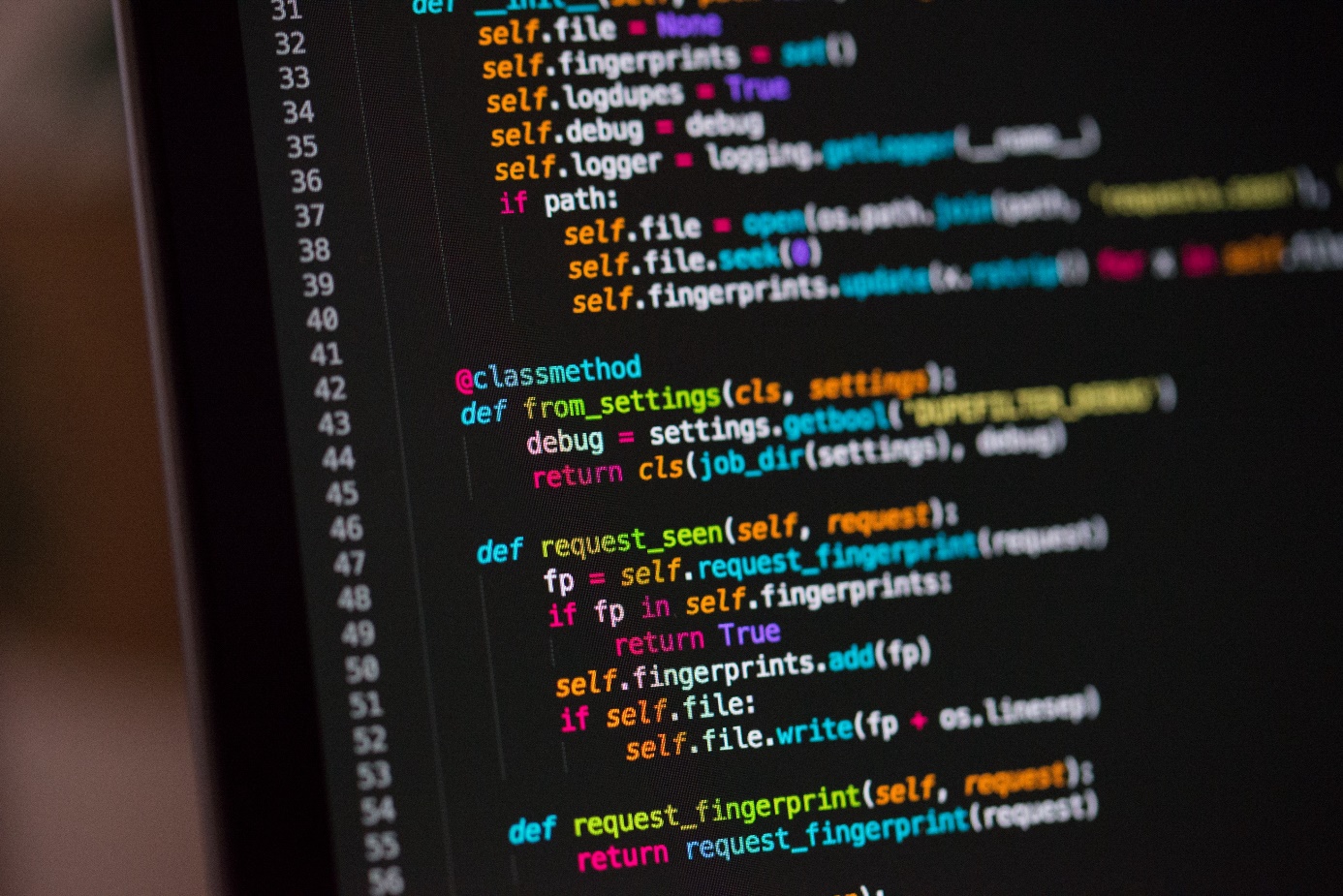
***Documentatie Tom Derks***



**Tom Derks – DB-05**

**Docenten : Ruben Steins, Tim Bijnen**

**Fontys, HBO ICT Software - Semester 2**

Inhoudsopgave

[Projectbeschrijving 3](#_Toc74924430)

[Leeruitkomsten uitwerking 4](#_Toc74924431)

[UI Schetsen 5](#_Toc74924432)

[Conceptueel Model 6](#_Toc74924433)

[Contextdiagram 7](#_Toc74924434)

[Testplan 8](#_Toc74924435)

[Test Matrix 9](#_Toc74924436)

[Requirements 10](#_Toc74924437)

[Use-Cases 12](#_Toc74924438)

[Moscow methode 23](#_Toc74924439)

[Klassen Diagram 24](#_Toc74924440)

[Architectuur 25](#_Toc74924441)

[Databaseontwerpen 26](#_Toc74924442)

# Projectbeschrijving

Gedetailleerde beschrijving van wat ik wil maken voor mijn individuele project.

**Hoofddoel:**

- De inventaris van een bedrijf weergeven

- Inventaris toevoegen aan de online weergaven

- Inventaris verwijderen uit de online weergaven

**Belangrijkste Gebruiker:**

- Bedrijven (specifiek dat van mijn vader)

**Vergelijkbare applicaties:**

- Ordoro (e-commerce inventory management system)

- Upserve (Restaurant inventory management system)

- Zoho Inventory (Inventory management system voor kleine bedrijven)

**Projectbeschrijving verhaal:**

Ik wil een inventaris management systeem maken voor het bedrijf van mijn vader waarin hij zijn inventaris kan zien, aanpassen door dingen toe te voegen en te verwijderen en mogelijk als ik extra tijd heb ook de verkochte producten en bestelde producten. Hier is dus het bedrijf van mijn vader de belangrijkste gebruiker. Vergelijkbare applicaties die betrekking hebben tot mijn project zijn Ordoro, Upserve en Zoho. Ordoro is het meest vergelijkbaar met mijn einddoel, ook met het verkoop en inkoop gedeelte. Upserve heeft een vooral een verkoop overzicht wat ik overzichtelijk vond en dus toepasselijk is voor mijn project en hoewel deze applicatie meer voor restaurants is, is de verkoop geschiedenis grafiek en functie een die mij wel aanstond. Als laatste vond ik het overzicht van de inventaris en alles bij elkaar van Zoho heel fijn en toegankelijk, en daar zou ik dus ook inspiratie van kunnen nemen

# Leeruitkomsten uitwerking

Uitwerkingen van de 7 leeruitkomsten en hoe ik ze ga halen.

**Leeruitkomst 1:**

- Constant feedback vragen aan docenten en eventueel bedrijf.

- Keuzes duidelijk aangeven in vorm van verslag en met sturing van docenten.

**Leeruitkomst 2:**

- Goed communiceren en verslag bijhouden op professionele wijze.

- Samen werken met de docenten door middel van feedback vragen zodat ik de richtlijnen die zij geven goed kan volgen.

- Regelmatig Reflecteren op mezelf aan de hand van feedback van docenten en mogelijk medestudenten. Aan de hand hiervan project bijwerken.

**Leeruitkomst 3:**

- Maak requirements voor het project mogelijk met hulp van bedrijf waarvoor ik het project maak.

- Maak een lijst van software functies aan de hand van deze requirements om toe te voegen in mijn project.

**Leeruitkomst 4:**

- Werken aan meerdere werkende prototypes om op te leveren met Object Oriented Programming en de basis technieken van programmeren.

- Meerdere extra oefeningen maken op kleinere schaal zodat ik goed weet wat ik doe voordat ik iets in mijn hoofd project zet(geprogrammeerd aan de hand van OOP en de standaard technieken).

**Leeruitkomst 5:**

- Verschillende algoritmes gebruiken in combinatie met de requirements die de stakeholders en ik hebben gemaakt.

- Extra algoritme opgaven maken om verder bewijslast toe te voegen.

- Verschillende Computationele Uitdagingen oplossen die zich voor doen met werken met grote nummers die zich voor doen bij het project.

**Leeruitkomst 6:**

- Database maken voor opslag van de data van de inventaris van het bedrijf, die aangepast kan worden door een aparte applicatie.

- Deze Database linken aan de hoofdapplicatie zodat deze de Database kan bijwerken (toevoegen, verwijderen en opvragen).

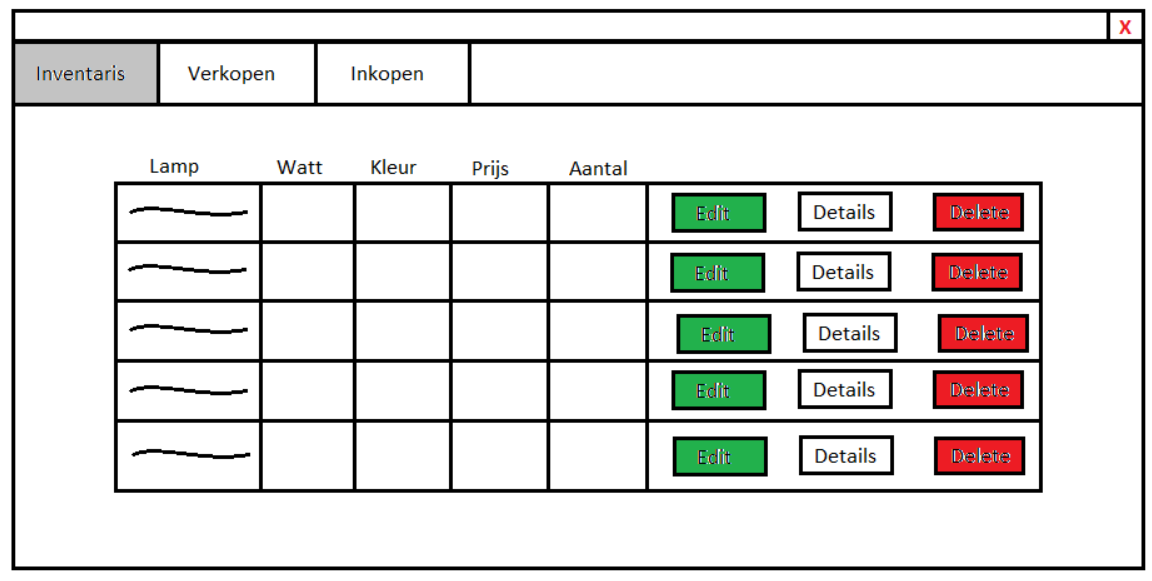
**Leeruitkomst 7:**

- Constant opleveringen doen van Prototypes en feedback vragen aan docenten en anderen om kwaliteit hoog te houden.

- Gebruik van unit testen en of versie beheer systemen (en dus github)

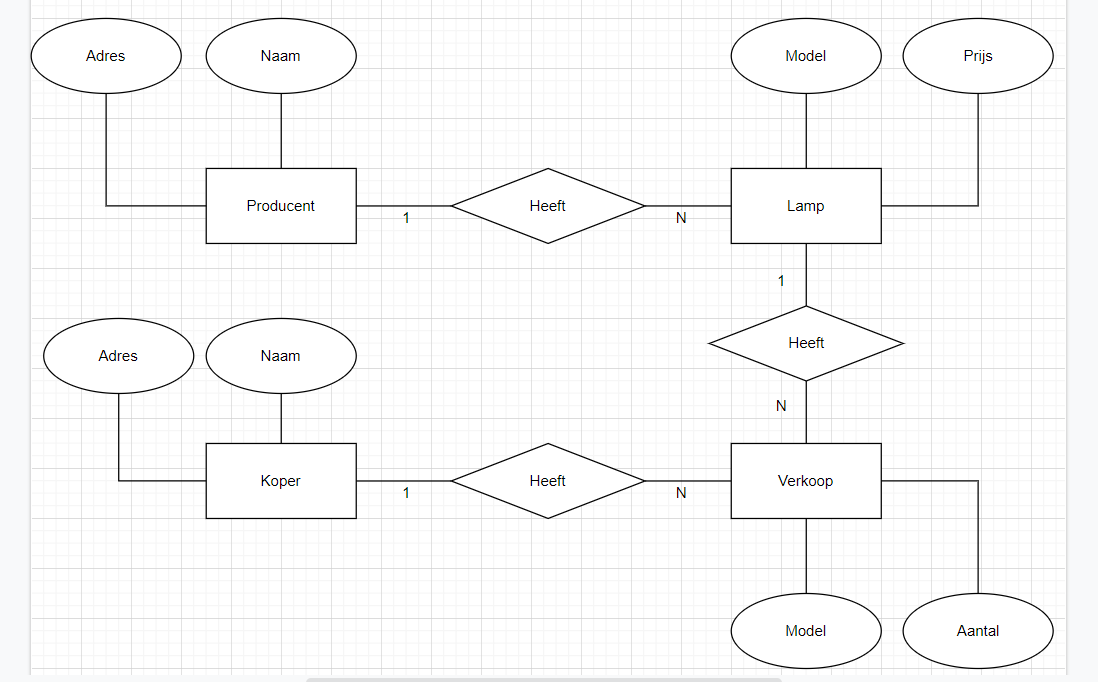
# UI Schetsen

Mijn schetsen van hoe mijn UI er ongeveer uit gaat zien.



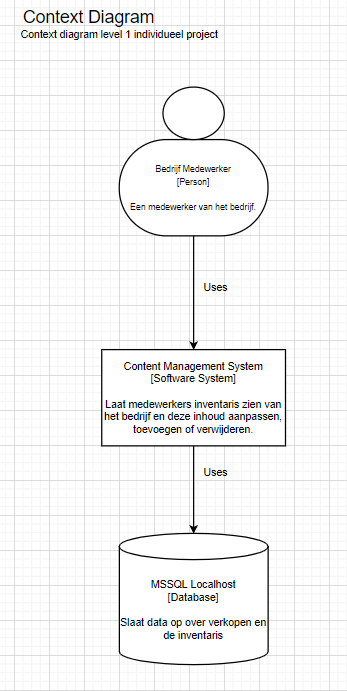
# Conceptueel Model

Een diagram die weergeeft hoe mijn applicatie er uit gaat zien. Ik heb ervoor gekozen om 1 verkoop niet meerdere lampen te geven omdat mijn vader dit niet wilt gebruiken als een verkoop applicatie maar puur als inventaris app. Hij heeft namelijk een andere manier van verkopen te weergeven dus hij vind het fijner als hij de verkopen apart heeft staan omdat dit meer als export is dan verkoop en het daarom niet uitmaakt of het in dezelfde verkoop zit, zolang het maar uit de inventaris is.



# Contextdiagram

Level 1 context diagram, deze weergeeft hoe er met mijn applicatie word gewerkt.



# Testplan

|  |  |  |  |
| --- | --- | --- | --- |
| Test case | Use case | Invoer | Verwachte uitvoer |
| TC01 | UC01 | Model: Test Lamp  Watt: 30  Kleur: 60  Prijs: 5,-  Producent: Phillips  Aantal: 50 | Lamp toegevoegd aan de inventaris |
| TC02 | UC02 | Delete functie activeren bij lamp | Verwijderd de lamp uit de database en inventaris. |
| TC03 | UC03 | Model: Test Lamp | Weergeeft alle Lampen die de Model naam Test Lamp bevatten. |
| TC04 | UC04 | Lamp: Test Lamp  Koper: Jan Willem  Aantal: 15 | Verkoop toegevoegd in verkopen, en aantal lampen afgerekend uit inventaris. |
| TC05 | UC08 | Activeer Producent Lamp functie | Weergeeft alle lampen van 1 producent. |
| TC06 | UC09 | Model: Test Edit  Watt: 35  Kleur: 65  Prijs: 5,-  Aantal: 55 | Veranderd de data van de lamp naar de nieuwe aangepaste versie. |
| TC07 | UC09 | Model:  Watt: 35  Kleur: 35  Prijs: 35,40  Aantal: 20 | Weergeeft een error dat een veld leeg staat. |
| TC08 | UC10 | Activeer Koper Verkoop functie | Weergeeft alle lampen van 1 koper. |
| TC09 | UC11 | Model: Test Lamp | Weergeeft Verkopen op naam in lijst vorm. |
| TC10 | UC01 | Model: Test Lamp  Watt: 30  Kleur: Wit  Prijs: 5,55  Producent: Philips  Aantal: 50 | Weergeeft dat watt geen letters kan hebben, omdat kleur in getallen word aangegeven. |
| TC11 | UC01 | Model:  Watt: 30  Kleur: 30  Prijs: 3,25  Producent: Phillips  Aantal: 30 | Weergeeft dat er een veld niet is ingevuld in de form. |

# Test Matrix

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | FR-01 | B-01.1 | B-01.2 | B-01.3 | FR-02 | B-02.1 | FR-03 | B-03.1 | FR-04 | B-04.1 | B-04.2 |
| TC01 | X |  |  | X |  |  |  |  |  |  |  |
| TC10 |  | X |  |  |  |  |  |  |  |  |  |
| TC11 |  |  | X |  |  |  |  |  |  |  |  |
| TC02 |  |  |  |  | X | X |  |  |  |  |  |
| TC03 |  |  |  |  |  |  | X | X |  |  |  |
| TC04 |  |  |  |  |  |  |  |  | X | X | X |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | FR-08 | B-08.1 | FR-09 | B-09.1 | B-09.2 | FR-10 | B-10.1 | FR-11 | B-11.1 |
| TC05 | X | X |  |  |  |  |  |  |  |
| TC06 |  |  | X | X |  |  |  |  |  |
| TC07 |  |  |  |  | X |  |  |  |  |
| TC08 |  |  |  |  |  | X | X |  |  |
| TC09 |  |  |  |  |  |  |  | X | X |

# Requirements

De Requirements van mijn applicatie.

FR-01 Lampen toevoegen  
Je moet lampen toe kunnen voegen in de inventaris.  
B-01.1 Lampen hebben 6 eigenschappen; Model, Watt, Kleur, Producent, Aantal en prijs, allemaal met eigen invul soorten(bv. Watt moet een nummer zijn).  
B-01.2 Geen van de invul vakken mogen leeg zijn.  
B-01.3 Prijs moet in euro’s.

FR-02 Lampen verwijderen  
Je moet de lampen in de inventaris kunnen verwijderen.  
B-02.1 Achter elke lamp moet een verwijder knop staan.

FR-03 Lampen zoekfunctie  
Je moet de lampen in de inventaris op kunnen zoeken.  
B-03.1 Zoekfunctie moet op model naam gaan.

FR-04 Verkopen toevoegen  
Je moet een verkoop kunnen toevoegen aan een aparte tabel.  
B-04.1 Bij het aanmaken kun je alleen kiezen uit bestaande lampen in de inventaris.  
B-04.2 Bij het aanmaken word de totaal prijs weergeven door het aantal in de verkoop maal de prijs.

FR-05 Verkopen verwijderen  
Je moet de verkopen in de tabel kunnen verwijderen.  
B-05.1 Als een verkoop verwijderd word moet het aantal van de verkoop terug in de inventaris.

FR-06 Inventaris automatisch updaten  
De inventaris moet automatisch geüpdatet worden als een verkoop toe gevoegd word.  
B-06.1 Als een verkoop gemaakt word moet het aantal lampen van de verkoop uit de inventaris.  
B-06.2 In de inventaris moeten lampen onder de 0 kunnen (bijv. - 3), dit was op aanvraag van het bedrijf

FR-07 Lampen details weergeven  
Je moet details kunnen zien als je op een lamp klikt in de inventaris.  
B-07.1 Dit moet alles weergeven wat in de tabel stond onder elkaar.  
B-07.2 Dit moet ook de verwijder en edit knop hebben net als de lijst weergaven.

FR-08 Lampen weergeven per producent  
Je moet de inventaris kunnen sorteren per producent.  
B-08.1 Elke producent moet een lijst kunnen weergeven van de lampen die ze gemaakt hebben.

FR-09 Lampen editen  
Je moet de lampen in de inventaris kunnen aanpassen.  
B-09.1 Je moet alles kunnen editen van de lamp behalve de Id.  
B-09.2 Er mag hier net als op de creatie pagina geen veld leeg zijn behalve de details.

FR-10 Verkopen weergeven per koper  
De verkopen moeten weergeven worden per koper.  
B-10.1 Elke koper moet een lijst kunnen weergeven met hun aankopen.

FR-11 Verkopen zoeken op model

De verkopen moeten gezocht kunnen worden op model van de lamp.  
B-11.1 De Model naam moet van de lamp komen die gekoppeld is aan de verkoop.

# Use-Cases

Mijn Use-Cases van al mijn requirements.

Lamp toevoegen (FR-01)

**Samenvatting**

Een bedrijf wil een nieuwe lamp toevoegen aan de inventaris.

**Actors**Medewerker bedrijf.

**Aannamen**Geen.

**Scenario**1. De actor gaat naar de create functie van de applicatie.

2. de software geeft invul opties weer voor een lamp aan te maken.

3. De actor vult de benodigde informatie in..

4. De software checkt of de ingevulde informatie correct is.

5. De actor bevestigd de informatie.

6. De software slaat de informatie op en brengt de actor naar de inventaris interface.

**Uitzonderingen**1. De actor heeft niet alle benodigde info ingevuld en krijgt een error (Scenario stap 3).

2. De actor voert de verkeerde datasoort in en krijgt hierdoor een error (Scenario stap 3).

**Resultaat**

Een nieuwe lamp is toegevoegd in de inventaris database en pagina lijst.

Lamp Verwijderen (FR-02)

**Samenvatting**

Een bedrijf wil een lamp verwijderen omdat deze niet meer nodig is in de inventaris.

**Actors**

Medewerker Bedrijf.

**Aannamen**

Er staan lampen in de database.

**Scenario**

1. De actor ziet een lamp die weg moet en activeert de verwijder functie.

2. De software stuurt de actor naar de verwijder interface.

3. De actor bevestigd dat hij/zij de informatie wilt verwijderen.

4. De software verwijderd de data en stuurt de actor terug naar de inventaris interface.

**Uitzonderingen**

1. De actor bevestigd zijn/haar keuze niet en word terug gestuurd naar de inventaris interface (Scenario stap 1).

**Resultaat**

De geselecteerde lamp is verwijderd uit de database en dus ook de inventaris tabel.

Lampen Zoekfunctie (FR-03)

**Samenvatting**

Een bedrijf wil een specifieke lamp vinden in de inventaris.

**Actors**

Medewerker bedrijf.

**Aannamen**

Er staan lampen in de database

**Scenario**

1. De actor voert de naam in van de lamp die hij/zij wilt zien en bevestigd dit.

2. De software weergeeft de lamp die overeen komt met de naam.

**Uitzonderingen**

1. De actor typt een naam die niet overeen komt met de namen van de lampen in de database waardoor de tabel niks weergeeft. (Scenario stap 1)

**Resultaat**

De actor ziet de lamp of lampen die de naam die hij heeft ingevoerd bevatten.

Verkopen toevoegen (FR-04)

**Samenvatting**

Een bedrijf wil een verkoop toevoegen aan de verkoop tabel.

**Actors**

Medewerker bedrijf.

**Aannamen**

Er staan lampen in de inventaris.

**Scenario**

1. De actor navigeert naar de verkoop pagina en gaat naar de create functie.

2. De software stuurt de actor naar het creatie interface.

3. De actor kies de lamp die verkocht word en voert het aantal verkochte lampen in.

4. De software controleert de data die is ingevoerd.

5. De actor klikt op de create/bevestig knop.

6. De software slaat de data op en stuurt de actor terug naar de verkoop interface.

**Uitzonderingen**

1. De actor voert een datatype in die niet overeen komt met de database waardoor hij een error krijgt. (Scenario stap 3).

2. De actor voert niks in waardoor hij een error krijgt (Scenario stap 3).

**Resultaat**

Er staat een nieuwe verkoop in de database en interface.

Verkopen Verwijderen (FR-05)

**Samenvatting**

Je moet verkopen kunnen verwijderen.

**Actors**

Medewerker bedrijf.

**Aannamen**

Er staan Verkopen en lampen in de database.

**Scenario**

1. De actor kiest de lamp die hij/zij wil verwijderen en activeert de delete functie.

2. De software stuurt de actor naar een bevestiging van de verwijdering.

3. De actor bevestigd dat hij de verkoop wil verwijderen.

4. De software verwijderd de data en stuurt de actor naar de verkoop interface.

**Uitzonderingen**

1. De actor bevestigd zijn keuze niet en gaat terug naar de verkoop interface (Scenario stap 1).

**Resultaat**

De Actor heeft een verkoop verwijderd en de verkochte lampen terug toegevoegd aan de inventaris.

Inventaris automatisch updaten(FR-06)

**Samenvatting**

Als iemand een verkoop aanmaakt, update de inventaris zich automatisch door de lampen te verwijderen.

**Actors**

Medewerker bedrijf.

**Aannamen**

Er staan lampen in de inventaris.

**Scenario**

1. De actor voegt een verkoop toe aan de verkoop tabel. (Use case FR-04)

2. De software pakt het aantal uit de verkoop en trekt deze af van de lamp in de inventaris.

**Uitzonderingen**

Geen.

**Resultaat**

De data in de inventaris past zich aan aan de veranderingen in andere tabellen.

Lampen Details weergeven(FR-07)

**Samenvatting**

Een actor kiest een lamp en ziet hiervan uitgebreide informatie.

**Actors**

Bedrijf medewerker.

**Aannamen**

Er staan lampen in de inventaris.

**Scenario**

1. De actor gebruikt de detail functie bij de lamp waarvan hij informatie wilt zien.

2. De software brengt de actor naar een detail interface met extra info.

**Uitzonderingen**

Geen.

**Resultaat**

De actor ziet meer informatie over de gekozen lamp.

Lampen weergeven per producent (FR-08)

**Samenvatting**

De lampen in de inventaris worden gesorteerd per producent.

**Actors**

Medewerker bedrijf.

**Aannamen**

Er staan lampen in de inventaris.

**Scenario**

1. De actor gaat naar de inventaris pagina selecteert een producent.

2. De software stuurt de actor naar een weergaven van alle lampen per producent.

**Uitzonderingen**

Geen.

**Resultaat**

De data is gesorteerd door de software op Producent.

Lampen editen (FR-09)

**Samenvatting**Een bedrijf wil een bestaande lamp editen in de inventaris.

**Actors**Medewerker bedrijf.

**Aannamen**Er staan lampen in de inventaris.

**Scenario**

1. De actor activeert de edit functie bij een lamp.

2. De software brengt hem/haar naar de edit interface.

3. De actor past de informatie aan die aangepast moet worden en bevestigd dit.

4. De software slaat de data op en stuurt de actor terug naar de inventaris interface.

**Uitzonderingen**

1. De actor heeft niet alle benodigde info ingevuld en krijgt hierdoor een error (Scenario stap 3).

2. De actor heeft verkeerde waardes ingevoerd en krijgt een error (Scenario stap 3).

**Resultaat**

De gekozen lamp is nu veranderd met de nieuw ingevoerde info.

Verkopen weergeven per koper (FR-10)

**Samenvatting**

De verkopen in de verkoop weergaven worden gesorteerd per koper.

**Actors**

Medewerker bedrijf.

**Aannamen**

Er staan verkopen in de verkoop pagina.

**Scenario**

1. De actor gaat naar de verkoop pagina selecteert een koper.

2. De software stuurt de actor naar een weergaven van alle verkopen per koper.

**Uitzonderingen**

Geen.

**Resultaat**

De data is gesorteerd door de software op Koper.

Verkopen zoeken op model (FR-11)

**Samenvatting**

De verkopen in de inventaris worden gefilterd op naam.

**Actors**

Medewerker bedrijf.

**Aannamen**

Er staan verkopen in de verkoop weergaven.

**Scenario**

1. De actor gaat naar de verkoop weergave pagina selecteert de zoekfunctie en voert de zoeknaam in.

2. De software filtert de verkopen op de naam van de lamp gekoppeld aan de verkoop.

**Uitzonderingen**

Als een naam word ingevuld die niet bestaat weergeeft de software niks (Scenario stap 1).

**Resultaat**

De data is gefilterd door de software op de ingevulde naam.

# Moscow methode

Mijn eerder gemaakte requirements geordend op hoe nodig ze zijn.

Must Have

1. Lamp toevoegen (FR-01)
2. Lamp Verwijderen (FR-02)
3. Verkopen toevoegen (FR-04)
4. Verkopen verwijderen (FR-05)
5. Lampen editen (FR-09)
6. Data overzichtelijk weergeven (FR-10)
7. Verkopen weergeven per Koper (FR-11)
8. Lampen per Producent weergeven (FR-08)

Should Have

1. Inventaris automatisch updaten (FR-06)

Could Have

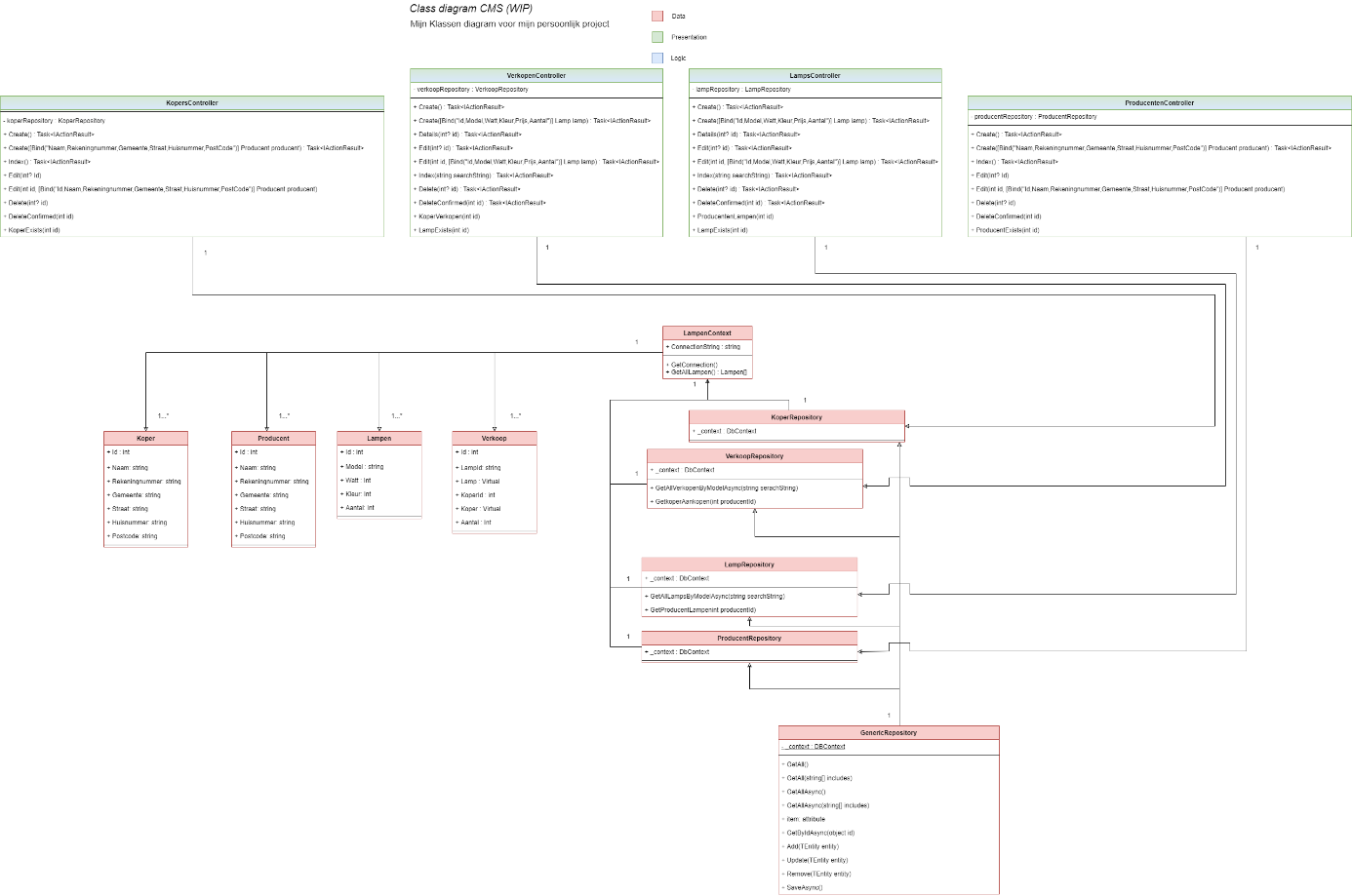
1. Lampen zoek functie (FR-03)
2. Lampen details weergeven (FR-07)

Would Have

1. **Geen**

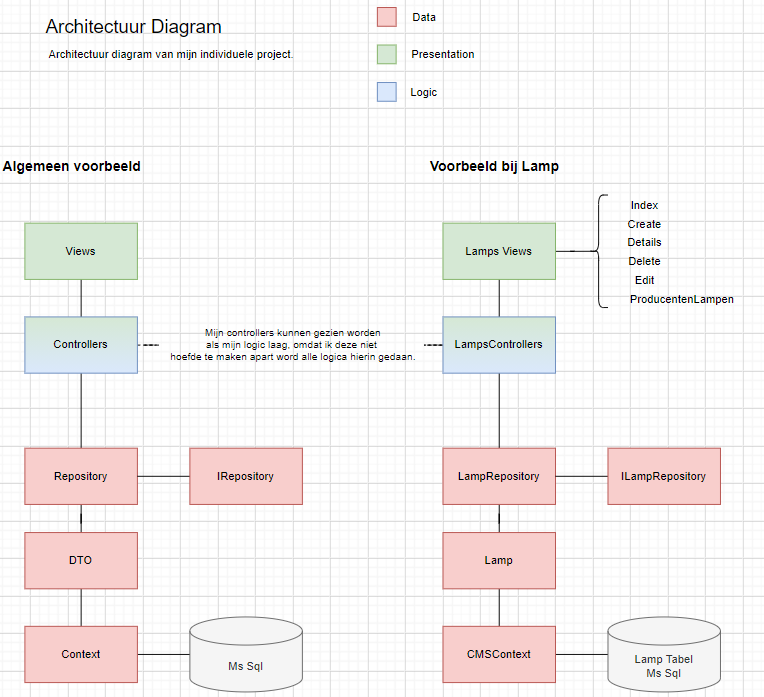
# Klassen Diagram

Gezien deze niet leesbaar is op een word pagina, heb ik deze apart toegevoegd in de map “Bijlagen” onder de naam “KlassenDiagram”. Hier hebben mijn controller 2 kleuren en behoren ze dus tot 2 lagen vanwege dezelfde reden dat dit gedaan is in mijn architectuur diagram, omdat ik geen logic laag nodig had en mijn controllers alle logica uitvoeren.



# Architectuur

Nog niet gemaakt omdat ik nog niet weet of mijn layers correct zijn. Je kunt deze diagram ook bekijken in png formaat in de “Bijlage” folder onder de naam “ArchitectuurDiagram”.



# Databaseontwerpen

