



Master of Science Industrial Science : Informatics  
Academic year 2013-2014

---

# VOP Project Dossier

---

Stambomen

Submitted on XX XXXX 2014

*Students:*

Kenzo Clauw  
Axl François  
Lowie Huyghe  
Sander Trypsteen  
Jelle Verreth



Master of Science Industrial Science : Informatics  
Academic year 2013-2014

---

# VOP Project Dossier

---

Stambomen

Submitted on XX XXXX 2014

*Student:*

Kenzo Clauw

Axl François

Lowie Huyghe

Sander Trypsteen

Jelle Verreth

## **Abstract**

# Contents

<b>1</b>	<b>Voorwoord</b>	<b>4</b>
<b>2</b>	<b>Introductie en situering</b>	<b>5</b>
2.1	Introductie . . . . .	5
2.2	Technische . . . . .	6
2.3	Situering . . . . .	7
<b>3</b>	<b>Taak-verdeling</b>	<b>8</b>
3.1	Methodologie . . . . .	8
3.2	Sprint 1 . . . . .	10
3.2.1	Kenzo Clauw . . . . .	10
3.2.2	Axl François . . . . .	10
3.2.3	Lowie Huyghe . . . . .	10
3.2.4	Sander Trypsteen . . . . .	10
3.2.5	Jelle Verreth . . . . .	10
3.3	Sprint 2 . . . . .	10
3.3.1	Kenzo Clauw . . . . .	10
3.3.2	Axl François . . . . .	10
3.3.3	Lowie Huyghe . . . . .	10
3.3.4	Sander Trypsteen . . . . .	10
3.3.5	Jelle Verreth . . . . .	10
3.4	Sprint 3 . . . . .	10
3.4.1	Kenzo Clauw . . . . .	10
3.4.2	Axl François . . . . .	10
3.4.3	Lowie Huyghe . . . . .	10
3.4.4	Sander Trypsteen . . . . .	10
3.4.5	Jelle Verreth . . . . .	10
3.5	Overzicht . . . . .	10
<b>4</b>	<b>Analyse</b>	<b>11</b>

<b>5</b>	<b>Interresantee Ontwerpbeslissingen</b>	<b>12</b>
<b>6</b>	<b>Interessante Implementatiekeuzes</b>	<b>13</b>
6.1	Bibliotheken . . . . .	14
6.1.1	Jersey . . . . .	14
6.1.2	Swagger Jersey . . . . .	14
6.1.3	SLF4J . . . . .	14
6.1.4	Junit . . . . .	14
6.1.5	Gedcom4j . . . . .	14
6.1.6	Sardine . . . . .	14
6.1.7	RestFB . . . . .	14
6.1.8	JCalendar . . . . .	14
6.1.9	Abego TreeLayout . . . . .	14
6.1.10	Java FX . . . . .	14
<b>7</b>	<b>Kwaliteitscontrole</b>	<b>15</b>
<b>8</b>	<b>Gekende problemen</b>	<b>16</b>
<b>9</b>	<b>Verbeterpunten en uitbreidingsmogelijkheden</b>	<b>17</b>
<b>10</b>	<b>Nabeschouwing en besluit</b>	<b>18</b>
	<b>List of Figures</b>	<b>19</b>
	<b>List of Tables</b>	<b>20</b>

# Chapter 1

## Voorwoord

Dit projectdossier is een schriftelijk neerslag van onze prestaties voor het vakoverschrijvend project<sup>1</sup>. Hierin bespreken we de verschillende facetten van ons project. Allereest zullen wij beginnen met de opdracht voor te stellen en ze te situeren. Hierbij zullen we proberen van een zo volledig mogelijk overzicht te geven van alle technische en niet-technische onderdelen. Vervolgens krijgt u meer uitleg over de manier waarop onze takenverdeling tot stand komt en het volledig overzicht van onze takenverdeling. Daarna bespreken we de analyse van het project aan de hand van enkele diagrammen en use cases. Vervolgens lichten we enkele belangrijke aspecten van ons project toe. We gaan hier zowel in op ontwerpbeslissingen als op implementatiekeuzes. Dan komen we aan de kwaliteitscontrole voldoet onze applicatie aan de functionele en niet-functionele vereisten zoals deze zijn opgesteld door de klant? Uiteraard zijn er in elk project wel problemen en we zullen het niet na laten om deze te bespreken. Als voorlaatste geven we nog even onze visie op het verdere verloop van de applicatie. Waar kan het beter, welke uitbreidingen zijn mogelijk. Ten slotte eindigen we met een nabeschuiving en ons besluit over het VOP.

---

<sup>1</sup>VOP

# Chapter 2

## Introductie en situering

### 2.1 Introductie

In deze paper kan u meer informatie vinden rond ons vop. Het onderwerp zoals u wellicht al gelezen hebt op het titel blad zijn stambomen. Het woord stambomen kan in verschillende contexten bekeken worden. We beperken ons echter enkel tot de genealogie. In genealogie zitten twee Griekse woorden verborgen namelijk Genea en Logos. Genea staat voor afkomst of afstamming en logos betekent wetenschap of kennis. Dus we bestuderen hier de afkomst, afstamming van een persoon. In dit project zijn er echter twee beperkingen:

Zonder dieper in te gaan op het onderwerp genealogie willen ook nog melden dat in dit project we ons slechts zullen beperken tot natuurlijk mogelijke afstammelingen. Hieronder verstaan wij dat er enkel man - vrouw relaties mogelijk zijn. We houden dus geen rekening met man - man of vrouw - vrouw relaties die dan kinderen zouden hebben.

Verder houden we ook geen rekening met de onderlinge relaties tussen echt-paren. Op zich zijn dat onze zaken niet maar wanneer een persoon een buitenechtelijke kind heeft dan zal hij dat niet kunnen invoeren in ons programma. Hieronder vallen dus ook half-broers of half-zussen. Deze zullen niet weergegeven worden in ons stamboom overzicht.



Nu de scope van de opdracht duidelijker is kunnen ingaan op wat er gevraagd is. Het is de bedoeling om een applicatie te maken waar een gebruiker stambomen kan ingeven en bekijken. De gebruiker moet deze stambomen kunnen raadplegen via een website en de bomen worden ingegeven via een desktop applicatie. Verder moet de gebruiker ook zijn bomen kunnen delen met zijn vrienden zowel binnen de applicatie als op sociale netwerk sites. Voor dit project zullen we ons qua sociale netwerk sites beperken tot Facebook.

## 2.2 Technische

We hebben op 10/02/2014 de inleidende presentatie van het vop gekregen. Hierin werden een aantal vereisten uitgelegd. Deze vereisten staan uitgebreid beschreven in VOP: de richtlijnen <sup>1</sup>. De belangrijkste hierin zijn dat er een docent de rol van klant zal spelen. Er zullen afspraken moeten worden gemaakt met de klant over wat al dan niet mogelijk is. Wat er moet gerealiseerd worden? We kwamen ook te weten dat het project in Java moest geschreven worden. Verder moest dit gebeuren door aan de hand van een drielagen architectuur.

**Presentatie laag** Deze laag bestond uit twee onderdelen namelijk een Java Desktop Client gemaakt in Java Swing en een website (HTML5, Javascript) die gebruikt maakt van Java Servlets. De nadruk uiteraard ligt hier bij het scheiden van onze logica van de presentatie.

**Domein logica** Hierbij was het de bedoeling om een REST-service in Java op te zetten waarvan de presentatie gebruik kan maken. We hebben hier van in het begin gekozen om Jersey te gebruiken.

**Data tier** Een relationele databank ging in staan voor de persistentie van onze gegevens. De databank die ons aangeboden werd door UGent is MySQL. We mochten in deze opdracht geen gebruik maken van ORM-tools<sup>2</sup>.

---

<sup>1</sup>ADD REFERENCE

<sup>2</sup>Object-Relational Mapping zoals Hibernate.

## 2.3 Situering

WAT?

# Chapter 3

## Taak-verdeling

### 3.1 Methodologie

Tijdens dit project zullen we gebruik maken van Scrum. Hierbij verdelen we het project in drie sprints. Aan het begin van elke sprint zitten we samen om te bespreken welke features we graag zouden opnemen. Hoe lang een bepaalde feature duurt om uit te werken. Na deze bespreking maken we een afspraak bij de klant.

Met de klant bespreken we dan welke features voor ons mogelijk zijn en welke features voor de klant noodzakelijk zijn. Hieruit volgt een contract, alle features die we tijdens een bepaalde sprint zullen realiseren.

Op het einde van de sprint kijken we dan welke features we gerealiseerd hebben en welke niet. Dankzij deze methode kunnen we inschatten of we tijdens het project moeten bijsturen. Het kan zijn dat we niet alle features gerealiseerd hebben dan moeten we harder werken tijdens de volgende sprint.

Tijdens elke sprint krijgen we van de docenten een voorgestelde lijst met features.



## **3.2 Sprint 1**

**3.2.1 Kenzo Clauw**

**3.2.2 Axl François**

**3.2.3 Lowie Huyghe**

**3.2.4 Sander Trypsteen**

**3.2.5 Jelle Verreth**

## **3.3 Sprint 2**

**3.3.1 Kenzo Clauw**

**3.3.2 Axl François**

**3.3.3 Lowie Huyghe**

**3.3.4 Sander Trypsteen**

**3.3.5 Jelle Verreth**

## **3.4 Sprint 3**

**3.4.1 Kenzo Clauw**

**3.4.2 Axl François**

**3.4.3 Lowie Huyghe**

**3.4.4 Sander Trypsteen**

**3.4.5 Jelle Verreth**

## **3.5 Overzicht**

## **Chapter 4**

### **Analyse**

## **Chapter 5**

# **Interresantee Ontwerpbeslissingen**





## Chapter 6

# Interessante Implementatiekeuzes

### 6.1 Bibliotheken

#### 6.1.1 Jersey

#### 6.1.2 Swagger Jersey

#### 6.1.3 SLF4J

#### 6.1.4 Junit

#### 6.1.5 Gedcom4j

#### 6.1.6 Sardine

#### 6.1.7 RestFB

#### 6.1.8 JCalendar

#### 6.1.9 Abego TreeLayout

#### 6.1.10 Java FX

## **Chapter 7**

### **Kwaliteitscontrole**

## **Chapter 8**

### **Gekende problemen**

## **Chapter 9**

### **Verbeterpunten en uitbreidingsmogelijkheden**

## **Chapter 10**

### **Nabeschouwing en besluit**

## List of Figures

## List of Tables