

C.A.S.A.M.-project Eindverslag

S.E.F. van Berkel (1262882)	B. Bijl (1312405)
J.Y.T. den Hollander (1268538)	S. Rabbelier (1308211)
B.M.W. Sedee (1263153)	N.N. Smit (1286455)

June 5, 2009

Contents

Voorwoord	4
1 Samenvatting	5
2 Inleiding	6
3 Probleemstelling en Analyse	7
3.1 Probleemstelling	7
3.2 Analyse	7
4 Planning	9
4.1 Voorgenomen planning	9
4.2 Werkelijke planning	9
5 Aanpak	10
5.1 Project gerelateerd	10
5.2 Database systeem	10
5.3 Image Processing	11
6 Implementatie	12
6.1 Project gerelateerd	12
6.2 Database systeem	12
6.3 Image Processing	12
7 Aanbevelingen	13
7.1 Bestandstypen van foto's	13
7.2 Verschillende kleuren in flash applicatie	13
7.3 Opslaan van een lege bitmap	14
7.4 Te grote foto's inladen in de flash-applicatie	14
7.5 opslaan van statistische data bij landmarks	14
7.6 analyse van deze statische data	14
7.7 uitgebreid user management en arts-view mode	14
7.8 morphen naar standaard AnatomicalView	14
7.9 uitwerken afstand meet applicatie	14
7.10 zoom op meerdere niveau's	14
7.11 schaalweergave in image	14
7.12 export on unix import on windows	14
8 Conclusie	15
9 Acknowledgement	15

10 Bijlagen	16
10.1 Oriëntatieverslag	16
10.2 Plan van Aanpak	16
10.3 Requirements Analysis Document	16

Voorwoord

1 Samenvatting

2 Inleiding

De afgelopen tien weken hebben wij met elkaar gewerkt aan een webapplicatie voor het C.A.S.A.M.-project . Voor het vervullen van deze opdracht hebben wij op diverse terreinen verschillende uitdagingen moeten overwinnen. Dit ging van de communicatie met experts van het Erasmus Medisch Centrum die geen verstand hadden van de IT-wereld, tot aan het werken met programmeertalen waar niemand ervaring mee had. Uiteindelijk hebben we een product gebouwd dat vele verschillende programmeertalen bij elkaar brengt en op deze manier gebruiken we de verschillende krachten in een nieuwe combinatie. Ook hebben wij het idee dat het product voldoet aan de eisen van het C.A.S.A.M.-project . Dit resultaat hebben wij ook te danken aan het Agile programmeerprincipe waardoor wij veel tussentijds contact hebben gehad met de opdrachtgever en de uiteindelijke gebruikers.

In dit verslag besteden wij aandacht aan de verschillende uitdagingen die voor ons lagen, de oplossingen die we daarvoor hebben gevonden en het research wat we daarvoor hebben moeten doen. In sectie 2 leggen we uit wat de probleemstelling is en daaraan gekoppeld wordt in sectie 3 een analyse gemaakt van het probleem en een oplossingsrichting gedefinieerd. In sectie 4 zullen we ingaan op de planning die we hebben gemaakt, om in sectie 5 in te gaan op de aanpak van ons probleem. Daarna zullen we in sectie 6 een toelichting geven op onze implementatie en een aantal bijzondere aspecten uitlichten. Om daarna in sectie 7 een analyse te maken over de mogelijkheden om het product uit te breiden en te verbeteren.

Bij dit verslag horen enkele document welke zijn toegevoegd als bijlage. Voorbeelden hiervan zijn het Plan van Aanpak, het orientatieverslag en het RAD-document van de database applicatie. In het verslag zal op verschillende momenten worden verwezen naar deze documenten.

3 Probleemstelling en Analyse

3.1 Probleemstelling

Het CASAM-project was op zoek naar een applicatie die hun kon helpen bij de anatomische vraagstukken die zij hadden. Het ging hierbij vooral om hulp bij het warpen en morphen van foto's en de mogelijkheden om op verschillende foto's bijzonderheden aan te kunnen geven en deze met elkaar te vergelijken. Al in het eerste gesprek bleek dat er op dat moment nog geen enkele applicatie lag en dat alle foto's gewoon werden opgeslagen op een harde schijf en dat er out-of-the-box software werd gebruikt voor de foto manipulatie. Op dat moment hebben wij voorgesteld om naast de foto manipulatie ook aandacht te besteden aan de administratie en opslag van het materiaal. Tenslotte kwamen zij ook met de vraag of het mogelijk was om bepaald materiaal dat in het ziekenhuis werd gemaakt, zoals rntgenfoto's, direct te importeren in het systeem. Ook wilde zij graag publicaties die hoorde bij een project erbij willen opslaan.

3.2 Analyse

In overleg met het CASAM project hebben we toen afgesproken om het project in drie delen op te delen: een database applicatie, Image processing en tenslotte de koppeling met andere systemen. Daarnaast werd meteen afgesproken dat de project groep op de TU Delft zou werken, dit omdat de facilitaire mogelijkheden in Delft beter zijn dan op het EMC. Om wel voldoende feedback te blijven ontvangen is er gekozen voor een Agile programmeerprincipe, op deze manier zou de opdrachtgever voldoende betrokken blijven bij het proces.

In de eerste week van het project hebben wij een lijst opgesteld van de verschillende functionaliteiten die in het systeem moesten komen. Van te voren was al duidelijk dat niet alle functionaliteiten gebouwd konden worden binnen de scope van dit project. Daarom zijn in samenspraak met de opdrachtgever de functionaliteiten van prioriteiten voorzien, uiteindelijk is daar een diagram uitgekomen, zie hiervoor bijlage A.

Mede op basis van het prioriteiten diagram hebben wij de verschillende uitdagingen gedefinieerd:

- Project gerelateerd
 - Gebrek aan medische en anatomie kennis
 - Gebrek van IT kennis aan de kant van de opdrachtgever
 - Samenwerking in een grote groep
- Database systeem
 - Gebruikersvriendelijkheid
 - Van diverse locaties bereikbaar
 - Eenvoudige manier om data toe te voegen aan het systeem

- Schaalbaarheid
- Image processing
 - Variaties in landmarks
 - Morphen van verschillende foto's naar n gemiddelde foto
 - Verplaatsen en roteren van foto's naar standard AnatomicalView

In de volgende secties lichten wij toe hoe we met deze uitdagingen zijn omgegaan.

4 Planning

We begonnen het project met het maken van een planning voor de komende 10 weken zodat er voor zowel de opdrachtgever als onszelf duidelijk was wanneer er wat gedaan zou zijn. Ivm met de Agile methoden zijn de eerste paar weken in meer detail geplande dan de latere weken met de bedoeling om de planning steeds bij te werken voor de komende (paar) weken. Deze 'voorgenomen' planning wordt besproken in paragraaf 4.1. Zoals verwacht is deze ruwe planning aangepast in de weken daarna, deze planning wordt besproken in 4.2.

4.1 Voorgenomen planning

Doordat we in het begin slechts een grove planning hebben gemaakt waren vooral de eerste paar weken in detail uitgewerkt. Er moesten twee 'documenten' af zijn voordat er met de implementatie van het product kon worden begonnen, het 'plan van aanpak' (PvA) en het 'requirement analysis document' (RAD). Tevens hebben we ons direct voorgenomen een 'demo' in de eerste weken van het project af te hebben zodat de opdrachtgever snel feedback kan geven op de richting waar het project in gaat.

Plan van Aanpak Tijdens de eerste meeting is het volgende schema voor het PvA afgesproken, waarbij de deadlines zijn afgesproken met de gelegenheid om verbeteringen door te voeren in het achterhoofd.

09/04: Concept PvA mailen 14/04: Verwerken feedback PvA 15/04: Meeting in erasmus 17/04: PvA af

Requirement Analysis Document Tevens is voor het RAD een (minder granulaire) plannin gemaakt, deze overlapt met de planning voor het PvA aangezien het schrijven van een nieuw document en het verwerken van verbeteringen aan een ander document samen kunnen gaan.

Nadat de tweede ronde van feedback op het PvA is verwerkt zal worden begonnen met het RAD, zodat deze zo snel mogelijk af is. Eventuele feedback op het PvA hierna wordt eerst verwerkt (werk aan het RAD moet dan wachten).

27/04: RAD af

Demo Om een degelijke demo te kunnen geven waarin zo veel mogelijk geëvalueerd kan worden was besloten om twee weken na het inleveren van het RAD te besteden aan het maken van 'spike solutions' en deze te integreren in een simpele applicatie die deze functionaliteit showcased.

4.2 Werkelijke planning

In deze sectie geven wij aan welke planning wij hadden en hoe deze aansluit op de uiteindelijke realisatie.

5 Aanpak

Een van de eerste beslissingen die moest worden genomen is de architectuur van het systeem. Al snel bleek dat de opdrachtgever een voorkeur had voor een webapplicatie die door het hele EMC op het netwerk simpel en snel gebruikt kon worden. Voor onze Image Processing kregen wij het advies van Dr. Botha om een aantal Python bibliotheken te bekijken en deze indien mogelijk te gebruiken.

De combinatie van deze feiten leidde ons richting Django, dit is een python framework voor het bouwen van websites in Python. Na wat research en de ervaringen die één van de projectleden al had met Django hebben we ervoor gekozen om Django te gaan gebruiken.

Voor wat betreft de visuele interactie hebben we gekeken naar verschillende Javascript bibliotheken, mede door de uitgebreide ervaringen van diverse leden hebben we uiteindelijk gekozen voor de combinatie van Prototype en Scriptaculous.

Tijdens het project bleek uiteindelijk dat er behoefte was aan een tekenapplicatie binnen de omgeving en deze was niet te realiseren via Javascript en is er uiteindelijk ook nog een Flash applicatie geschreven welke communiceert met Javascript.

5.1 Project gerelateerd

In de begin periode hebben wij ons ingelezen op de basis anatomische kennis die nodig was om de experts van het EMC te kunnen volgen. Hierbij hebben we vooral veel gebruik gemaakt van de kennis van een van de groepsleden. Ook zijn we een dag langs geweest bij het EMC en hebben daar een rondleiding gehad door het laboratorium en de snijzalen waar onderzoek wordt gedaan.

Een andere uitdaging was de relatief grote groep studenten voor dit project. Om te voorkomen dat we elkaar te veel in de weg zouden zitten hebben we de diverse delen van het project opgesplitst en de taken duidelijk verdeeld. Daarnaast hebben we gebruik gemaakt van software zoals Trac, dit is een Project Management Systeem. Daarnaast hadden we een locatie geregeld op de TU Delft waar we iedere dag hebben gewerkt en waar we elke ochtend een werkoverleg konden hebben voor wat er die dag zou gebeuren. Hierdoor bleef iedereen goed op de hoogte van de status van het hele project.

5.2 Database systeem

Zoals hierboven al aangegeven zouden we gaan werken met Django en moest het beschikbaar worden als webapplicatie. Django is een MVC opgebouwde taal met een duidelijk abstractie laag voor het database gedeelte. Na diverse brainstorm sessies zijn we uiteindelijk uitgekomen op een klasse diagram wat de basis zou gaan vormen voor het database systeem. Het diagram staat in bijlage B. Voor de interface hebben we ervoor gekozen om veel gebruik te maken van Javascript en zijn bibliotheken. Dit gaf ons eenvoudige toegang tot diverse visuele effecten, zoals drag en drop, fade en de sliders). Daarnaast konden

we met behulp van Javascript gebruik maken van de AJAX techniek, hiermee konden server requests sturen en de pagina updaten zonder dat de gebruiker hier last van had.

5.3 Image Processing

Dit was een van de grootste uitdagingen van het project. Niemand had ervaring met de verschillende bibliotheken of uitgebreide ervaring met Image Processing in het algemeen. Uiteindelijk is er, mede op advies van Dr. Botha, gekozen voor de VTK bibliotheek. Al snel bleek dat er veel research nodig was om dit te kunnen gaan gebruiken en zijn we lang bezig geweest met het bekend raken met de begrippen over Point Distribution Model, Thin Plane Spline Transformatie, etc.

6 Implementatie

In de tweede week van het project zijn we echt begonnen met het bouwen van het systeem. We zijn begonnen met *spiked solutions* van verschillende basis functionaliteiten, zoals het uploaden van images en het plaatsen van landmarks geplaatste images met behulp van drag & drop. In een later stadium hebben we voor de grotere uitdagingen *spiked solutions* geschreven, hieronder vallen bijvoorbeeld de Image morphing. Als een *spiked solutions* werkte werd deze aan het bestaande systeem gekoppeld, op deze manier hadden we in een vrij korte tijd een basis systeem met haar basis functionaliteiten. En konden we ons concentreren op de Javascript functies welke zouden gaan zorgen voor de AJAX afhandeling en de visuele effecten.

6.1 Project gerelateerd

Daar waar het mogelijk en handig was hebben we geprogrammeerd in tweetallen. Tijdens de implementatie fase hebben we veel gebruik gemaakt van de beschikbare whiteboards, deze waren handig voor het brainstormen en als iemand vast liep dan kon het probleem getekend en uitgelegd worden. Dit alles zorgde voor een goede synergie en hield iedereen de motivatie die hij had goed vast. Ook tijdens de implementatie fase hebben we veel contact gehad met de leden van het C.A.S.A.M.-project en zijn er verschillende tussentijdse presentaties geweest. Steeds hebben we na deze presentaties een uitgebreid overleg gehad over de gang van zaken en gevalueerd over hoe het project er voor stond. In sommige gevallen hebben we ook onze prioriteitenlijst aangepast aan de hand van deze vergaderingen.

6.2 Database systeem

Lorem ipsum

Voorbeeld van de implementatie van de bitmap creator python-javascript-html-flash-javascript-python

Op het eind van het project hadden we veel bereikt. Maar ook is er werk blijven liggen. Voorbeelden hiervan zijn: de ViewMode voor artsen en de analyse van statische data.

6.3 Image Processing

Na heel veel research en veel theoretische papers konden we hier gaan proberen om VTK werkend te krijgen en een aantal *spiked solutions* te schrijven hiervoor. Door een gebrek aan goede documentatie en heldere uitleg heeft dit alles veel meer tijd gekost dan we hadden verwacht.

Uiteindelijk zijn een aantal onderdelen van de Image Processing blijven liggen. Voorbeelden hiervan zijn de transformatie naar de Standard Anatomical View.

7 Aanbevelingen

Hieronder een korte toelichting op werkzaamheden waar wij niet aan toe zijn gekomen. Sommige van deze werkzaamheden zijn niet gebeurd omdat ze buiten de scope van het project vielen of tijdens het project bedacht werden. In enkele gevallen hadden we wel gepland om het werk te doen maar zijn we er niet aan toe gekomen. In de toelichting geven we ook een korte samenvatting van wat er in brainstorm sessies en/of vergaderingen over is afgesproken. Wij hopen op deze manier dat toekomstige groepen op een eenvoudige en snelle manier kunnen verder werken aan het product.

7.1 Bestandstypen van foto's

Op dit moment kan de morph-functie alleen foto's aan die het bestandstype jpeg hebben. Alhoewel dit het meest voorkomende type is zou het een verbetering zijn als het systeem ook met andere typen om kan gaan. Een probleem hierbij is, dat de foto's bij het uploaden wel de extensie 'jpg' krijgen, maar het eigenlijk nog geen jpeg foto's zijn. Het kan nu bijvoorbeeld gebeuren dat er een gif-afbeelding wordt geupload, die vervolgens de extensie jpg krijgt. Hierdoor geeft de vtkJPEGReader een error, en kan er met dit plaatje niet gemorphed worden. Het weergeven van de afbeelding in de browser gaat echter wel goed, waardoor het voor de gebruiker niet duidelijk is dat er eigenlijk iets mis is met de afbeelding.

Een oplossing zou kunnen zijn om bij het uploaden van de afbeelding de Python Imaging Library te laten kijken naar de foto, en deze om te laten zetten naar een echt jpeg bestand. Het gevaar hierbij is echter, dat er door de compressie van het jpeg formaat het een en ander aan scherpte van de foto verloren gaat. Een eventuele oplossing voor dit probleem zou kunnen zijn om helemaal geen jpeg-bestanden meer op te slaan, en alle foto's te laten omzetten naar het PNG-formaat. Bij deze omzetting gaat vrijwel geen informatie in de foto verloren, en voor dit formaat zijn dezelfde VTK-library's beschikbaar als voor het jpeg formaat.

Dit is echter een probleem waar wij binnen het project zodanig laat tegenaan liepen, dat er geen tijd meer was om dit op een fatsoenlijke manier op te lossen.

7.2 Verschillende kleuren in flash applicatie

Binnen onze applicatie is het op dit moment al mogelijk om een bitmap te tekenen in een willekeurige kleur. Wat echter nog niet mogelijk is, is om in één bitmap meerdere kleuren op te slaan. Voor de gebruikers zou het echter intuïtiever zijn om dit wel te kunnen, vooral omdat er bij het tekenen van de bitmap wel met verschillende kleuren gewerkt kan worden. Het moet de gebruiker dan expliciet verteld worden dat de hele bitmap wordt opgeslagen in de kleur waarmee het laatst is gewerkt.

7.3 Opslaan van een lege bitmap

* alle correcties weggooien en geen wijziging opslaan? * lege bitmap opslaan?

7.4 Te grote foto's inladen in de flash-applicatie

7.5 opslaan van statistische data bij landmarks

Een bekende wens vanuit de gebruikers bij het EMC is dat er statistische data opgeslagen kan worden bij de aangegeven landmarks. Deze data heeft dan bijvoorbeeld betrekking op de diepte waarop een vene op verschillende punten in het been ligt. In de huidige situatie wordt deze data apart in een Excel-sheet bijgehouden, en naar een statisticus gebracht. Deze maakt hier vervolgens in aparte software (SPSS) een aantal grafieken van, die de gebruikers eigenlijk bij de andere data op zouden willen slaan. In de toekomst zou het mogelijk moeten zijn om deze data bij de landmarks zelf op te slaan. In een ideale situatie zouden zelfs de grafieken in het systeem moeten kunnen verschijnen, waarna de gebruikers van een willekeurig punt op de grafiek de waarde op kunnen vragen.

Vanwege de complexiteit van het opslaan van de statistische data, het later genereren van de bijbehorende grafieken, en het kunnen opvragen van de waarden op een willekeurig punt, hebben wij hier nog geen stappen voor ondernomen. Een volgende groep zou kunnen beginnen met het enkel toevoegen van de data, en deze overzichtelijk weer te geven, zonder hier grafieken van te maken. Het maken van de grafieken zou bijvoorbeeld gedaan kunnen worden met de PyCha module.

7.6 analyse van deze statische data

7.7 uitgebreid user management en arts-view mode

7.8 morphen naar standaard AnatomicalView

7.9 uitwerken afstand meet applicatie

7.10 zoom op meerdere niveau's

7.11 schaalweergave in image

7.12 export on unix import on windows

8 Conclusie

In de afgelopen tien weken hebben wij een systeem gebouwd dat gebruikt kan worden door het CASAM-project binnen het EMC ten behoeve van hun onderzoek naar de anatomie. In onze ogen voldoet het systeem aan de eisen die de opdrachtgever ons gegeven heeft. In dit project hadden wij op verschillende terreinen grote uitdagingen. Van project management tot het verwerken van de Image processing in onze applicatie. Wij zijn van mening dat de meeste van deze uitdagingen gewonnen zijn en dat zij hebben meegedragen aan een bijzonder product. Zoals eigenlijk bij elk ICT-project is er zeker nog ruimte over voor uitbreiding en verbetering van het huidige product. Deze vervolg werkzaamheden zouden in dezelfde of andere samenstelling kunnen plaatsvinden. Een volgende groep zou zelf moeten kijken in welke richting zij door willen, uiteraard kunnen de punten die genoemd worden in dit verslag als een uitgangspunt worden gebruikt voor de planning van deze werkzaamheden.

9 Acknowledgement

Wij willen graag een aantal mensen en organisaties bedanken voor hun hulp en bijdrage aan het welslagen van dit project: Dr. Botha voor de begeleiding van de kant van de TU Delft en zijn hulp met de visualisatie bibliotheken. Dr. KleinRensink en Dhr. Kerver voor hun begeleiding van de kant van het CASAM project Ing. Slats voor het beschikbaar maken van een zaal en andere faciliteiten aan de Drebbelweg waar wij tien weken konden werken.

10 Bijlagen

10.1 Oriëntatieverslag

10.2 Plan van Aanpak

10.3 Requirements Analysis Document