## PROJECTWERK PROGRAMMEREN MODULE 4

Cursist : Tim Quirynen

Afgewerkt op 08/02/15

**PROGRAMMEERTAAL : JAVA**

**PROGRAMMEERTOOL : NETBEANS**

**TYPE APPLICATIE : JAVA SWING APPLICATIE**

**ANDERE GEBRUIKTE TOOLS : USBWEBSERVER**

# Inhoudsopgave

1. Probleemomschrijving
2. Analyse
   1. USE CASE
   2. CLASS DIAGRAM
   3. SEQUENCE DIAGRAMS
      1. SEQUENCE VERTEGENWOORDIGER
      2. SEQUENCE PRODUCTIEMEDEWERKER/TECHNIEKER
3. Oplossingsbeschrijving
   1. Overzicht
   2. Database
      1. Overzicht
      2. “LABELFORMAT”
      3. “DEVICE”
      4. “FIELDS”
      5. “TEMPERATUREFIELDS”
      6. “SIZE”
      7. “IMAGEEXAMPLE”
   3. LIBRARY
      1. Overzicht
      2. Entities
      3. Services
      4. DAL
      5. Business Logic
   4. CODE FIRST
   5. SWING APPLICATIE
      1. Inleiding
      2. Werking
4. Problemen
5. Besluit

# 1.Probleemomschrijving

Digi Belgium is een bedrijf dat onder andere printende weegschalen levert.

Een deel daarvan wordt gebruikt om producten te labelen in een supermarkt, slager, bakker, ...

Een klant die nieuwe weegschalen aankoopt, bespreekt met de vertegenwoordiger hoe zijn weegschalen moeten worden ingesteld.

Momenteel wordt er aan de hand van verschillende Excel documenten de volledige setup doorgegeven.

Deze excel documenten worden door mij beheerd.

Dit is echter een tijdelijke oplossing, vermits Excel niet volledig geschikt is voor bepaalde zaken.

Een voorbeeld is de layout van het etiket dat de klant op zijn produkten kleeft.

De mogelijkheden zijn eindeloos :

* Er zijn verschillende afmetingen van etiket mogelijk, gaande van 40x46mm tot 80x120mm
* Er zijn tientallen velden die men kan afdrukken, bvb ingrediënten, barcode, houdbaarheidssdatum,....
* Deze zijn te combineren in honderden, misschien wel duizenden combinaties

Natuurlijk zijn er bepaalde combinaties die regelmatig terugkomen, wanneer de vertegenwoordiger opgeeft dat de klant een ‘standaard’ etiket wil, laden we bvb deze layout in het toestel :



In vele gevallen vraagt de klant echter een specifieke layout voor zijn etiket.

Dit vraagt veel werk voor productieafdeling, waar ik zelf ook werk.

Een etiketlayout from scratch programmeren vraagt heel wat ‘programmeerwerk’.

Daarom willen we van een ‘standaardbibliotheek’ vertrekken, waar de klant zijn keuze uit kan maken.

Deze formaten zijn reeds op voorhand geprogrammeerd, waar er heel wat tijd en werk bespaard wordt.

Op dat moment had ik een Word document aangemaakt met al de voorbeelden die ik op voorhand heb gemaakt.

In de praktijk bleek dit niet voldoende te zijn.

Bij vele orders vroeg men een layout die nog niet in de standaardbibliotheek voorkwam.

Dan kwam het idee van een bibliotheek die stelselmatig wordt uitgebreid.

En een applicatie om deze bibliotheek aan te roepen en te beheren.

Ook de vertegenwoordigers moeten deze applicatie kunnen gebruiken om samen met de klant zijn layout te kiezen.

Mijn baas was van in het begin tegen een web applicatie. Hij vroeg me om een Excel werkmap te maken, waar men de velden kon aanklikken die men nodig heeft.

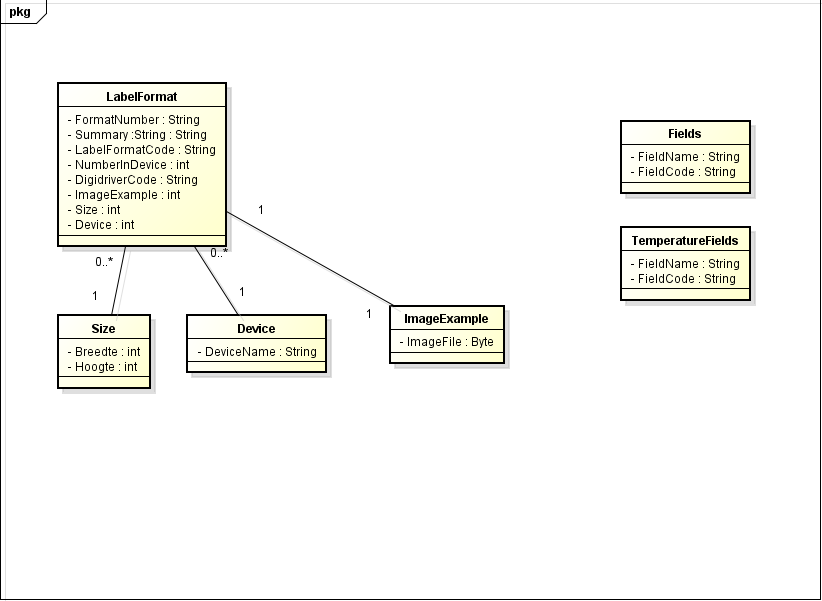
Dit is gelukt, maar Excel bleek niet altijd stabiel te werken:

* Vertegenwoordigers hebben soms niet voldoende pc-kennis,
* Op oudere Excel versies bleek de achterliggende VBA-code vast te lopen
* Een update van eind 2014 van Microsoft stuurde alles in de war
  + Macro’s werkten niet meer, waar ik telefonisch alle vertegenwoordigers moest uitleggen, hoe ze de update terug moesten verwijderen.
* Hoe meer VBA code die er in een excel werkmap zit, hoe zwaarder het wordt om te laden.
* Code schrijven in VBA is niet zo professioneel of overzichtelijk als in Java of C#

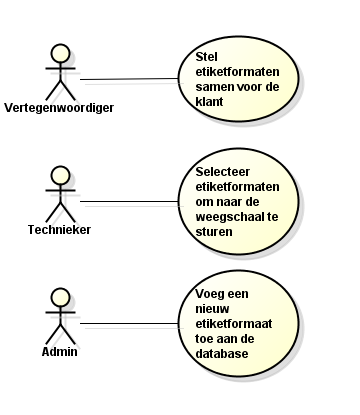
Daarom is een applicatie die zowel voor de vertegenwoordigers als voor de productiemedewerkers gemakkelijk te gebruiken is, een veel betere oplossing.

# 2. ANALYSE

## 2.1 CLASS DIAGRAM

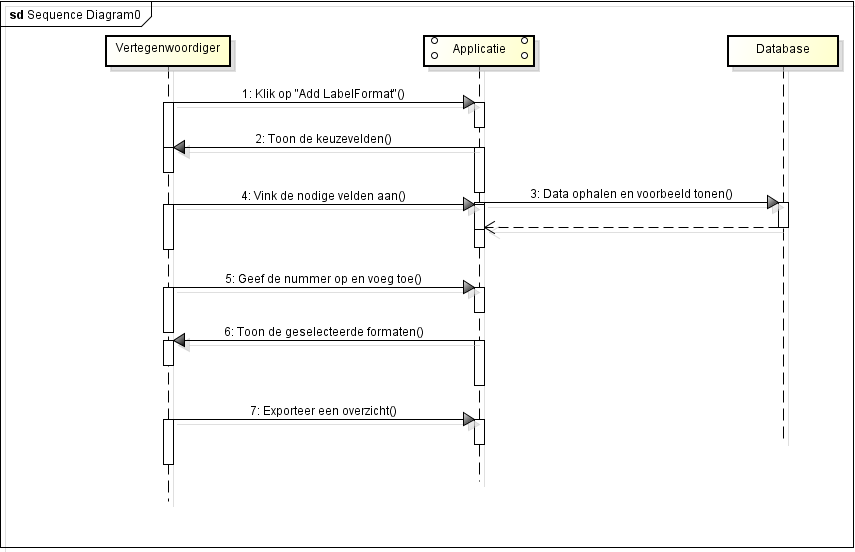


## 2.2 USE CASE DIAGRAM

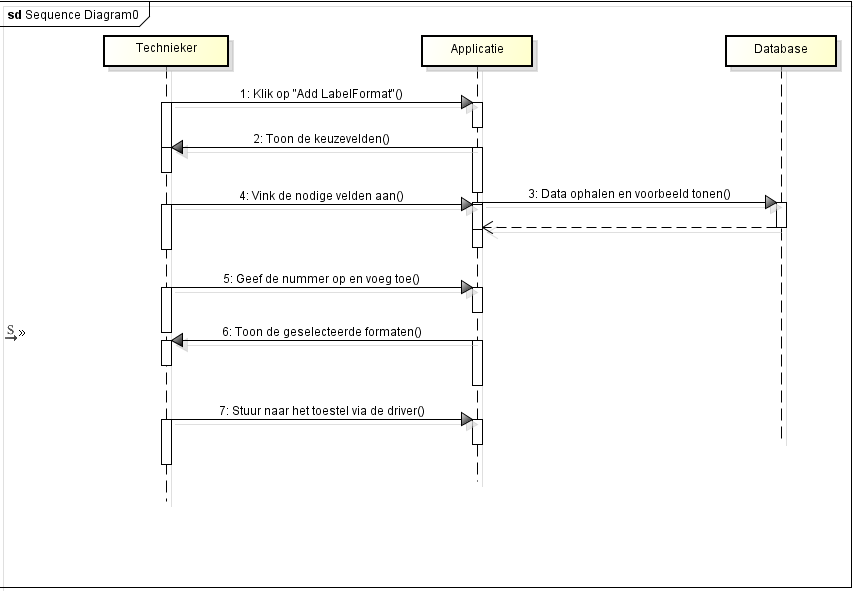


## 2.3 SEQUENCE DIAGRAMS

### 2.3.1 SEQUENCE VERTEGENWOORDIGER



### 2.3.2 SEQUENCE PRODUCTIEMEDEWERKER



# **3.OPLOSSINGSBESCHRIJVING**

## 3.1 Overzicht

De oplossing bestaat uit :

* Een database
* Een aparte library met :
  + Entities
  + Services
  + DAL
  + Business Logic
* Een code first applicatie
* Een Java swing applicatie

## 3.2 DATABASE

### 3.2.1 Overzicht

De database bestaat uit volgende tabellen :

* + LABELFORMAT
  + DEVICE
  + SIZE
  + FIELDS
  + TEMPERATUREFIELDS
  + IMAGEEXAMPLE

### 3.2.2 “LABELFORMAT”

* De etiketformaten met:
  + Afbeelding van het formaat
    - Verwijzend naar aparte tabel waar de afbeeldingen worden opgeslagen
    - ONE-TO-ONE RELATIE
  + Digidriverdata van het formaat, dit kan worden ingelezen door de weegschaal
  + Unieke code van het formaat, dit dient als referentie.
    - Bvb **Sm5500\_60x49\_InBcPdUdTfTt**
      * Dient voor een weegschaal uit de Sm5500 reeks
      * Afmetingen van het etiket zijn 60x49
      * De velden die op het etiket staan zijn
        + In : Ingrediënten
        + Bc : Barcode
        + Pd : Packed date (Inpakdatum)
        + Ud : Use by date (houdbaarheidsdatum)
        + Tf : Temperature Fixed (vaste bewaartemperatuur)
        + Tt : Two Temperatures (zowel een minimum als een maximum bewaartemperatuur)
      * Volgende velden staan zowiezo op al de formaten:
        + Artikelnaam
        + Winkelnaam
        + Eenheidsprijs
        + Totaalprijs
        + Aantal stuks / gewicht
      * Later kunnen deze velden ook toegevoegd worden
  + Device : het type toestel waarvoor dit formaat is gemaakt
    - verwijzend naar aparte tabel
    - MANY-TO-ONE RELATIE
  + Size : de afmetingen van het formaat
    - Verwijzend naar aparte tabel met breedte en hoogte
    - MANY-TO-ONE RELATIE

### 3.2.3 DEVICE

In deze tabel worden de verschillende toestellen bijgehouden

* + Sm5500 : “High end”
  + Sm5100 : “Low end”

De 2 types komen van verschillende fabrikanten

De Sm5500 kan veel meer aan, de Sm5100 is een low budget toestel

De digidriverdata voor beide toestellen komen niet 100% overeen.

### 3.2.4 FIELDS

Dit zijn de velden waarmee een combinatie kan gemaakt worden.

Ze bestaan uit een FieldCode en een FieldName

Bvb :

* + Id : 1 , FieldCode : “In”, FieldName : “Ingrediënten”
  + Id : 2 , FieldCode : “Bc”, FieldName : “Barcode”
  + …..
* De tabellen Fields en Temperaturefields zijn niet gelinked aan de tabel “Labelformats”
  + Er kunnen gemakkelijk velden bijgevoegd worden in de toekomst
  + De combinaties van velden die gebruikt zijn in een labelformaat worden aangeduid in de labelformatcode
  + Dit heb ik gedaan omdat anders de structuur van de database te complex zou worden

### 3.2.5 TEMPERATUREFIELDS

* + TemperatureFields heb ik een aparte tabel gezet
    - Deze worden anders benaderd dan de andere “Fields”.
    - Komen in een aparte listbox in de applicatie.
    - Er is slecht 1 combinatie mogelijk van een temperatuur:
      * Vaste maximumbewaartemperatuur
      * Variabele maximumbewaartemperatuur
      * Vaste min. en max. Bewaartemperatuur
      * Variabele min. en max. Bewaartemperatuur
    - Vast : alle artikelen die dit formaat gebruiken krijgen dezelfde temperatuur.
    - Variabel : de temperatuur wordt uit het artikel gehaald.

### 3.2.6 SIZE

* Hier worden de mogelijke afmetingen van de etiketten bewaard
* Hoogte en breedte

### 3.2.7 IMAGEXAMPLE

* Hier worden de afbeeldingen van de formaten bewaard

## 3.3 LIBRARY

### 3.3.1 Overzicht

* Entities
* Services
* DAL
* Business Logic

### 3.3.2 Entities

* Voor elke tabel in de database bestaat er een entity
* Buiten de velden die in de database zitten, zijn er een paar extra properties:

Labelformat :

* + formatNumber
    - dit wordt niet bewaard in de database
    - het formaatnummer wordt geselecteerd in de applicatie alvorens door te sturen. Meer info later.
  + Summary
    - Dit wordt niet bewaard in de database
    - Dit is een overzicht van de opbouw van het etiketformaat
    - Dit wordt gegenereerd door een methode in de BusinessLogic

### 3.2.3 Services

* Hier worden de SQL queries opgebouwd en uitgevoerd
* Bvb om een labelformaat op te vragen zijn er 3 parameters nodig
  + Device
  + Size
  + LabelformatCode

### 3.2.4 DAL

* De Data Acces Layer bestaat uit interfaces en klassen die deze interfaces implementeren
* In de interfaces wordt beschreven welke functionaliteit moet voorzien zijn
* In de klassen implementeren we deze functionaliteit
* Een voorbeeld in DALabelformats : FindBySelection

### 3.2.5 Business Logic

* Hier heb ik verschillende methodes geplaatst die nodig zijn om een applicatie rond de database te doen werken
  + AddImage :
    - Dit is een klasse die een image bestand kan openen en een array van bytes teruggeeft
  + CreateFormatFile :
    - Dit is een klasse die aan de hand van de lijst van geselecteerde formaten een digidriver data file opbouwt.
    - Hier worden ook de geselecteerde formaatnummers geplaatst in de digidriverdata van elk formaat
    - In de database staan zij standaard op “F1”
      * + In digidriverdatacode = AA0070
  + CreateSummaryFile :
    - Deze klasse kan aan de hand van de geselecteerde labelformaten

een overzicht aanmaken in tekstformaat van hoe de formaten zijn opgebouwd...

* + - Dit kan gebruikt worden door de vertegenwoordiger wanneer hij wil doorgeven welke formaten de klant wil
  + GenerateFormatCode:
    - Deze klasse genereert de labelformatcode aan de hand van selecties die de gebruiker heeft gemaakt
    - Vermits de listboxes in de applicatie hun gegevens rechtstreeks uit de database halen, komt de volgorde (index + 1) overeen.
    - Bvb wanneer de gebruiker in de listbox “Barcode” aanvinkt :
      * De index hiervan is 1, in de database telt men niet vanaf 0, dus daar is de ID : 2 (index + 1)

## 3.4 CODE FIRST

* Deze applicatie bouwt de database op, met behulp van de aparte library
* De tabellen worden aangemaakt en ingevuld
* De applicatie staat ingesteld op “drop and create”
  + Ik heb nog niet alle bestaande formaten toegevoegd
  + De structuur van de database kan nog wijzigen
* Er worden een 15 tal formaten al toegevoegd
  + Ik heb er al honderden, maar in deze ‘evaluatieversie’ zet ik ze nog niet allemaal erin

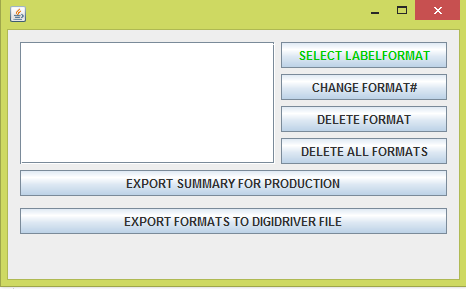
## 3.5 SWING APPLICATIE

### 3.5.1 Inleiding

* De Swing Applicatie kan door zowel de vertegenwoordiger als de productiemedewerker gebruikt worden
* De vertegenwoordiger kiest de etiketformaten die de klant nodig heeft
* Indien er geen afbeelding verschijnt, klikt de verkoper op “To Create” ipv “Ok”
* In het overzicht dat de productiemedewerker krijgt, kan hij zien dat het formaat nog gecreëerd moet worden
* De productiemedewerker kan starten van een formaat dat er dicht bij aan leunt, zodat hij/zij niet zo veel werk meer heeft.

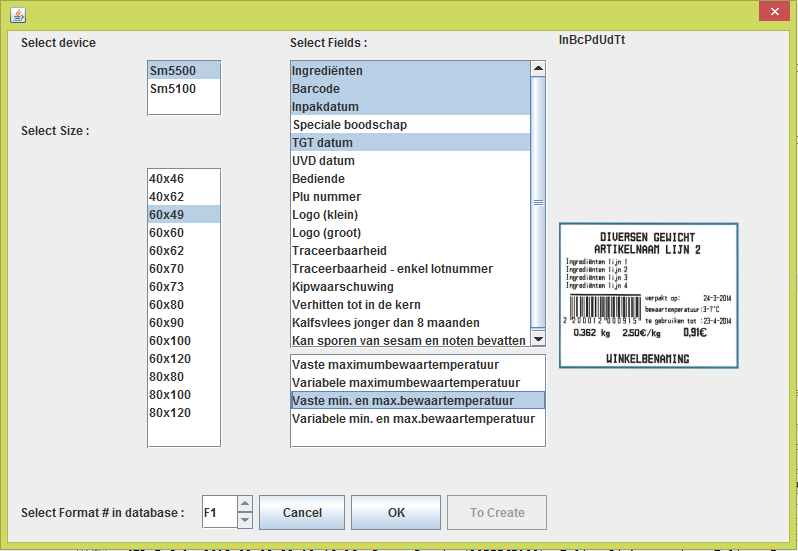
### 3.5.2 Werking :

Klik op “Select LabelFormat”



Maak je selectie:

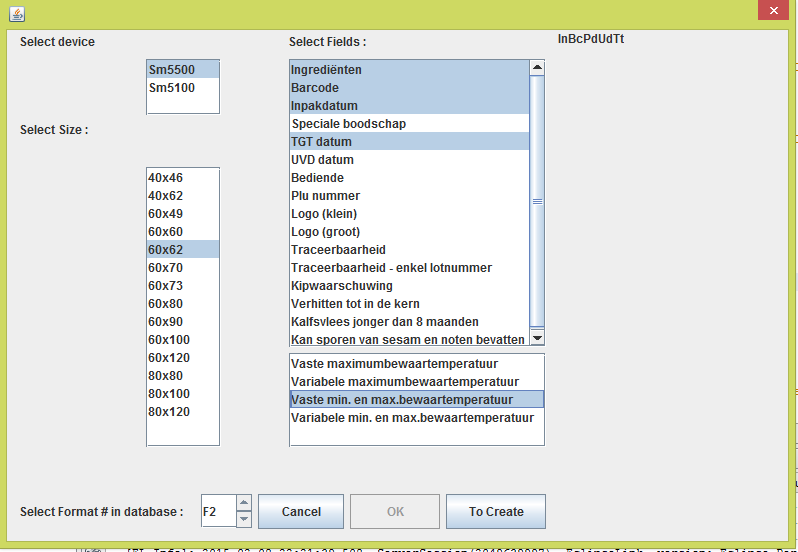
* Device : het type toestel
* Size : de afmetingen van het etiket
* Fields : Selecteer alle velden die je nodig hebt



Indien er een afbeelding verschijnt, betekent dit dat het labelformaat al reeds bestaat in de database en kan je bevestigen met “OK”

Indien er geen afbeelding verschijnt :

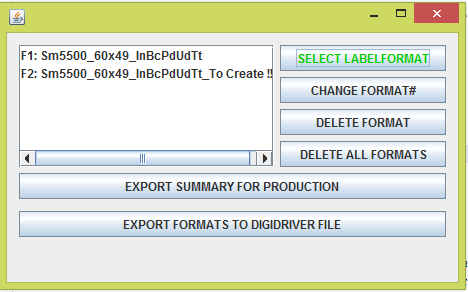
* + Vertegenwoordiger : klik op “To Create”
  + Productiemedewerker : zoek een formaat dat er dicht bij aanleunt en klik dan op “OK”



Nadat de formaten geselecteerd zijn :

* + Vertegenwoordiger : klik op “Export Summary For Production”
    - Je kan nu een tekstbestand bewaren met een overzicht van de gevraagde formaten, dit stuur je naar de administratie
  + Productiemedewerker: klik op “Export Formats to Digidriver File”

Deze file kan in de weegschaal geladen worden



# 4. PROBLEMEN

* Het bewaren van de etiketformaten in de database hou ik voorlopig voor mezelf
  + De productiemedewerkers hebben niet allemaal de kennis

van het het softwarepakket “Labeldesigner” waarmee ik de formaten aanmaak

* + Zij hebben ook meestal niet de tijd en de routine om dit te doen
  + Ik kijk op voorhand welke formaten er gevraagd worden voor orders
  + Ik werk voorlopig vanuit “Code First”
    - Nog niet alle formaten zijn toegevoegd
    - Ik wil de formaten zowel ‘los’ , in mapjes op de pc, als in de database
    - Zo kan ik de structuur van de database nog wijzigen later
* Het aantal velden van het formaat zal moeten uitgebreiden worden
  + Bvb sommige klanten willen een formaat zonder prijzen,
  + Een infolabel met ingrediënten en/of bereidingsadvies
  + Vanaf 2016 zullen voedingswaarden verplicht worden
    - Verschillende velden zoals:

Kalorieën, vet, zout , suiker,.

* Voorlopig heb ik een MySQL database gebruikt, omdat ik ook een WEB gedeelte wou maken, wegens gebrek aan tijd , is dit niet gelukt
* Een JavaDB database zou misschien beter zijn, omdat dit sneller is...
* Bijkomend probleem is dat wanneer ik een update op de database doe,

Bij elke vertegenwoordiger ook de diezelfde database moet overschreven

* + Een functie “update database” toevoegen

# 5.BESLUIT

* Dit project biedt een sterke basis, maar moet nog verder uitgewerkt worden.
* Dit project zal evolueren met de tijd.
* Ik kan dit testen en evalueren op de productieafdeling, alvorens dit op de pc van de vertegenwoordigers te zetten.
* Vermoedelijk ga ik dit project opnieuw maken in C#, omdat ik daar toch liever mee werk.