschrijf de algemene structuur van het systeem met een software architectuur diagram.

3.3. Technologieën en frameworks

Beschrijf de technologieën en frameworks die worden gebruikt (bijv. programmeertalen, databases, API's).

4. Gedetailleerd Ontwerp

4.1. Interface-ontwerp

Beschrijf API's, UI-Wireframes en/of mockups en andere interfaces die het systeem gebruikt.

4.2. Datastromen

Beschrijf hoe gegevens door het systeem stromen, inclusief sequentiediagrammen of flowcharts.

4.3. Datamodel

Definieer de database-structuur en de belangrijkste entiteiten met ER-diagrammen.

4.4. Module-overzicht

Beschrijf de verschillende modules en hun verantwoordelijkheden. Gebruik een Module Dependency Diagram of een klassendiagram

5. Implementatie Details (optioneel)

5.1. Programmeertaal en coding standaarden

Beschrijf de coding conventies en standaarden die worden gevolgd.

5.2. Ontwerpprincipes

Beschrijf design patterns, best practices en SOLID-principes.

5.3. Beveiliging en authenticatie

Beschrijf hoe het systeem beveiligd wordt en welke authenticatie- en autorisatiemechanismen worden gebruikt.

6. Testing en Validatie

6.1. Teststrategie

Beschrijf concreet wat er wel, maar ook niet, getest moet worden. Geef hierbij ook aan welke testtypes (unit tests, integratietests, acceptatietests) het hier betreft.

6.2. Prestatie-eisen

Definieer verwachte prestaties en benchmarks.

6.3. Foutafhandeling en logging (CI/CD)

Beschrijf hoe fouten worden gelogd en afgehandeld.

7. Deployments en Onderhoud (optioneel)

7.1. Deployment strategie

Beschrijf hoe en waar het systeem wordt ingezet (bijv. cloud, on-premise).

7.2. Monitoring en onderhoud

Beschrijf hoe het systeem wordt bewaakt en onderhouden na de livegang.

7.3. Schaalbaarheid

Beschrijf hoe het systeem kan opschalen bij toenemend gebruik.

8. Globale planning

Laat hier in grote lijnen zien wanneer wat af zou moeten zijn (deadlines). Hiermee laat jij jezelf (en je team) ook zien hoeveel tijd er is om iets af te krijgen. Dit kun je ook gebruiken in je scrumboard. De epics en userstories zet je in een los document of voeg je toe als bijlage.

9. Bronnen

Voeg hier een lijst van APA-bronnen toe. Zie Verwijzingen → Bronnen beheer, Citaat invoegen en Bibliografie.

10. Bijlagen

Voeg extra documentatie toe zoals API-specificaties, configuratiebestanden, of aanvullende diagrammen.