



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

Vakoverschrijdend Eindproject

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

Vakoverschrijdend Eindproject

Stambomen

Submitted on XX XXXX 2014

Student:

Kenzo Clauw

Axl François

Lowie Huyghe

Sander Trypsteen

Jelle Verreth

Abstract

Contents

1	Voorwoord	4
2	Introductie en situering	5
2.1	Introductie	5
2.2	Technische	6
2.3	Situering	7
3	Taak-verdeling	8
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
4	Analyse	11

5	Interresantee Ontwerpbeslissingen	12
6	Interessante Implementatiekeuzes	13
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
7	Kwaliteitscontrole	15
8	Gekende problemen	16
9	Verbeterpunten en uitbreidingsmogelijkheden	17
10	Nabeschouwing en besluit	18
	List of Figures	19
	List of Tables	20
A	Appendix	21
A.1	Opdracht sprint 1	22
A.2	Opdracht sprint 1	23
A.3	Opdracht sprint 2	24
A.4	Opdracht sprint 2	25

Chapter 1

Voorwoord

Dit projectdossier is een schriftelijk neerslag van onze prestaties voor het vakoverschrijvend project¹. Hierin bespreken we de verschillende facetten van ons project. Allereerst 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.

¹VOP

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 ¹. 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-tools².

¹ADD REFERENCE

²Object-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. Deze lijsten kan je vinden als appendix bij dit document.

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

Appendix A

Appendix

A.1 Opdracht sprint 1

STAMBOMEN

DOEL

Door van elke persoon de (biologische) vader en moeder bij te houden, worden boomstructuren opgebouwd. Hierbij kan het voorkomen dat deze ouders onbekend zijn, hetzij omdat niet geweten is wie zij zijn, hetzij omdat de gebruiker de gegevens nog niet heeft ingevoerd.

Verder houdt het systeem van elke persoon de voor- en familienaam, het geslacht, de geboortedatum en de geboorteplaats bij. De geboortedatum en -plaats kunnen daarbij ontbreken.

Elke gebruiker kan verschillende stambomen beheren. Dit betekent dat hij/zij als enige de structuur en persoonsgegevens kan wijzigen; dit is het *bewerken* van de stamboom. Bij het bewerken worden enkel de in de huidige stamboom aanwezige personen weergegeven.

Bij het *consulteren* daarentegen voegt het systeem alle beschikbare stamboomgegevens samen, om zo tot een zo volledig mogelijk overzicht te komen. Indien een persoon met identieke gegevens in meerdere bomen voorkomt, wordt daarbij aangenomen dat het om dezelfde persoon gaat. Er wordt gestart van de stamboom van de huidige gebruiker; dit noemen we de *referentiepersoon*. Men kan een andere referentiepersoon selecteren door deze aan te klikken in de stamboom.

Een gebruiker kan bij elke stamboom aangeven of hij deze wil *delen* met andere gebruikers van de applicatie. Is dit niet het geval, dan wordt de inhoud ervan verborgen bij het consulteren. Er kan worden gedeeld met iedereen, met niemand, of enkel met de vrienden van de gebruiker. Voor deze laatste mogelijkheid kunnen gebruikers vriendenlijsten aanleggen.

Naast de klassieke boomweergave is er ook de *teletijdmachine*, een wereldkaart met tijdsaanduiding. Hierop worden de geboorteplaatsen gevisualiseerd van de voorouders en het nageslacht van een persoon naar keuze. Men kiest bovendien een datum; enkel de familieleden die op die datum in leven waren, worden opgenomen in de visualisatie.

De applicatie wordt deels als webapplicatie, deels als desktopapplicatie aangeboden. Overleg daarom met de klant onder andere over welke componenten zich waar bevinden.

De lijst met features is een voorstel. Ken er story points aan toe en maak een planning op om met de klant tot een consensus te komen over het contract voor sprint 1. Implementeer enkel het gevraagde.

A.2 Opdracht sprint 1

SPRINT 1

1. De gebruiker maakt een account aan.
2. De gebruiker logt in op de webapplicatie.
3. De gebruiker logt in op de desktopapplicatie.
4. De gebruiker maakt een nieuwe stamboom aan.
5. De gebruiker opent een bestaande stamboom.
6. De gebruiker voegt een persoon toe aan de huidige stamboom.
7. De gebruiker wijzigt de gegevens van een persoon in de huidige stamboom.
8. De gebruiker verplaatst een persoon in de huidige stamboom.
9. De gebruiker verwijdert een persoon uit de huidige stamboom.
10. De gebruiker importeert een GEDCOM-bestand.
11. De gebruiker verstuurt een vriendschapsverzoek.
12. De gebruiker aanvaardt een vriendschapsverzoek.
13. De gebruiker weigert een vriendschapsverzoek.
14. De gebruiker bekijkt zijn vriendenlijst.
15. De gebruiker verwijdert iemand uit zijn vriendenlijst.
16. De gebruiker consulteert zijn samengestelde stamboom.
17. De gebruiker bekijkt de gegevens van een persoon in een stamboom.
18. De gebruiker wijzigt de referentiepersoon van een stamboom.
19. De gebruiker kiest de referentiepersoon voor de teletijdmachine.
20. De gebruiker kiest de datum voor de teletijdmachine.
21. De teletijdmachine markeert de geboorteplaatsen van de familieleden die op de ingevoerde datum in leven waren.
22. Het systeem bevat een grote hoeveelheid stamboomgegevens.

A.3 Opdracht sprint 2

STAMBOMEN

DOEL

Net als bij sprint 1 is de lijst met features een voorstel. Voeg deze samen met de eventuele features die in sprint 1 nog niet (volledig) werden gerealiseerd. Deel opnieuw de features op in taken en ken er story points aan toe. Stem af met de klant om zo snel mogelijk tot een definitief contract te komen.

SPRINT 2

1. De gebruiker wijzigt de weergavetaal.
2. Het systeem houdt een overzicht bij van relevante gebeurtenissen.
3. De gebruiker voegt een foto toe aan een persoon.
4. De gebruiker verwijdert een foto van een persoon.
5. De gebruiker zoomt in op de stamboom. Naarmate meer ruimte beschikbaar is, worden meer gegevens weergegeven bij een persoon.
6. De gebruiker koppelt een persoon aan zijn/haar Facebookaccount.
7. De gebruiker doorzoekt de voor hem beschikbare stambomen.
8. De gebruiker geeft aan dat twee personen uit verschillende stambomen aan elkaar gelijk zijn.
9. Het systeem detecteert gelijkaardige personen uit verschillende stambomen en meldt de gelijkenissen aan de betrokken gebruikers.
10. De gebruiker bevestigt dat twee personen aan elkaar gelijk zijn.
11. De gebruiker geeft aan dat twee personen niet aan elkaar gelijk zijn.
12. De gebruiker doorzoekt de openbare gebruikerslijst.
13. De gebruiker bekijkt een openbaar profiel.
14. De gebruiker maakt zijn/haar profiel openbaar.

A.4 Opdracht sprint 2

15. De gebruiker geeft aan of een stamboom wordt gedeeld met alle gebruikers van de applicatie, enkel met zijn/haar vrienden, of met niemand.
16. De gebruiker registreert zich met zijn/haar Facebookaccount.
17. De gebruiker koppelt zijn/haar account aan zijn/haar persoonlijke Facebookaccount.
18. De gebruiker logt in met Facebook.
19. De gebruiker deelt stamboomgegevens op Facebook.
20. De gebruiker voegt een Facebookvriend toe aan zijn/haar vriendenlijst.
21. De gebruiker bekijkt een overzicht van de activiteiten van zijn/haar vrienden.
22. De moderator wijzigt de gegevens van een persoon.
23. De moderator wijzigt het profiel van een gebruiker.
24. De moderator blokkeert tijdelijk het account van een gebruiker.
25. De gebruiker bekijkt een animatie van de teletijdmachine.
26. De gebruiker vertraagt of versnelt de teletijdmachine.
27. De gebruiker pauzeert de teletijdmachine.
28. De gebruiker navigeert naar een tijdstip met de teletijdmachine.
29. De gebruiker wijzigt het thema van de applicatie.
30. De ontwikkelaar wijzigt de huisstijl van de applicatie.