

## Documentatie Tom Derks

```
31     def __init__(self, settings):
32         self.file = None
33         self.fingerprints = set()
34         self.logdups = True
35         self.debug = debug
36         self.logger = logging.getLogger(__name__)
37         if path:
38             self.file = open(os.path.join(path, 'requests.log'),
39                             'a')
40             self.file.seek(0)
41             self.fingerprints.update(e.request() for e in self.requests)
42
43     @classmethod
44     def from_settings(cls, settings):
45         debug = settings.getbool('SUPERFILTER_DEBUG')
46         return cls(job_dir(settings), debug)
47
48     def request_seen(self, request):
49         fp = self.request_fingerprint(request)
50         if fp in self.fingerprints:
51             return True
52         self.fingerprints.add(fp)
53         if self.file:
54             self.file.write(fp + os.linesep)
55
56     def request_fingerprint(self, request):
57         return request_fingerprint(request)
```

Tom Derks – DB-05

Docenten : Ruben Steins, Tim Bijnen

Fontys, HBO ICT Software - Semester 2

## Inhoudsopgave

Projectbeschrijving .....	3
Leeruitkomsten uitwerking .....	4
Requirements .....	5
Use-Cases .....	6
Moscow methode .....	16
Klassen Diagram .....	17
UI Schetsen .....	18
Contextdiagram .....	19
Architectuur.....	20
Databaseontwerpen.....	21

## 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 Computationale 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)

## 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, Aantal, prijs en details.

B-01.2 Geen van de invul vakken mogen leeg zijn behalve details.

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 naam gaan.

### FR-04 Verkoop 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 weergegeven door het aantal in de verkoop maal de prijs.

### FR-05 Verkoop 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.

B-07.3 Dit moet extra info weergeven over de lamp die niet zichtbaar is in de tabel.

### FR-08 Lampen sorteer functie

Je moet de inventaris kunnen sorteren per lamp.

B-08.1 Dit moet ascending en descending kunnen zijn.

B-08.2 Dit moet kunnen op watt en kleur.

### 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 Data overzichtelijke weergeven

De data van beide de inventaris en verkopen moet op een duidelijke manier zijn weergegeven.

B-10.1 Data moet in een overzichtelijk lijstje worden weergegeven dit op meer detail kan gaan per lamp d.m.v. de details knop.

## 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 bevestigt 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 bevestigt dat hij/zij de informatie wilt verwijderen.
4. De software verwijdert de data en stuurt de actor terug naar de inventaris interface.

### **Uitzonderingen**

1. De actor bevestigt 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 bevestigt 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 bevestigt dat hij de verkoop wil verwijderen.
4. De software verwijdert de data en stuurt de actor naar de verkoop interface.

### **Uitzonderingen**

1. De actor bevestigt 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 sorteer functie(FR-08)

### **Samenvatting**

De lampen in de inventaris worden gesorteerd op specifieke volgorde.

### **Actors**

Medewerker bedrijf.

### **Aannamen**

Er staan lampen in de inventaris.

### **Scenario**

1. De actor gaat naar de inventaris pagina en activeert de sorteer functie.
2. De software geeft optie(s) waarop de actor de tabel kan sorteren.
3. De actor kiest de sorteer methode.
4. De software sorteert de data op de volgorde aangegeven door de actor.

### **Uitzonderingen**

Geen.

### **Resultaat**

De data is gesorteerd door de software.

## 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 bevestigt 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.

## Data overzichtelijk weergeven(FR-10)

### **Samenvatting**

Als een actor de app opent kan hij/zij de data van de applicatie overzichtelijk zien.

### **Actors**

Medewerker bedrijf.

### **Aannamen**

Er staan lampen in inventaris en/of verkopen.

### **Scenario**

1. De actor opent de software.
2. De software start op en haalt data op.
3. De actor gaat naar de inventaris of verkoop interface.
4. De software haalt de data van een van de twee interfaces op en weergeeft dit in een tabel formaat voor de actor.

### **Uitzonderingen**

Geen.

### **Resultaat**

De actor kan de opgeslagen data overzichtelijk bekijken.

## 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)

### Should Have

1. Inventaris automatisch updaten (FR-06)
2. Lampen details weergeven (FR-07)

### Could Have

1. Lampen zoek functie (FR-03)
2. Lampen sort functie (FR-08)

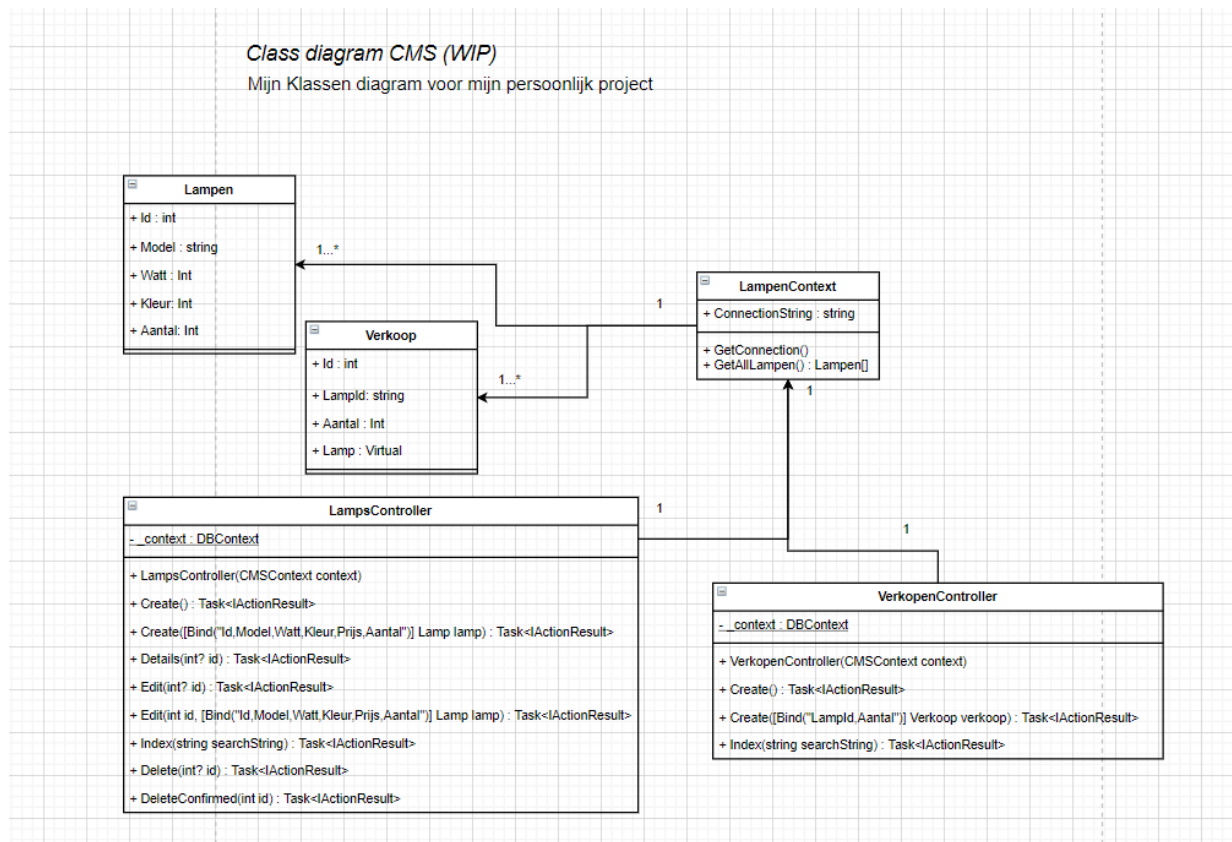
### Would Have

1. **Geen**



# Klassen Diagram

Mijn klassen diagram van mijn hoofd project.



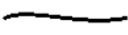
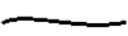
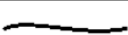


## UI Schetsen

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

Inventaris

Verkopen

Inkopen

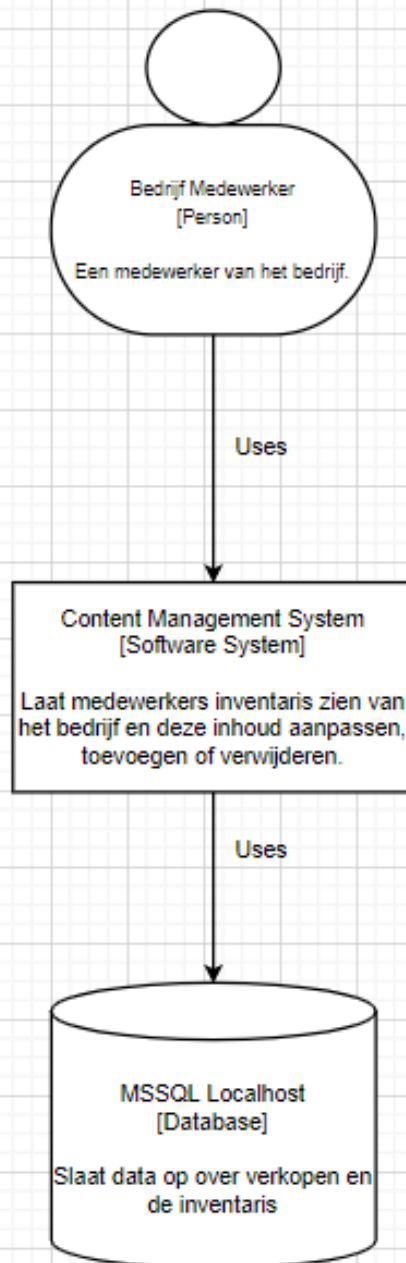
Lamp	Watt	Kleur	Prijs	Aantal			
					Edit	Details	Delete
					Edit	Details	Delete
					Edit	Details	Delete
					Edit	Details	Delete
					Edit	Details	Delete

## Contextdiagram

Level 1 context diagram.

### Context Diagram

Context diagram level 1 individueel project

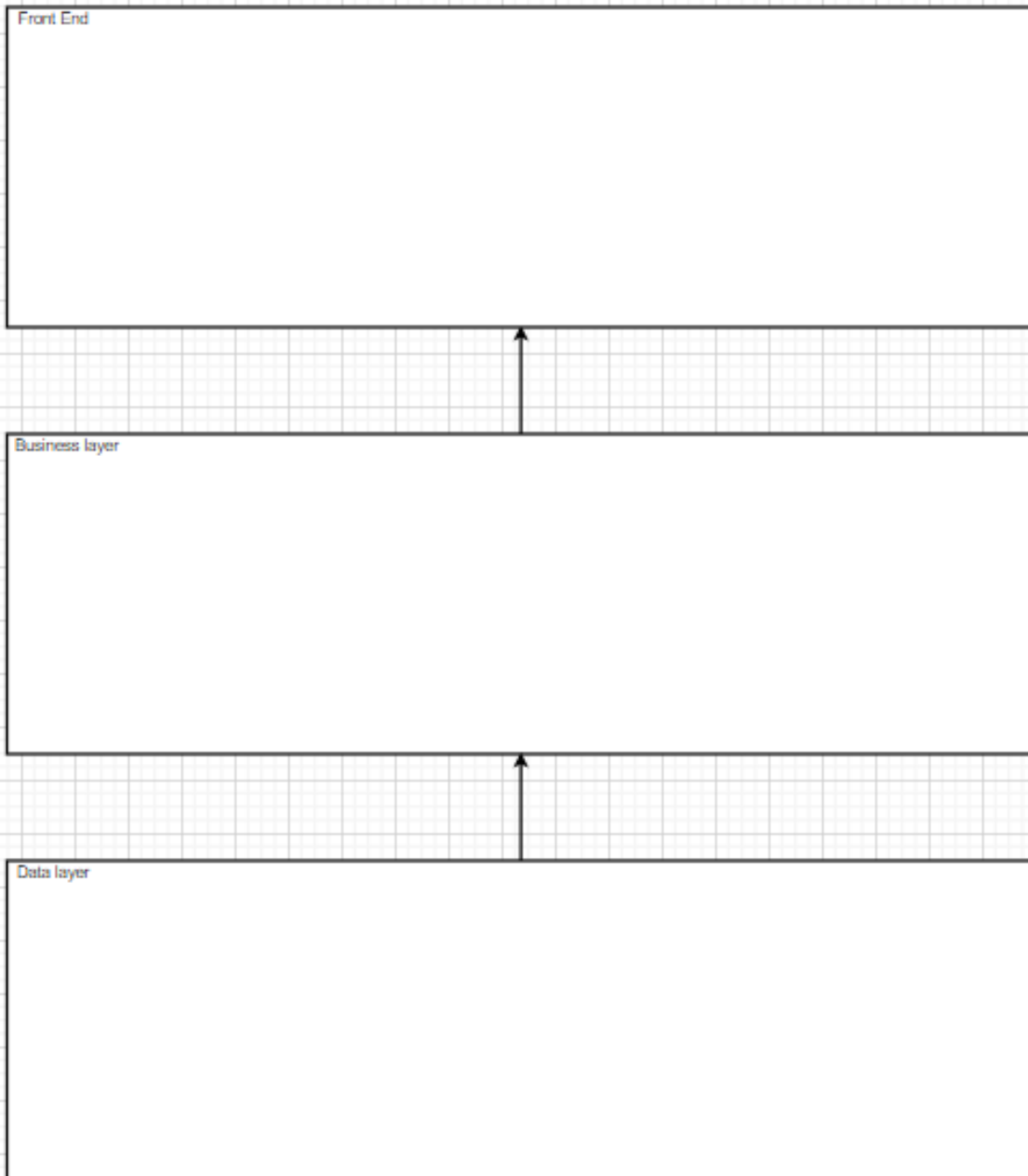


## Architectuur

Nog niet gemaakt omdat ik nog niet weet of mijn layers correct zijn.

### Architectuur Diagram

Architectuur diagram van mijn individuele project.

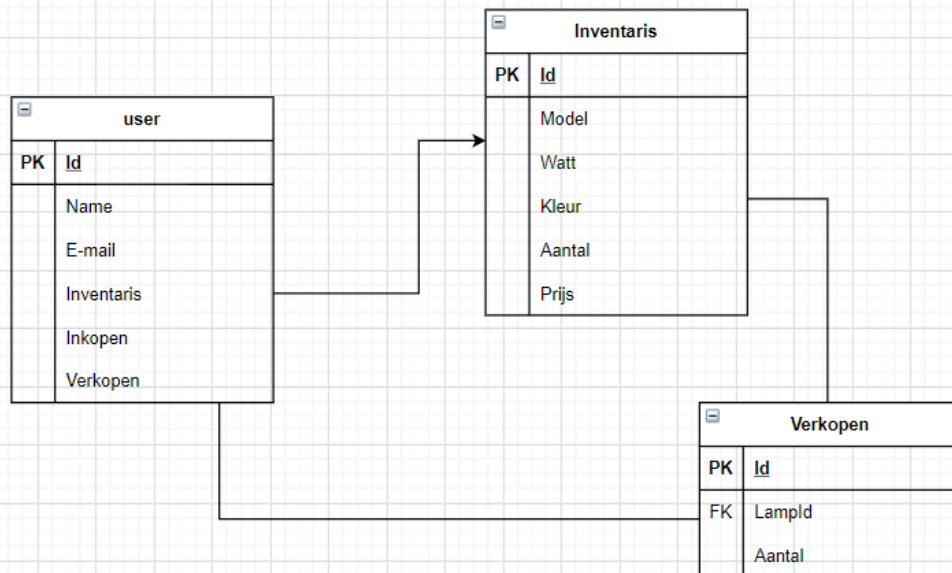


## Databaseontwerpen

Mijn Database ontwerp met een user.

### ER- Diagram

Database Schets voor mijn persoonlijk project



Mijn database ontwerp met de producenten en Kopers.

Database schets met 2 extra tabellen

