Totstandkoming documentatie

Sil Gosker

**Versiebeheer**

|  |  |  |
| --- | --- | --- |
| **Wie** | **Wat** | **Wanneer** |
| Sil Gosker | Opzet van het document | 2025-02-11 |
|  |  |  |

Contents

[Inleiding 4](#_Toc190177533)

[Gedachteproces 5](#_Toc190177534)

[Verzamelen requirements 5](#_Toc190177535)

[Vaststellen assets en use cases 5](#_Toc190177536)

[Verwerken use cases 5](#_Toc190177537)

[Ontwikkelen use cases 6](#_Toc190177538)

[Revisie op risico’s 6](#_Toc190177539)

[Aansluiting op SDLC 6](#_Toc190177540)

# Inleiding

Dit document zal dieper ingaan op de totstandkoming van het functioneel ontwerp. Dit omvat het gedachteproces en aanpak van het functioneel ontwerp. Herstructureringen en hun reden worden niet opgenomen. Als dit relevant is, kan naar het versiebeheer (git) gerefereerd worden. Dit is beschikbaar in [WebDevShowCase/Documentatie at main · SilGosker/WebDevShowCase](https://github.com/SilGosker/WebDevShowCase/tree/main/Documentatie).

Om van globaal naar specifiek te gaan, zal chronologisch besproken hoe het functioneel ontwerp tot stand is gekomen. Dit begint bij het opschrijven van het denkproces, waarbij op ieder onderdeel binnen dit proces dieper ingegaan zal worden.

# Gedachteproces

Afbeelding met schermopname, lijn, Lettertype, diagram

Door AI gegenereerde inhoud is mogelijk onjuist.Dit hoofdstuk zal dieper ingaan op het globale gedachteproces. Hiervoor is een schema opgesteld aan de hand van de bpmn techniek in de Camunda Modeler. Het kan gevonden worden in Figuur 1: Procesdiagram ontwikkelen showcase. Per proces zal uitgelegd worden wat dit betekent en wat de input en uitkomsten zijn.

Figuur 1: Procesdiagram ontwikkelen showcase

Gedurende dit hele proces worden risico’s binnen de applicatie vastgesteld en gedocumenteerd. Alle risico’s worden vormgegeven d.m.v. evil user stories. Per risico wordt het probleem en zijn corresponderende oplossing besproken. De manier waarop dit gebeurt verschilt van stap tot stap. Per stap zal hier meer context aan gegeven worden.

## Verzamelen requirements

Dit omvat het analyseren wat de opdrachtgever precies wilt. De meest standaard wijze is interviews houden met de opdrachtgever. Incidenteel kan het zijn dat er een requirementsanalyse is opgesteld. In dat geval hoeft deze alleen geanalyseerd- en aangevuld te worden waar nodig.

Het vaststellen van risico’s gebeurt kan tijdens deze stap vormgegeven worden door vragen of aanbevelingen te doen tijdens een interview. Denk bijvoorbeeld aan een requirement die risicovolle functionaliteit bevat. In dat geval kan hierop doorgevraagd worden en, wanneer relevant, de requirement geschrapt als deze teveel risico’s met dit meebrengt. Dit gaat natuurlijk altijd in overleg met de opdrachtgever.

## Vaststellen assets en use cases

Wanneer alle requirements duidelijk zijn, kunnen deze gegroepeerd worden op basis van use cases. Alle relevante assets worden vastgesteld en opgeschreven in het security-analyse document.

Het vaststellen van risico’s tijdens deze stap wordt vormgegeven door het kwalitatief inschatten van de waarde binnen verschillende aspecten van de assets. Als de totale waarde van de asset hoog genoeg wordt geschat, worden direct maatregelen bedacht om deze te beschermen.

## Verwerken use cases

Wanneer de use cases, assets en maatregelen voor de assets vastgesteld zijn, kan de start gemaakt worden voor de ontwerpfase. Hierin worden de gebruikelijke aspecten die in de ontwerpfase terugkomen gemaakt. Denk hierbij aan het maken van use case diagrammen, wireframes, C4-diagrammen, state diagrams en/of activity diagrams.

Het vaststellen van risico’s tijdens deze stap gebeurt voornamelijk na het definiëren van het use case diagram. Bijna alle alternatieve scenario’s kunnen gekoppeld worden aan een risico, en dus ook aan een maatregel.

## Ontwikkelen use cases

Als de ontwerpfase als “af” gezien wordt, kan de realiseer-fase van start gaan, waarbij een use case daadwerkelijk ontwikkelt gaat worden. Dit betekent code schrijven totdat het voldoet aan de eerder genoemde requirements en ontwerpen.

Ook tijdens de realiseerfase kan het voorkomen dat een nieuw risico ondervonden wordt. Dit kan een oud risico zijn waar geen oplossing voor gevonden was, of een nieuw, niet eerder gevonden risico.

## Revisie op risico’s

Wanneer de realiseerfase af is (en de code getest), worden de risico’s nog een keer nagelopen. Is voor ieder risico een oplossing gevonden? Als dit niet het geval is, wordt een kort onderzoek uitgevoerd om te bepalen of dit opgelost kan worden.

## Aansluiting op SDLC

Afbeelding met tekst, schermopname, cirkel

Door AI gegenereerde inhoud is mogelijk onjuist.Dit proces is gebaseerd op het SDLC- ontwikkelingsproces. Dit betekent dat ieder genoemd aspect terugkomt in deze cirkel. De vormgeving hiervan kan teruggevonden worden in Figuur 2: gedachteproces aansluiting op SDLC.

Figuur 2: gedachteproces aansluiting op SDLC