

Ontwerp

<Naam van het Project>

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Plaats | : |  |  |  |
| Datum | : | *dd-mm-jjjj* |  |  |
| Versie | : | *x.x* |  |  |
| Status | : | *Concept / Definitief* |  |  |
| Auteur | : | *Naam* |  |  |
| Studentnummer | : |  |  |  |
| Klas | : |  |  |  |

# Revisiehistorie

|  |  |  |
| --- | --- | --- |
| **Datum** | **Versie** | **Omschrijving** |
| 12-5-2021 | 0.1 | 1e opzet Ontwerpdocument |
|  |  |  |

Inhoudsopgave

[1 Inleiding 3](#_Toc35508466)

[1.1 Optionele paragrafen: 3](#_Toc35508467)

[2 Overzicht 3](#_Toc35508468)

[3 Uses cases 3](#_Toc35508469)

[3.1 Actoren 3](#_Toc35508470)

[3.2 Use case templates 3](#_Toc35508471)

[3.3 Wireframes 3](#_Toc35508472)

[4 Domeinmodel 3](#_Toc35508473)

[5 Technologieën 4](#_Toc35508474)

[6 Overdracht 4](#_Toc35508475)

[7 Referenties 4](#_Toc35508476)

# Inleiding

Neem hier op hoe het ontwerp tot stand gekomen is alsmede voor wie het bedoeld is met eventueel een lezersadvies.  
  
Start: Neem de inleiding over van je PvA door copy-pasten.

1. Trek de lezer
2. Beschrijf de achtergrond van de opdrachtgever en de casus
3. Beschrijf de kwestie en aanleiding
4. Noem de oplossing
5. Leeswijzer

*Aangepast op wat er in die versie van het ontwerpdocument moet worden opgeleverd; zie hiervoor de opdracht bij de Sprint op Canvas.*

# Overzicht

Korte beschrijving wat het systeem geacht wordt te doen. Eventueel indeling in subsystemen met een korte beschrijving van een subsysteem.

# Use cases

Use case diagram met use case samenvattingen voor elke use case.

Zie bij de cursus modelling hoe je dit model maakt.

## Actoren

Bij een actor hoort ook een beschrijving, de *actor description*. In een actor template wordt naast de rolnaam een beschrijving van de rol ingevuld en daarmee een *role description* geschreven.  
  
*Voor studenten die dit al hebben gedaan in het PvA, kun je dit gewoon copy-pasten.*

# Domeinmodel

Tip: Voer de zelfstandignaamwoordenmethode uit op basis van je use case samenvattingen. Maak aan de hand ervan een conceptueel model.

Voor het conceptueel domeinmodel voor sprint 1 zijn er de volgende eisen:

1. Domeinklassen
2. Attributen
3. Associaties (/ Aggregaties / Composities)
4. Multipliciteiten
5. Generalisaties

Dus voorlopig nog geen operaties en methoden; ook geen navigeerbaarheid. Geen pijlen moeten er getoond worden. Visibility’s zijn ook nog niet verplicht.

De meeste entiteiten uit het conceptueel datamodel hebben een tegenhanger in de vorm van een JAVA domeinklasse. Van de student wordt verwacht dat er een UML klassen diagram wordt opgeleverd voor de domeinklassen. Zorg dat je UML klassendiagram volledig is.

Aandachtspunten:

- Neem alle attributen op met bijbehorende types

- Neem bij elke associatie de multipliciteiten en de rolnamen op.

- Maak gebruik van de juiste notatie. Zie de cursus OODC

Neem het diagram op met daarbij een korte beschrijving van alle entiteiten. Zie bij de cursus modelling hoe je dit model maakt.

|  |  |
| --- | --- |
| Entiteit | Beschrijving |
|  |  |
|  |  |
|  |  |

**Business rules op het domeinmodel**

Daarnaast kent een domeinmodel eventueel extra regels op het domein (business rules) die je niet goed in UML kan vertalen. Deze regels neem je hier puntsgewijs op.

# Technologieën

*Hier hoeft voor sprint 1 nog niks te komen.*

Voor het realiseren van de applicatie wordt gebruik van de volgende methodieken en technieken: UML, Java, SQL, HTML, CSS, J2EE (Servlets), Rest (Jax-RS), Applicatieserver

& HTTP-protocol en een datastore. Beschrijf ook welke frameworks je gebruikt, hiervan neem je ook de versie en het licentiemodel op.

# Overdracht

*Hier hoeft voor sprint 1 nog niks te komen.*

Hier wordt een installatiehandleiding en een sumiere gebruikershandleiding beschreven. Ook eventuele gebruikersnamen en wachtwoorden moet bekend worden om de werking van het programma aan te tonen.

# Referenties

*Doe hier alleen referenties, als ze nodig zijn.*

Geef hier de bronnenlijst. Gebruik de APA stijl om de bronnen te vermelden.