



Stanley Safari Lodge

Victoria Falls, Zambia

AFSTUDEERVERSLAG

door

M.W. Henneman, 99005151

Ter afronding van de studie
Informatica en Informatiekunde
IVIT aan de

Haagse Hogeschool te 's
Gravenhage

24 maart 2005

Opdrachtgever: R. De Gruijter, Stanley Safari Lodge

Afstudeeropdracht: Sobsys, Stanley's Online Bookingssystem

REFERAAT

Henneman, Mark Wouter

Afstudeerverslag van Mark Henneman ten behoeve van het afronden van de studie Informatica en Informatiekunde in de richting Informatievoorziening en Informatietechnologie (IVIT) aan de Haagse Hogeschool te 's-Gravenhage. De afstudeeropdracht met de titel “Stanley’s Online Boekingssysteem” werd uitgevoerd bij Stanley Safari Lodge te Livingstone, Zambia in de periode 15 november 2004 tot en met 24 maart 2005.

Descriptoren

- Afstudeeropdracht
- Stanley Safari Lodge
- Boekingssysteem
- PHP
- HTML
- MySQL
- Rational Unified Process (RUP)
- Unified Modelling Language (UML)

VOORWOORD

Dit is het afstudeerverslag van Mark Henneman, (nu nog) student Informatica en Informatiekunde, opleidingsvariant Informatievoorziening en Informatietechnologie aan de Haagse Hogeschool te 's-Gravenhage. In dit verslag beschrijf ik de werkzaamheden die ik heb verricht bij het ontwikkelen van SOBSYS, Stanley’s Online Boekingssysteem, een boekingssysteem ter ondersteuning van het boekingsproces bij Stanley Safari Lodge. Tijdens de afstudeeropdracht heb ik gebruik gemaakt van Rational Unified Process (RUP), Unified Modelling Language (UML) en de scriptingtaal PHP.

Graag wil ik van de gelegenheid gebruik maken om een aantal mensen te bedanken die mij zeer geholpen hebben bij het met succes afronden van deze opdracht, maar ook van de studie.

Allereerst wil ik de opdrachtgever en tevens bedrijfsmentor Reinout de Gruijter bedanken voor het beschikbaar stellen van deze buitengewoon geweldige opdracht en de prachtige ervaringen die ik heb mogen opdoen in Zambia. Daarnaast bedank ik hem ook voor de goede begeleiding die ik heb gekregen en uiteraard voor de vele interessante discussies die we hebben gehad over redundante informatie en relationele databases.

Naast de opdrachtgever wil ik ook het overige personeel van Stanley Safari Lodge bedanken voor de prettige tijd die ik in Livingstone heb gehad. Kitty en Betty, van Miles Travel wil ik ook bedanken voor het feit dat zij hebben meegeholpen om het systeem te testen en enkele nuttige tips konden geven ter verbetering ervan.

Daarnaast bedank ik de begeleidend docent vanuit de hogeschool de heer A. Van der Molen en de heer P. Van der Sluijs. Hen wil ik bedanken voor de bruikbare adviezen die ik van hen heb gekregen en onder andere bij het opstellen van het klassediagram en de hulp bij de totstandkoming en goedkeuring van de opdracht.

Ook bedank ik collega-student Eric Oud Ammerveld. Met hem heb ik, vooral tijdens het ontwerpen van het systeem interessante discussies gehad en heeft hij goede feedback gegeven op verschillende tussenproducten. Hij was in dezelfde periode aan het afstuderen, zodat we elkaar van goede adviezen konden voorzien.

Studiebegeleider Wim Looije wil ik ook graag bedanken voor de begeleiding die hij tijdens mijn hele studie heeft gegeven en de interesse die hij heeft getoond.

Als laatste, maar zeker niet de minste wil ik mijn ouders bedanken omdat ze mij tijdens mijn hele studie gesteund hebben en altijd voor me klaar stonden. Mede door hen heb ik deze studie kunnen volgen.

Uiteraard wens ik u ook heel veel plezier bij het lezen van dit verslag!

Livingstone, Zambia, 24 februari 2004

Mark Henneman.

INHOUDSOPGAVE

1. INLEIDING	6
2. STANLEY SAFARI LODGE	8
2.1 Algemeen.....	8
2.2 Boeken.....	8
2.3 Activiteiten	8
2.4 Travelagents	8
2.5 Probleemdomein	9
2.6 Huidige werkwijze boeken.....	9
2.6.1 Excelsheet.....	9
2.6.2 Het mitte bord.....	10
2. TOTSTANDKOMING VAN DE OPDRACHT	11
2.1 Verwerving van de opdracht	11
2.2 Totstandkoming opdrachtsomschrijving.....	11
3. DOORLOPEN INCEPTION PHASE.....	13
3.1 Opstellen plan inception phase	13
3.2 Interview met de heer De Gruijter.....	14
3.3 Opzetten ontwikkelomgeving	14
3.4 Opstellen projectplan.....	15
3.4.1 Kwaliteitsborging.....	15
3.4.2 Te gebruiken methoden en technieken	16
3.4.3 Risicomanagement.....	16
3.4.4 Eisen en beperkingen.....	<i>Fout! Bladwijzer niet gedefinieerd.</i>
3.5 Opstellen visiedocument.....	16
3.5.1 Voordelen en mogelijkheden van het nieuwe systeem.....	16
3.5.2 De problemen die opgelost zullen worden	16
3.5.3 Betrokkenen.....	17
3.5.4 Voornaamste use cases	17
3.6 Opstellen risicodocument	17
3.7 Opstellen use case diagram.....	18
3.8 Opstellen eisendocument.....	19
3.9 Ontwerpen prototype	20
4. DOORLOPEN ELABORATION PHASE.....	21
4.1 Opstellen plan elaboration phase.....	21
4.2 Opstellen Klassediagram.....	22
4.2.1 Identificeer alle mogelijke kandidaatklassen	22
4.2.2 Identificeren van associaties	23
4.2.3 Identificeren van overervingen	25
4.2.4 Aangeven van de multipliciteit en rolnamen.....	26
4.2.5 Identificeren van attributen.....	26
4.2.6 Regels voor attribuutnamen.....	26
4.2.7 Identificeren van operaties	28
4.3 Relationeel Databaseontwerp.....	30
4.4 Implementeren database	32
4.4 Objectgeoriënteerde databaselaag.....	32

4.5 Bijwerken projectdocumenten	34
5. DOORLOPEN CONSTRUCTION PHASE	36
5.1 Iteratieplan Construction phase	36
5.2 Productdocumentatie.....	37
6. ITERATIE I - TRAVELAGENTS.....	37
6.1 Use case inloggen	39
6.2 Use case controleren beschikbaarheid	39
6.3 Use case algemene gegevens boekingen	40
6.4 Use case transfers	41
6.5 Use case gasten	41
7. ITERATIE II - MANAGER.....	42
7.1 Use case inloggen	42
7.2 Use case boekingen.....	42
7.3 Use case boekingsstatus wijzigen.....	43
7.4 Use case boeking zoeken	43
7.5 Use case defragmenteren	46
8. ITERATIE III - OVERZICHTEN.....	44
8.1 Use case Overzicht beschikbaarheid en status.....	44
8.2 Use case Overzicht ‘White board’	45
9. TRANSITION PHASE.....	46
9.1 Testen.....	48
9.2 Debuggen en nieuwe eisen implementeren.....	48
9.3 Invoeren nieuwe systeem.....	48
9.4 Onderhoudsplan.....	49
10. PROJECTEVALUATIE	50
10.1 Proces.....	50
10.2 Product	50
LITERATUURLIJST	51
BIJLAGE A – USE CASE TEMPLATES	52
BIJLAGE B – SEQUENTIEDIAGRAMMEN.....	53
BIJLAGE C – VOORLOPIGE PLANNING.....	54

EXTERNE BIJLAGEN

1. INLEIDING

Dit is het afstudeerverslag van Mark Henneman. De opdracht behelst het ontwerpen, bouwen en implementeren van een webbased boekingssysteem ten behoeve van Stanley Safari Lodge in Livingstone, Zambia.

Het verkrijgen van de opdracht en de totstandkoming van de opdrachtschrijving leest u in het eerste deel. Een omschrijving van het bedrijf vindt u eveneens in dit deel.

Het volgende deel, deel II, gaat over de werkzaamheden die ik voor Stanley Safari Lodge heb uitgevoerd. Dit deel is verdeeld in een aantal verschillende hoofdstukken, welke globaal overeenkomen met de iteraties in het project. De hoofdstukindeling is chronologisch gestructureerd, waardoor de RUP-fasen duidelijk te herkennen zijn. De RUP-fasen zijn de inception phase, de elaboration phase, de construction phase en de transition phase.

Na de beschrijving van de werkzaamheden vindt u een apart hoofdstuk voor projectevaluatie.

DEEL I

TOELICHTING OP DE OPDRACHT

2. STANLEY SAFARI LODGE

In dit hoofdstuk wordt een beschrijving geven van het gastbedrijf: Stanley Safari Lodge. De bedrijfsanalyse is pas in week 6 van de afstudeerperiode uitgevoerd, omdat ik daarvoor nog niet op locatie was. In de eerste week heb ik wel een aantal punten beschreven, maar pas later kreeg ik pas een goed beeld van het bedrijf.

2.1 ALGEMEEN

Stanley Safari Lodge is een exclusieve kleinschalige lodge met slechts 7 kamers/suites, enkele kilometers van de Victoria Falls in Zambia. De gasten komen hoofdzakelijk uit Europa en de Verenigde Staten.

Het merendeel van de gasten verblijft enkele dagen aldaar en combineert de Victoria Falls met een safari in het Botswaanse Okavango Delta, het Zambiaanse South Luangwa National Park of een bezoek aan Zuid-Afrika. De gasten hoeven het verblijf zelf niet te regelen. Dat regelen de reisagentschappen en touroperators via veelal e-mail, de telefoon en per fax met de manager van de lodge.

2.2 BOEKINGEN

Boeking worden verzorgd door een reserveringspersoon welke niet op de lodge verblijft. De eigenaar verzorgt in sommige gevallen ook enkele reserveringen. Vanuit een reserveringsoogpunt is andere informatie (of op zijn minst een andere presentatie) gewenst dan vanuit een operationeel manager oogpunt. Beschikbaarheid van kamers is hierbij één van de meest belangrijke overzichten aangezien soms telefonisch beschikbaarheid wordt opgevraagd.

2.3 ACTIVITEITEN

De lodge boekt ook activiteiten die de gasten willen ondernemen tijdens hun verblijf in Livingstone. De manager regelt dat via de radio door degene die de activiteit aanbiedt te contacteren. Per boeking krijgt de lodge een commissie variërend van 10 tot 30 procent. Een enkele greep uit de activiteiten die op locatie worden aangeboden zijn: een helikoptervlucht over de Falls en het National Park, rafting, quadbiking, ‘elephant-back safari’ en een bootsafari over de Zambezi River.

Reisbureaus kunnen bij het boeken van een kamer al zogenaamde ‘pre-booked’ activiteiten aangeven als de cliënten bij voorbaat al weten welke activiteiten ze willen gaan ondernemen.

2.4 TRAVELAGENTS

Travelagents zijn de reisbureaus die voor hun eigen cliënten (dus de toekomstige gasten van Stanley) een kamer boeken bij de lodge. Reisbureaus kunnen zich over de gehele wereld bevinden, over het algemeen bevinden ze zich in Zuid-Afrika, Europa en Noord-Amerika. Reisbureaus krijgen per boeking die zij maken een commissie variërend van 15 tot 25 procent. Zo krijgt bijvoorbeeld MilesTravel in Eindhoven, waar ik in december op bezoek ben geweest, 20 procent van het totale boekingsbedrag.

2.5 PROBLEEMDOMEIN

Het probleemgebied rondom de boekingen van de kamers. De travelagents doen hun boekingen vooral via e-mail. Omdat nooit alle benodigde gegevens (paspoortgegevens, data, transferdata etc.) meteen door de reisagenten worden aangeleverd kan het aantal e-mailberichten wel eens oplopen tot meer dan vijftien, en dat per boeking. Naast dit probleem wilt de manager, als hij in Europa of elders in de wereld is, direct inzicht kunnen hebben in de boekingen en de beschikbaarheid van de kamers. Op de huidige manier van boeken gaat dit niet. Hier zal ik in de volgende paragraaf meer over vertellen.

2.6 HUIDIGE WERKWIJZE BOEKINGEN

2.6.1 Excelsheet

Als reisbureaus een boeking willen doen wordt er een in eerste instantie een e-mailbericht naar de manager of naar de afdeling ‘Reserveringen’ gestuurd met de vraag of er tijdens de aangegeven periode (bijvoorbeeld 20 november tot en met 23 november) een kamer vrij is en zo ja welke kamer dat is. De manager hanteert voor het controleren van de beschikbaarheid een zelfgemaakt excelsheet.

No	PAX	NAMES	TRANSFER IN	TRANSFER OUT
1	2	Mr. John Lawrence and Mrs. S. W. 1265	1	1
2	2	Mr. John Lawrence and Mrs. S. W. 1265	1	1
3	2	Mr. John Lawrence and Mrs. S. W. 1265	1	1
4	2	Mr. John Lawrence and Mrs. S. W. 1265	1	1
5	2	Mr. John Lawrence and Mrs. S. W. 1265	1	1
6	2	Mr. John Lawrence and Mrs. S. W. 1265	1	1
7	2	Mr. John Lawrence and Mrs. S. W. 1265	1	1
8	2	Mr. John Lawrence and Mrs. S. W. 1265	1	1
9	2	Mr. John Lawrence and Mrs. S. W. 1265	1	1
10	2	Mr. John Lawrence and Mrs. S. W. 1265	1	1
11	2	Mr. John Lawrence and Mrs. S. W. 1265	1	1
12	2	Mr. John Lawrence and Mrs. S. W. 1265	1	1
13	2	Mr. John Lawrence and Mrs. S. W. 1265	1	1
14	2	Mr. John Lawrence and Mrs. S. W. 1265	1	1
15	2	Mr. John Lawrence and Mrs. S. W. 1265	1	1
16	2	Mr. John Lawrence and Mrs. S. W. 1265	1	1
17	2	Mr. John Lawrence and Mrs. S. W. 1265	1	1
18	2	Mr. John Lawrence and Mrs. S. W. 1265	1	1
19	2	Mr. John Lawrence and Mrs. S. W. 1265	1	1
20	2	Mr. John Lawrence and Mrs. S. W. 1265	1	1
21	2	Mr. John Lawrence and Mrs. S. W. 1265	1	1
22	2	Mr. John Lawrence and Mrs. S. W. 1265	1	1
23	2	Mr. John Lawrence and Mrs. S. W. 1265	1	1
24	2	Mr. John Lawrence and Mrs. S. W. 1265	1	1
25	2	Mr. John Lawrence and Mrs. S. W. 1265	1	1
26	2	Mr. John Lawrence and Mrs. S. W. 1265	1	1
27	2	Mr. John Lawrence and Mrs. S. W. 1265	1	1
28	2	Mr. John Lawrence and Mrs. S. W. 1265	1	1
29	2	Mr. John Lawrence and Mrs. S. W. 1265	1	1
30	2	Mr. John Lawrence and Mrs. S. W. 1265	1	1
31	2	Mr. John Lawrence and Mrs. S. W. 1265	1	1

Het moge duidelijk zijn dat dit excelsheet niet de meest gebruiksvriendelijke userinterface heeft, waardoor er een grote kans bestaat (en dat is al vaak voorgekomen) dat er een fout wordt gemaakt bij het controleren of invoeren van boekingen.

2.6.2 Het witte bord

Voor het operationele vlak gebruikt de manager een groot wit bord waar per kamer en per datum de boekinggegevens worden weergegeven. Hieronder heb ik een foto gemaakt om u een betere indruk te geven hoe het bord eruitziet. Het witte bord is voor elk personeelslid beschikbaar. Per kamer wordt dan weergegeven wie in de kamer verblijft en welke extra wensen en opmerkingen daarbij horen.

	MON 4	TUE 5	WED 6	THU 7	FRI 8	SAT 19	SUN 20
1							
2							
3							
4							
5							
6							
7							
8							
9							
10							

Bijvoorbeeld Mevr. Jansen uit kamer 1 is vegetarisch. De keuken weet dan dat er éénmaal minder vlees voorbereid moet worden. Of familie Pietersen heeft al een zogenaamde ‘pre-booked’ activiteit rafting aangegeven, toen zij bij het

reisbureau een kamer bij de lodge boekte. Rafting zal dan op het bord worden geschreven met daarbij de datum van de activiteit.

2. TOTSTANDKOMING VAN DE OPDACHT

Dit hoofdstuk zal een beschrijving geven hoe ik aan de afstudeeropdracht gekomen ben en hoe ik in contact gekomen ben met de heer De Gruijter van Stanley Safari Lodge en hoe de opdrachtschrijving tot stand is gekomen.

2.1 VERWERVING VAN DE OPDRACHT

Na een aantal sollicitatiebrieven te hebben gestuurd naar bedrijven in Nederland heb ik in totaal een drietal gesprekken gehad met potentiële gastbedrijven. Twee daarvan konden mij geen geschikte afstudeeropdracht bieden, terwijl ik bij de ander afgewezen werd. Mijn stage heb ik ook in het buitenland heb gedaan (op Curaçao, Nederlandse Antillen) welke voor mij een goede ervaring was. Omdat het buitenland mij trekt heb ik besloten eens op het Internet te zoeken naar afstudeeropdrachten in het buitenland. Twee opdrachten spraken mij erg aan, het ontwikkelen van een ordersysteem in Zuid-Spanje en deze opdracht die ik uiteindelijk gekozen heb, het ontwikkelen van een boekingssysteem. Het eerste bedrijf dat mij een stageplek aanbood was het bedrijf in Zuid-Spanje. Een aantal dagen erna belde de heer De Gruijter uit Zambia op om mij dáár een stageplek aan te bieden. De keuze was voor mij snel gemaakt. De opdracht in Zambia bood voor mij veel meer uitdaging en de heer De Gruijter kwam erg overtuigend op mij over.

2.2 TOTSTANDKOMING OPDRACHTSOMSCHRIJVING

Tijdens het goedkeuringstraject voor de afstudeeropdracht heb ik in totaal zes gesprekken gehad met drie verschillende personen. De afstudeercoördinator, te weten de heer A. Schipper en twee begeleidend docenten, te weten de heer A. Van der Molen en de heer P. Van der Sluijs. Mede door deze gesprekken is de opdrachtschrijving tot stand gekomen. In totaal heb ik drie gesprekken met de heer A. Van der Molen gehad.

In eerste instantie was het de bedoeling dat mijn verblijf in Zambia drie maanden zou duren. Omdat er minimaal twee gesprekken nodig waren tussen de bedrijfsmentor, de student en de begeleidend docent heb ik besloten om het verblijf in Zambia met een maand in te korten.

Er kleven een aantal nadelen aan het afstuderen in het buitenland. Deze nadelen waren voor mij erg duidelijk; er was minder contact met de begeleidend docenten en je verliest sneller het contact met school. Echter, ik heb de afstudeercoördinator, mede middels een motivatiebrief, weten te overtuigen dat de opdracht een grote kans van slagen had.

De opdrachtschrijving kunt u vinden in de externe bijlage hoofdstuk 2.

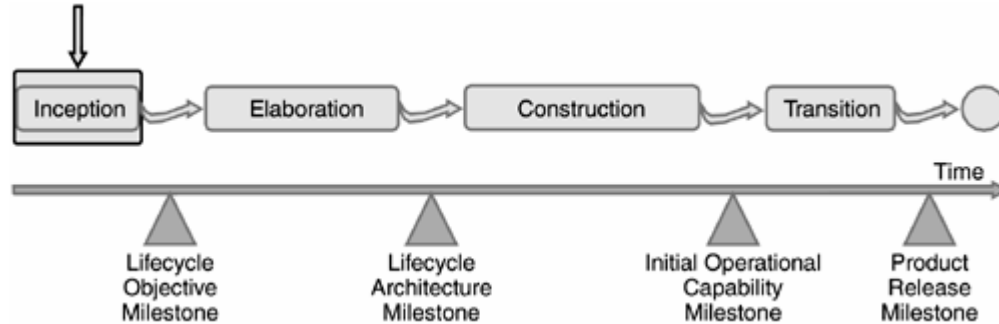
DEEL II

BESCHRIJVING INHOUDELIJKE ACTIVITEITEN

In dit deel van het afstudeerverslag worden de werkzaamheden die ik tijdens mijn afstudeerproject gedaan heb beschreven. Hier zal per fase uit de Rational Unified Proces beschreven worden welke activiteiten ik heb uitgevoerd. De vier fasen van RUP zijn de inception phase, elaboration phase, construction phase en de transition phase.

3. DOORLOPEN INCEPTION PHASE

Dit hoofdstuk gaat over de Inception Phase binnen de Rational Unified Proces. Onderstaand figuur geeft aan ‘waar’ ik me tijdens het project bevond. Zoals de pijl aangeeft zit ik hier dus in de Inception Phase.



Het hoofddoel van de inception phase is het bereiken van overeenstemming tussen belanghebbenden over de levenscyclusdoelstellingen van het project. De primaire doelstellingen van de inception phase zijn:

- Het bepalen van de reikwijdte van het project en de grensvoorwaarden. Dit zorgt voor een duidelijke afbakening van het project.
- Het onderscheiden van de kritische use cases van het systeem geeft antwoord op de vraag wat de belangrijkste functionaliteit van het systeem is.
- Het demonstreren van tenminste één kandidaatarchitectuur, om in een duidelijk schematisch overzicht weer te geven hoe het systeem er uit gaat zien.
- Het schatten van de kosten en het tijdschema van het hele project (inclusief gedetailleerde berekeningen voor de elaboration phase)
- Het inschatten van potentiële risico's (bronnen van onvoorspelbaarheid) om de gevolgen hiervan zo veel mogelijk in te perken.

3.1 OPSTELLEN PLAN INCEPTION PHASE

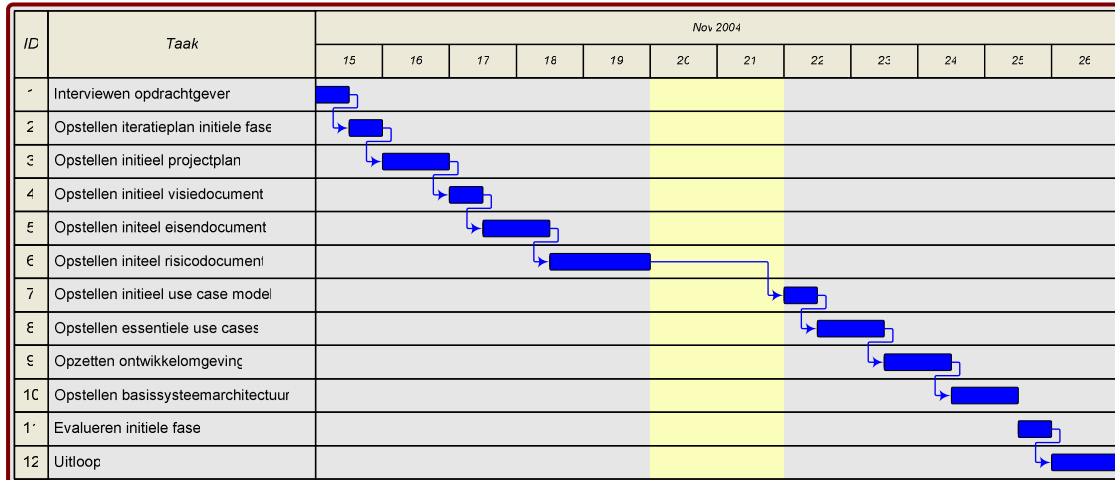
In deze fase worden de volgende artefacten opgeleverd:

- Projectplan
- Eisendocument
- Risicodocument
- Visiedocument
- Use case model
- Opgezette ontwikkelomgeving
- Basissysteemarchitectuur

Alle activiteiten die in de opdrachtschrijving genoemd worden gedekt in het activiteitenplan dan RUP voorschrijft. Alleen het interviewen van de opdrachtgever wordt niet gedekt. Ik beschouw dit als een activiteit die op elk willekeurig moment kan plaatsvinden. De opdrachtgever en ik werkten in hetzelfde kantoor dus ik kon ten alle tijden vragen stellen aan de opdrachtgever. Het opstellen van het plan van aanpak en het formuleren van de reikwijdte van het project vallen onder respectievelijk het initiele projectplan en het visiedocument. Het opstellen van de globale eisen valt onder het

opstellen van een eisendocument en het opstellen van de essentiële use cases spreekt voor zich. Het samenstellen van de architectuur valt onder het opstellen van de basissysteemarchitectuur.

De planning van de inception phase op basis van de bovenstaande activiteiten ziet er als volgt uit:



De planning geeft aan dat ik alle artefacten seriëel heb opgeleverd. Dit is echter niet het geval. Ik heb me bijvoorbeeld niet één dag lang bezig gehouden met het eisendocument. Steeds als er een eis boven tafel kwam, welke voortkwam uit de verschillende gesprekken met de opdrachtgever, noteerde ik deze in het eisendocument. Ditzelfde geldt voor het risicodocument. De artefacten worden dus parallel aan elkaar gemaakt.

Verder in dit hoofdstuk wordt per paragraaf elke activiteit besproken die in deze fase uitgevoerd is.

3.2 DISCUSSIE MET DE HEER DE GRUIJTER

De eerste stagedag hebben de heer De Gruijter en ik bijna helemaal besteed aan discussiëren over het systeem. Dit informeel interview te noemen. We hebben de wensen en eisen van het te bouwen systeem besproken en ik heb daarvan de belangrijkste punten opgeschreven. Op basis van dit interview zijn de documenten als het visiedocument, het eisendocument voortgekomen.

3.3 OPZETTEN ONTWIKKELOMGEVING

Door het opzetten en testen van de ontwikkelomgeving wordt het risico dat er tijdens de constructie phase iets fout gaat met de software, zodat bijvoorbeeld het project vertraging op kan lopen verkleind.

De opdrachtgever heeft voor het ontwikkelen van het boekingssysteem een laptop ter beschikking gesteld. Op deze laptop stond aanvankelijk Windows 2000 geïnstalleerd. Omdat de heer De Gruijter deze laptop al zo'n twee jaar in gebruik had, zonder hem opnieuw te installeren heb ik besloten om de laptop geheel opnieuw in te richten. Er is

gekozen voor een laptop omdat er op verschillende locaties aan het project gewerkt gaat worden. Ook omdat in Zambia regelmatig (1x per week) de stroom uitvalt kun je door middel van een laptop toch nog een tijd doorwerken. Op deze manier kan het werk gemakkelijk meegenomen worden.

Server: database(management), PHP-parser: Appserv 2.4.1

Appserv is een open project dat PHP, Apache, MySQL, Zend Optimizer, phpMyAdmin en Perl als één pakket aanbiedt. Ik heb voor versie 2.4.1 gekozen, omdat daar de meest gangbare (lagere) versies van bovenstaande tools worden gecombineerd. Het risico dat een hostingprovider hogere versies van de tools niet ondersteunt wordt hierdoor verkleind.

De volgende tools worden door Appserv gecombineerd:

- phpMyAdmin databasemanager version 2.6.0-rc1
- PHP Information version 4.3.8
- Perl Version 5.8.4 (niet van belang bij dit project)
- Apache Webserver version 1.3.31
- PHP Script Language version 4.3.8
- MySQL database version 4.0.20
- Zend Optimizer version 2.5.3 (optimaliseert PHP-code)

Implementatie: Macromedia Dreamweaver MX, Homesite 5.0, Topstyle

Dreamweaver wordt gebruikt voor het (initieel) ontwerpen van schermen en lijsten in HTML. Homesite wordt gebruikt voor het bewerken van maken en bewerken van zowel HTML-pagina's als PHP-scripts. Voor het maken van Cascading StyleSheets (CSS) wordt gebruik gemaakt van TopStyle. Een programma dat bij het Macromedia-pakket wordt meegeleverd.

3.4 OPSTELLEN PROJECTPLAN

Aan de hand van de opdrachtschrijving heb ik een eerste versie van het projectplan gemaakt. Alle activiteiten die in de planning in de opdrachtschrijving staan heb ik één op één gekopieerd. Naar mijn idee had ik al een redelijke schatting gemaakt van de planning. Voor de planning heb ik, zoals in de vorige paragraaf staat beschreven, gebruik gemaakt van MS Visio. In de interne bijlage C vindt u de voorlopige planning van het project:

3.4.1 Kwaliteitsborging

Voor het beheersen van de kwaliteit van het project heb ik gebruik gemaakt van de SMART. SMART is een belangrijk en berucht hulpmiddel voor het beheersen van de kwaliteit van projecten. SMART staat voor:

- Specifiek; doelstelling nastreven
- Meetbaar; is/blijft de opdrachtgever tevreden?
- Aanwijsbaar; Wat hebben we gemaakt?
- Realistisch; Is/blijft het haalbaar?
- Tijdgerelateerd; planning bewaken

Naast SMART had ik ook nog MAGIE kunnen gebruiken. Dit komt ongeveer op hetzelfde neer en betekent: Meetbaar, Acceptabel, Gedragen, Inspirerend en Engageren. Echter SMART heb ik al een paar keer gebruikt en heb ik logischerwijs dit keer gebruikt.

3.4.2 Te gebruiken methoden en technieken

Ik heb gekozen om tijdens het project gebruik te maken van UML voor het ontwerpen van onder andere het klassediagram en de datamodelgereedschappen die MS Visio ter beschikking stelt. Een andere mogelijkheid was om ERD-schematechnieken te gebruiken maar omdat ik daar geen behoorlijk software voor kon vinden heb ik het bij MS Visio gehouden. Verder maak ik gebruik van interviewtechnieken voor het achterhalen van belangrijke informatie. Daarnaast gebruik ik voor het plannen van het gehele project en de verschillende fases daarin een GANTT-diagram.

3.4.3 Risicomanagement

Risicomanagement behoort tot een van de belangrijkste elementen bij het systeemontwikkelingstraject. RUP biedt uitstekende mogelijkheden voor het beheersen van risico's. Namelijk door het maken van een risicodocument.

Voor het volledige projectplan verwijs ik u naar externe bijlagen hoofdstuk 4.

3.5 OPSTELLEN VISIEDOCUMENT

De eerste belangrijke artefact die de Rational Unified Process voorschrijft is het visiedocument. Aan de hand van het interview met de heer de Gruijter en de opdrachtomschrijving heb ik de volgende punten in het visiedocument besproken

- Voordelen en mogelijkheden
- De problemen die opgelost zullen worden
- De betrokkenen
- De voornaamste use cases
- Voornaamste niet-functionele eisen

In totaal bestond mijn visiedocument uit twee pagina's. Dat is niet veel maar omvatte de belangrijkste elementen voor het project.

3.5.1 Voordelen en mogelijkheden van het nieuwe systeem

De belangrijkste twee daarvan zijn:

- | |
|---|
| <ul style="list-style-type: none">▪ Er kan “any place, any time” toegang verkregen tot het boekingssysteem▪ Het boekingsproces verloopt sneller en efficiënter |
|---|

Bovenstaande twee punten zullen al een ruime dekking geven aan de probleemstelling die in de opdrachtomschrijving gegeven.

3.5.2 De problemen die opgelost zullen worden

Hierin wordt beschreven welke problemen door het nieuwe boekingssysteem opgelost zullen worden. In grote lijnen komt het op hetzelfde neer als wat al in de opdrachtomschrijving staat bij de probleemstelling. Echter, hier worden de problemen meer gedetailleerd beschreven, zodat een beter beeld verkregen worden van de problemen die zich voor doen bij Stanley Safari Lodge.

De problemen die besproken worden zijn:

- Reisagenten kunnen niet direct de beschikbaarheid van de kamers controleren.
- Er wordt inefficiënt via e-mail gewerkt.
- De opdrachtgever wil, ook als hij niet op de lodge verblijft, toegang hebben tot het boekingssysteem.

3.5.3 Betrokkenen

Hierin worden alle betrokken van het project opgesomd. Van toekomstige gebruikers tot de projectleden.

De reisagent MilesTravel is betrokken omdat twee medewerkers hebben meegedacht over het nieuwe systeem. In een later stadium (in de transition phase, zie hoofdstuk 9) zullen zij ook het nieuwe systeem gaan testen. Ik heb aan de opdrachtgever hun contactgegevens gevraagd omdat ik graag hun mening en visie over het systeem graag wilde hebben.

3.5.4 Voornaamste use cases

De voornaamste use cases zullen het grootste deel van de doelstelling dekken. Deze heb ik bepaald aan de hand van het gesprek met de heer de Gruijter en met de reeds gemaakte opdrachtoomschrijving. Met name de use case boekingen is erg belangrijk, omdat dit de basis is voor het boekingssysteem. Deze use case kan ook weer onderverdeeld worden in kleinere use cases. Ook inloggen is een erg belangrijke use case, waar veel aandacht aan besteed moet worden. Het is immers uiterst belangrijk dat de gegevens beveiligd zijn, zodat onbevoegden niet bij de gastgegevens en de boekingsgegevens kunnen komen. Er wordt minder nadruk gelegd op de helpfunctie. De meeste functies die in het boekingssysteem zullen voorkomen zijn voor de hand liggend. De volgende essentiële use cases zijn in het visiedocument opgenomen:

- Boekingen
- Inloggen
- Wijzigen boekingsstatus
- Wijzigen kamerverdeling (defragmentatie)
- Overzichten

3.6 OPSTELLEN RISICODOCUMENT

De gevolgen van risico's zo minmaal houden is erg belangrijk tijdens een softwareontwikkeltraject. Risico's zijn die dingen, welke een project kunnen doen falen. Ik wil in een zo kort mogelijke tijd waardige software bouwen, maar als ik te snel ga kan ik een systeem opleveren met te veel fouten. Als ik te langzaam ga kan een systeem opleveren die niet eens af is. Als ik nieuwe technologieën ga gebruiken heb ik meer risico op het maken van fouten. Als ik juist geen nieuwe technologieën ga gebruiken loop ik het risico achter te blijven op de concurrentie. Het laatste voorbeeld zal waarschijnlijk geen probleem zijn aangezien ik het boekingssysteem webbased zal maken. Ook gebruik ik voor mezelf geen nieuwe technologieën. Ik ben bekend met HTML en Javascript en heb PHP en MySQL ook al redelijk onder de knie.

Een ander risico dat zich wel voordoet is dat ik nog niet echt bekend ben RUP. Om het risico zoveel mogelijk in te perken heb ik een aantal boeken over RUP aangeschaft en geleend uit een bibliotheek om zoveel mogelijk kennis erover te vergaren als wel als naslag te gebruiken.

Net als de meeste documenten in RUP wordt ook het risicodocument gedurende het hele project aangepast. Het kan niet zo zijn dat je een risicodocument maakt en hem vervolgens ergens opslaan en hem nooit meer inziet. Vandaar dat ik in deze fase ook spreek over een

initieel risicodocument. Er kan ten alle tijden nog risico’s bijkomen. Hoewel bij mij de meeste risico’s al in de inception phase naar bovenkwamen, zijn er ook in de overige fasen een aantal bijgekomen.

De volgende punten zijn per eis in het risicodocument opgenomen:

- Omschrijving
- Prioriteit; hoe hoog is het risico?
- Datum van invoer
- Wat moet ik doen om het risico te beheersen?

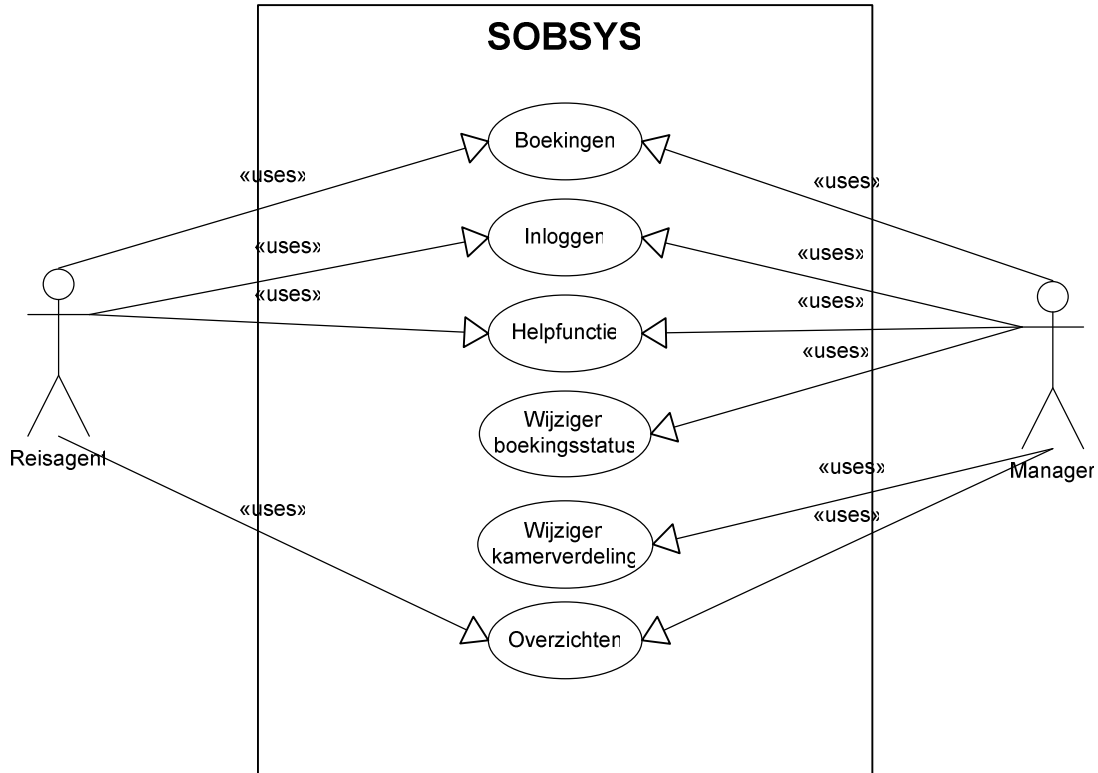
Hieronder staat een fragment uit het risicodocument:

Geen goede architectuur		Hoog	23/11
	Begin niet te snel aan de constructiefase, als je nog niet helemaal zeker bent van de architectuur.		
	Begin pas aan de constructiefase als je zeker bent van de architectuur.		
RUP is geen doel, maar een hulpmiddel		Gemiddeld	23/11
	Als je teveel bezig bent met het goed uitvoeren van RUP		
	Vergeet niet dat je een systeem moet ontwikkelen.		

3.7 OPSTELLEN USE CASE DIAGRAM

Samen met de opdrachtgever heb ik het visiedocument besproken stap voor stap langsgelopen. We waren het er over eens dat we beide met de neus de goede richting in stonden. Hierna ben ik begonnen met het use case diagram.

Aan de hand van onder andere de opdrachtomschrijving, het projectplan en het visiedocument ben ik tot een lijst gekomen met de belangrijkste use cases, de voornaamste use cases. Deze use cases heb ik vervolgens overzichtelijk in een model geplaatst die er als volgt uit ziet.



Zoals u kunt zien zijn er twee actoren. De manager en de reisagent. De twee actoren communiceren met het systeem SOBSYS, aangegeven door het grote vierkant. De teksten binnen de ovale vormen geven de namen aan van de use cases.

3.8 OPSTELLEN EISENDOCUMENT

Ter aanvulling van het de essentiële use cases en het use case model heb ik een eisendocument gemaakt. Hierin staan eisen die niet opgenomen zijn in het use case model of in de use cases. Het is dus een aanvulling op deze. De eisen die worden geformuleerd in het eisendocument bevatten de volgende gegevens:

- Omschrijving van de eis
- Prioriteit (hoog, medium, laag); is het belangrijk voor de opdrachtgever?
- Status (open, gestart, afgerond)
- Datum van inventarisatie
- Moeilijkheidsfactor (1-5) op basis van mijn programmeerervaring

Een fragment van het eisendocument ziet er als volgt uit:

#	Omschrijving	Prior.	Mo.	Status	Datum	Pre
1	De tijdslimiet voor boekingen te wijzigen < 2 wkn					
2	Er moet een helpfunctie beschikbaar zijn.					
3	Reisagenten moeten online een beschikbaarheid van de kamers controleren.					
4	Reisagenten moeten een boekingsoverzicht kunnen laten genereren.					
5	Bij een boeking moet er een hoofdboeker aangegeven kunnen worden. (geschrapt)					

Een volledige eisendocument kunt u vinden in hoofdstuk 5 van de externe bijlagen.

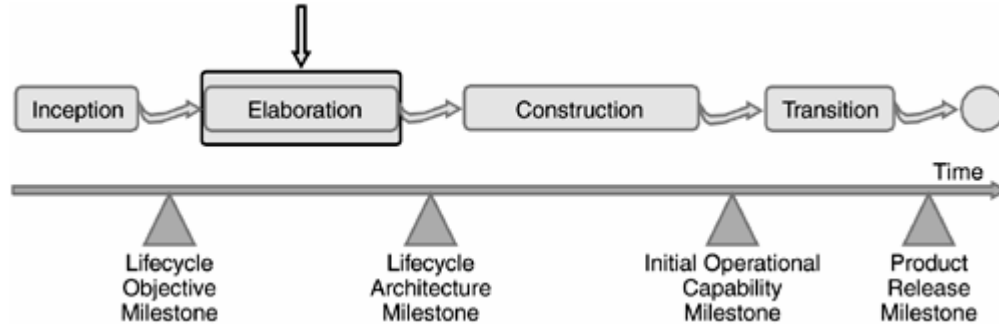
3.9 ONTWERPEN PROTOTYPE

In de inception phase had ik gelijk een interview met een tweetal toekomstige gebruikers van het systeem. Twee medewerkers van het reisbureau MilesTravel heb ik kunnen interviewen en wat zij verwachtten van een boekingssysteem. Het is belangrijk dat gebruikers het nieuwe systeem zich een beetje visueel kunnen voorstellen, in ieder geval de functies en mogelijkheden van het systeem zullen zijn. Daarom heb ik voorafgaand aan het interview een prototype gemaakt. Dit prototype geeft summier een userinterface en geeft in elk scherm aan wat de toekomstige (potentiële) functies er onder andere zullen worden geïmplementeerd. Onderstaand is een screenshot van het prototype welke ik heb laten zien tijdens het interview, waardoor de medewerkers van het reisbureau enig idee konden krijgen over het boekingssysteem en zodat ze enkele nieuwe ideeën op konden doen, zodat ik deze in een later stadium kon implementeren.

General	Rooms	Guests	Activities	Confirmation	Invoice
displays: - bookingsnumber - name of the booking - begindate - enddate features: - check availability					

4. DOORLOPEN ELABORATION PHASE

Dit hoofdstuk gaat over de fase Elaboration phase binnen de Rational Unified Proces. Onderstaand figuur geeft aan ‘waar’ ik me tijdens het project bevond. Zoals de pijl aangeeft zit ik hier in de Elaboration Phase.



Het algemene doel van de elaboration phase is dus de ‘Lifecycle Architecture Milestone’. De mijlpaal van de architectuur.

Doelstellingen elaboration phase

De doelstellingen van elaboration phase zijn:

- Een betere beschrijving/begrip van de eisen
- Ontwerp, implementeren en valideren van de basisarchitectuur
- Beheersen van de risico's en het maken van een gedetailleerdere planning
- Verfijnen van de use case-beschrijvingen

4.1 OPSTELLEN PLAN ELABORATION PHASE

Het iteratieplan heb ik gemaakt op basis van mijn ervaring met het maken van een klassediagram en een relationeel databaseontwerp.

De elaboration phase zou volgens het projectplan van 29 november 2004 tot 15 december 2004 duren. De fase startte daadwerkelijk ook op de 29e van november, omdat de inception phase geen uitloop had. Voor de elaboration phase had ik besloten om de volgende artefacten op te leveren:

- klassediagram
- relationeel databaseontwerp
- aangepast use case model
- aangepaste use cases en nieuwe use cases templates
- bijgewerkte planning

De planning van de elaboration phase op basis van de bovenstaande artefacten ziet er als volgt uit:

IC	Task Name	Duration	Nov 2004		Dec 2004															
			29	30	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	Opstellen klassediagram	2c																		
2	Opstellen overige use cases	2c																		
3	Aanpassen use cases	2c																		
4	Opstellen relationeel databaseontwerp	1c																		
5	Maken prototype van het systeem	5c																		
6	Planning bijwerker	1c																		

Verder in dit hoofdstuk wordt per paragraaf elke activiteit besproken die in deze fase uitgevoerd is. Allereerst in de volgende paragraaf een korte toelichting over UML waarin ik aangeef welke diagrammen ik ga gebruiken en welke niet.

4.2 OPSTELLEN KLASSEDIAGRAM

Het klassediagram toont de statische structuur van het softwaresysteem weergegeven als klassen en hun relaties. Dit is de basis voor het te ontwikkelen systeem. Erg belangrijk dus. In deze paragraaf vertel ik stap voor stap uit hoe het klassediagram tot stand is gekomen. Het systeem zal in de toekomst worden uitgebreid met meerdere modules. Er zal in de toekomst onder andere een activiteitsmodule om gasten in te delen in verschillende activiteiten (rafting, gamedrive et cetera). Daarnaast zal er ook nog een facturenmodule komen voor de facturering aan de gasten, welke als optionele use case is meegenomen tijdens dit project. Met deze toekomstige uitbreidingen diende tijdens de ontwikkeling van met name het klassediagram en het databaseontwerp dus rekening gehouden te worden. Immers deze twee modellen vormen het fundament van een systeem.

4.2.1 Identificeer alle mogelijke kandidaatklassen

Het doel van deze stap is om zonder beperkingen zoveel mogelijk kandidaatklassen te vinden. Pas in een volgende stap heb ik hieruit een selectie gemaakt. Het belang van deze eerste stap is dat we ons blikveld zoveel mogelijk verruimen. We laten onszelf maar al te vaak van tevoren inperken om allerlei redenen die niet expliciet gemaakt worden. Bij het achterhalen van mogelijke objecten heb ik gebruik gemaakt van twee manieren: *het onderstrepen van de zelfstandige naamwoorden in de reeds gemaakte use-cases* en *het houden van een brainstormsessie*.

Zoals in het vorige hoofdstuk al te lezen viel heb ik in de inception phase al een aantal use-cases gemaakt. Na het onderstrepen van de relevante zelfstandige naamwoorden in de reeds gemaakt use-cases heb ik daarvan een lijst gemaakt.

Vervolgens, na het opstellen van de initiële lijst van mogelijke klassen ben ik gaan brainstormen over welke klassen er nog meer van belang zouden kunnen zijn bij het boekingsysteem. Zodoende werd de lijst aangevuld met een aantal potentiële klassen. Vervolgens, na de brainstormsessie, zijn ben ik bij elke kandidaatklasse stil gaan staan en heb ik bekeken of deze wel of niet opgenomen gaat worden in de eerste versie van het klassendiagram. Met andere woorden of de kandidaatklasse ook daadwerkelijk een klasse is. Bovenstaand geheel heeft geresulteerd in een *kandidaatklasselijst*.

In bijlage D ziet u dit resultaat.

4.2.2 Identificeren van associaties

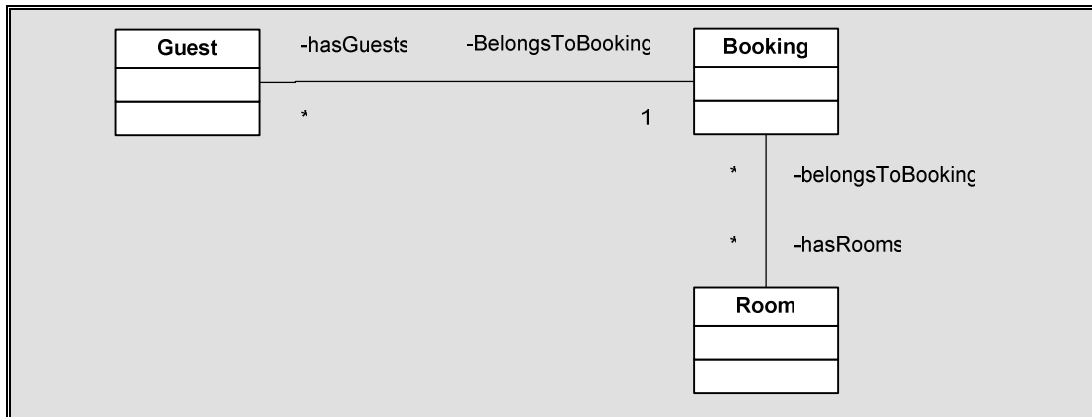
Deze stap is vergelijkbaar met het identificeren van de kandidaatklassen. In principe is er een mogelijke associatie tussen iedere twee klassen. In deze stap zoeken we echter naar paren van klassen, die iets met elkaar te maken hebben in de context van het boekingssysteem. Eerst heb ik alle mogelijke associaties tussen de klassen opgesomd. Ik heb per paar bekeken of er een relevante relatie bestaat tussen de twee. Vervolgens heb ik deze aan het klassediagram toegevoegd, inclusief de rolnamen en multipliciteit.

Gasten niet aan kamer koppelen

In eerste instantie had ik de klasse Contact_Guest aan de klasse Roombooking gerelateerd, omdat ik dacht dat het wel handig zou kunnen zijn om te weten welke gast in welke kamer zat. Zoals dit in onderstaand figuur is weergegeven.



Na overleg met de opdrachtgever heb ik deze associatie laten varen. De reden hiervoor was dat het voor de opdrachtgever irrelevant was om te weten wie er in welke kamer slaapt. Bovendien kost het de reisagent een extra handeling om een gast aan een specifieke kamer te koppelen en mochten de gasten van één bepaalde boeking de behoefte hebben om van kamer te verwisselen wordt deze functie waardeloos. Kortom het enige wat deze associatie op zou leveren was administratieve extra werk. Daarom heb ik het op de volgende manier geïmplementeerd:



Gasten niet aan activiteit koppelen

Dit is eigenlijk hetzelfde verhaal als ik hierboven heb beschreven. In eerste instantie was mijn bedoeling dat gast gekoppeld worden aan een activiteit. Omdat dit voor de prijs van de gehele activiteitenboeking waardeloze informatie was en eveneens extra administratief extra werk was heb ik deze associatie ook weggelaten.

Transfer is een activiteit

Een transfer is een rit van of naar Stanley Safari Lodge gedaan door een Vendor of door de lodge zelf. Volgens de opdrachtgever kan een transfer als een activiteit beschouwd

worden, omdat een transfer immers ook door derden geleverd kan worden en er ook een ‘prijskaartje’ aanhangt, zodoende heb ik deze niet opgenomen als klasse in het klassediagram. In de toekomst zal er waarschijnlijk een veld als IsTransfer bijkomen om aan te geven of het over een transfer of een activiteit gaat. Een transfer valt nu onder de klasse ‘Service_Transfer’.

4.2.3 Identificeren van overervingen

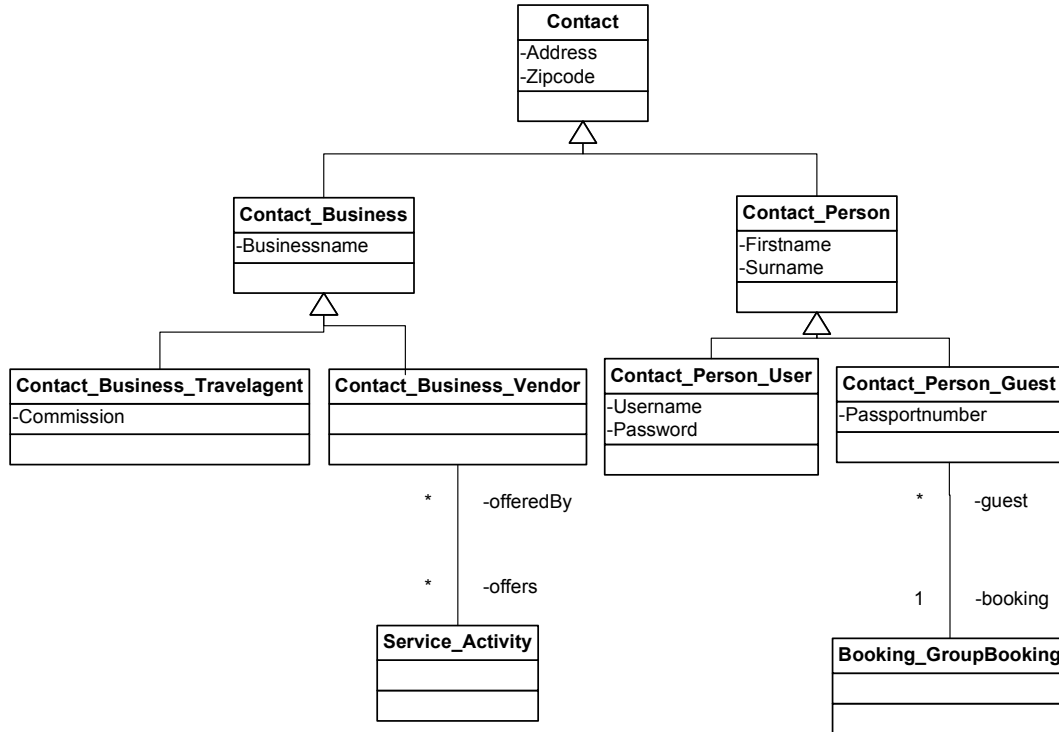
Het is in UML niet gebruikelijk om de overervingen te identificeren vlak na het identificeren van de relaties. Normaliter wordt dit pas gedaan na de identificatie van de attributen en operaties.

Echter, omdat ik al spoedig zag dat een aantal klassen veel overeen zullen komen en veel gemeenschappelijke attributen en operaties zouden hebben heb ik ervoor gekozen om reeds nu enkele overervingen aan te brengen. Vooral om dat ik nogal veel klassen heb gevonden was het handig om nu al wat generalisaties aan te brengen om het overzicht in het klassediagram te bewaren. Er waren in totaal reeds drie overervingen te onderscheiden. Service en product konden samengevoegd worden, omdat deze beide geleverd worden door een derde. Bovendien was al duidelijk dat zij ongeveer dezelfde attributen zouden bevatten (price, description).

De opdrachtgever had het idee geopperd had om in de toekomst contactgegevens als woonplaats, adres etc. van de gasten vast te leggen om eventueel later een ‘bedankbriefje’ of foto van de lodge op te sturen als herinnering. Met het oog op deze wens moesten dus ook de adresgegevens van de gasten bijgehouden worden. Omdat deze attributen ook voor de klassen Vendor, Travelagent gelden heb ik daar meteen een superklasse van gemaakt, namelijk Contact. Om maar gelijk het onderscheid te maken tussen een bedrijf/instelling en een persoon heb ik Contact opgesplitst in de subklassen Business en Person. Person kon dan ook weer onderverdeeld worden in User en Guest, immers van User moesten ook de adresgegevens vastgelegd worden. Het grote voordeel voor de opdrachtgever voor deze constructie is dat er meteen een adresboek werd aangelegd waar alle contactgegevens van alle contactpersonen werden vastgelegd.

Ook bij de klasse Business kon weer onderscheid gemaakt kunnen worden tussen Vendor en Travelagent.

Na toepassing van de codestandaarden die ik in het vorige hoofdstuk heb behandeld zijn dus de volgende super- en subklassen ontstaan: Contact, Contact_Business, Contact_Person, Contact_Business_Travelagent, Contact_Business_Vendor, Contact_Person_Guest, Contact_Person_User. Om bovenstaande te verduidelijken hieronder een fragment uit het klassediagram:



4.2.4 Aangeven van de multipliciteit en rolnamen

Nadat ik de associaties aangegeven heb, heb ik de multipliciteit en de rolnamen aangegeven bij de associaties. Voor de rolnamen heb ik de meest voor de hand liggende namen gekozen.

4.2.5 Identificeren van attributen

Nadat ik de basis van het klassediagram heb gemaakt ben ik op dezelfde wijze als waarop de kandidaatklasselijst tot stand is gekomen begonnen met het identificeren van de attributen.

4.2.6 Regels voor attribuutnamen

Alvorens te beginnen met het identificeren van de attributen heb ik een aantal naamgevingregels opgesteld waaraan ik me moest houden. Deze regels zijn:

Alle attribuutnamen en klassenamen worden in het klassediagram al in de Engelse taal geschreven. Dit leek me verstandig aangezien ik me ook voorgenomen heb om in de programmacode in het Engels te gaan schrijven. Het is immers waarschijnlijker dat, mocht de applicatie ooit aangepast worden, de code door een niet-Nederlandstalig persoon wordt aangepast dan door een Nederlandstalig persoon. Naast dat ik het klassediagram in het Engels maak, houd ik een vertaaltabel van de belangrijkste woorden (klassenamen, attribuutnamen, operatienamen) bij waar gemakkelijk is op te zoeken wat de Nederlandstalige vertaling van een bepaald woord is. Overigens zal de documentatie van de applicatie wel geheel in het Engels zijn.

Regels voor klassenaam

Een klassenaam begint met een hoofdletter en bestaat altijd uit de enkelvoudsvorm van een zelfstandig naamwoord. Mocht een klassenaam bestaan uit meerdere zelfstandig naamwoorden dan worden ook de daarop volgende zelfstandig naamwoorden geschreven met een hoofdletter.

Regels voor attribuutnaam

Alle attributen beginnen met een hoofdletter, een variabele kan een zelfstandig naamwoord, een voltooid deelwoord of een werkwoord (in de jivvorm) gecombineerd met een voltooid deelwoord (bijvoorbeeld ‘Creation’, ‘Created’ of ‘IsCreated’). Bovendien begint elk zelfstandig naamwoord met een hoofdletter. Verder moeten de attribuutnamen zo veelzeggend mogelijk zijn en worden afkortingen vermeden om onbegrip van de attributen te voorkomen.

Een aantal attributen kon ik uit de lijst van kandidaatklassenlijst van de vorige stap halen. Iedere kandidaat die ik niet als klasse gekozen hebt kan ik nu toewijzen aan één van de bestaande klassen als attribuut.

Voor de meeste klassen lag de keuze voor de attribuutnamen voor de hand. Zoals dat bij de klasse ‘Contact’ duidelijk het geval was. Alleen de klasse ‘Contact_Business_Travelagent’ had een opvallend extra veld, namelijk ‘Commission’, welke het percentage weergeeft van het geldbedrag dat de reisagent ontvangt als deze een boeking maakt bij de lodge.

De extra attributen bij de klasse ‘Contact_Person_Guest’ zijn ‘Passportnumber’, ‘IsChild’ en ‘Nationality’. De nationaliteit en paspoortnummer van de gast is erg belangrijk voor de manager van de lodge. Alléén met deze gegevens, en uiteraard de namen van de gasten, kan de manager van de lodge de visa aanvragen bij de desbetreffende grenspost. ‘IsChild’ is duidelijk. Het geeft aan of de gast een kind is ja of nee. De reden dat dit niet automatisch wordt gedaan aan de hand van een geboortedatum is omdat de grens af en toe vaag is. De ene keer registreert de manager een gast van veertien (officieel ligt bij Stanley Safari Lodge de lat op twaalf jaar) jaar als kind en de andere keer registreert hij deze als volwassene. Het onderscheid tussen een volwassene en een kind wordt gemaakt, omdat een kind een korting krijgt van vijftig procent als deze bij zijn ouders op de kamers slaapt. Dit is één van de eisen die gesteld is en te lezen is in het eisendocument.

Commentaar bij elke klasse

Ik heb de opdrachtgever geadviseerd om voor alle zekerheid bij elke klasse een attribuut ‘commentaar’ op te nemen. De reden hiervoor is dat er altijd wel een attribuut vergeten kan worden, welke zodoende dan in het commentaarveld opgenomen kan worden.

Attributen voor klasse ‘Booking’

Ik vond zelf de klasse ‘Booking’ een aparte behandeling verdiende, aangezien deze klasse de belangrijkste klasse van het systeem is. Het is immers een boekingssysteem. Voor een boeking wil de opdrachtgever de volgende gegevens vastleggen: bookingname, status, amountpaid.

Door wie de boeking is aangemaakt, welke gasten er bij de boeking behoren worden al in de associaties vastgelegd. De ‘Status’ geeft aan of de boeking ‘Provisional’, ‘Confirmation requested’ of ‘Confirmed’. Een boeking krijgt de status ‘Confirmed’ als de betaling van de boeking is binnengekomen op de rekening van de lodge. Een boeking heeft als standaard-status ‘Provisional’. Pas als de reisagent zeker is dat de gasten kiezen voor Stanley Safari Lodge zal de reisagent een aanvraag doen tot confirmatie, een zogenaamde ‘Confirmation request’.

Ik heb de opdrachtgever tevens geadviseerd om de volgende attributen op te nemen in klassen: ‘Bookingtime’, ‘DateInvoiced’ en ‘Confirmtime’. Aanvankelijk had ik alleen ‘IsInvoiced’ geopperd, maar uiteraard biedt ‘DateInvoiced’ veel meer informatie, namelijk de datum en zodra de datum ingevuld is, weet je ook dat de boeking gefactureerd is. De ‘Bookingtime’ geeft weer wanneer de boeking is aangemaakt. Een reisagent kan dus opvragen wanneer deze en door wie de boeking is aangemaakt. En ‘Confirmtime’, dit attribuut geeft de tijd aan wanneer de boeking geconfirmeerd is.

NumberOfPersons is redundant?

Ja, dat is redundant. Echter, het aantal personen móet bekend zijn als de factuur opgemaakt gaat worden. De factuur wordt voordat de gasten aankomen opgemaakt en verstuurd naar het reisbureau. Alle gegevens van de gasten hoeven dus nog niet bekend te zijn voordat de factuur opgemaakt wordt. Dus moet een boeking snel geconfirmeerd worden of heeft de reisagent geen tijd om snel alle gastgegevens in te vullen dan kan dat al gedaan worden.

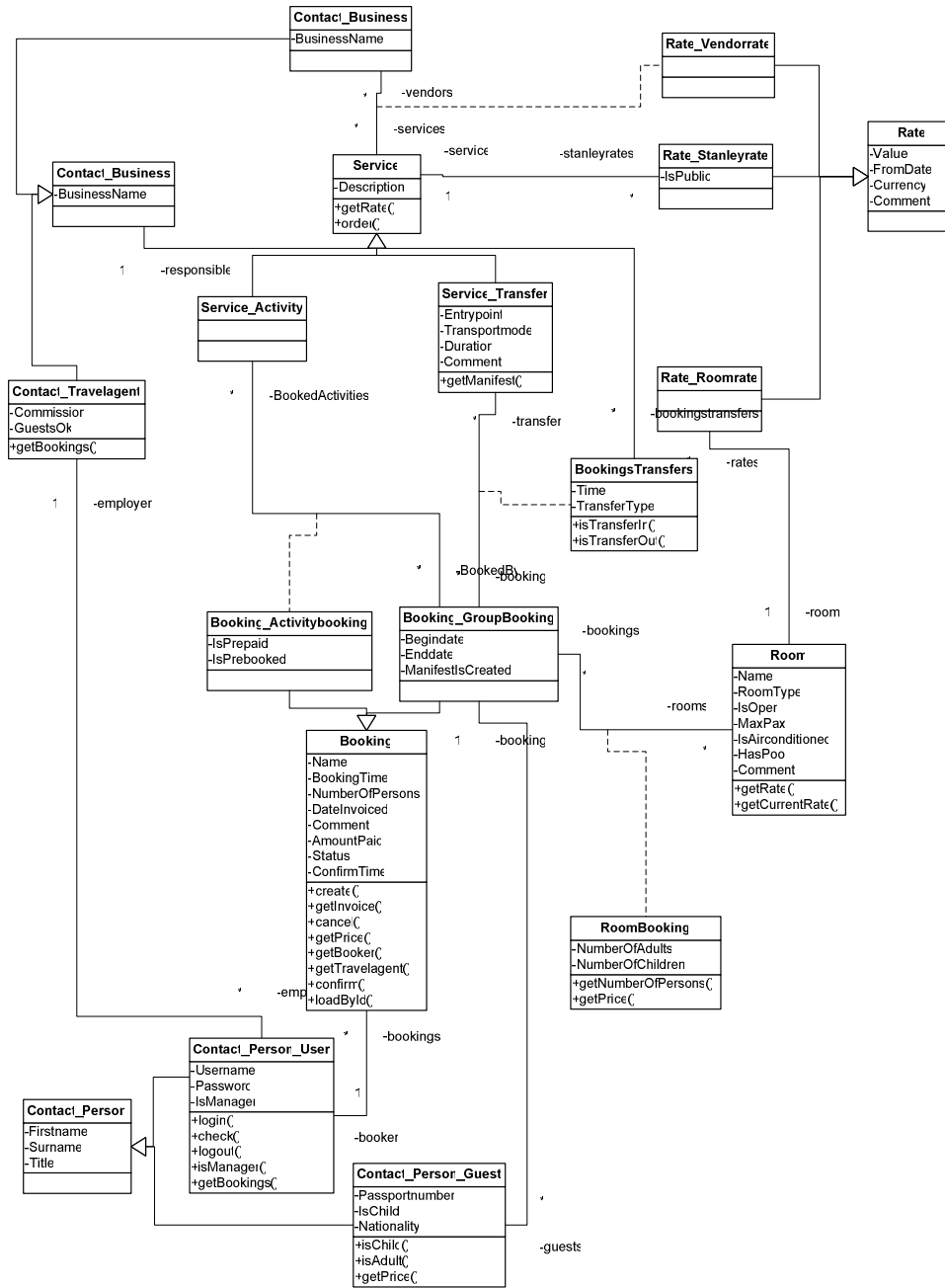
Rates

De ratesklasse heb ik besloten nog niet te implementeren, omdat dit een complexe zaak is, aangezien er per kamer en per aantal personen en ook weer per periode verschillende rates bestaan. Ook behoort dit niet tot de belangrijkste zaken van het boekingssysteem. De rates worden alleen voor de facturatiemodule gebruikt en deze gaat (waarschijnlijk) niet geïmplementeerd worden.

4.2.7 Identificeren van operaties

De operaties heb ik eigenlijk net zo gedaan als het identificeren van de attributen. Alleen heb ik daar geen lijst van gemaakt, omdat er tijdens het programmeren van het systeem (in de construction phase) zeker nog operaties bij zouden komen.

Hieronder vindt u het klassediagram zoals deze uiteindelijk geworden is.



4.3 RELATIONEEL DATABASEONTWERP

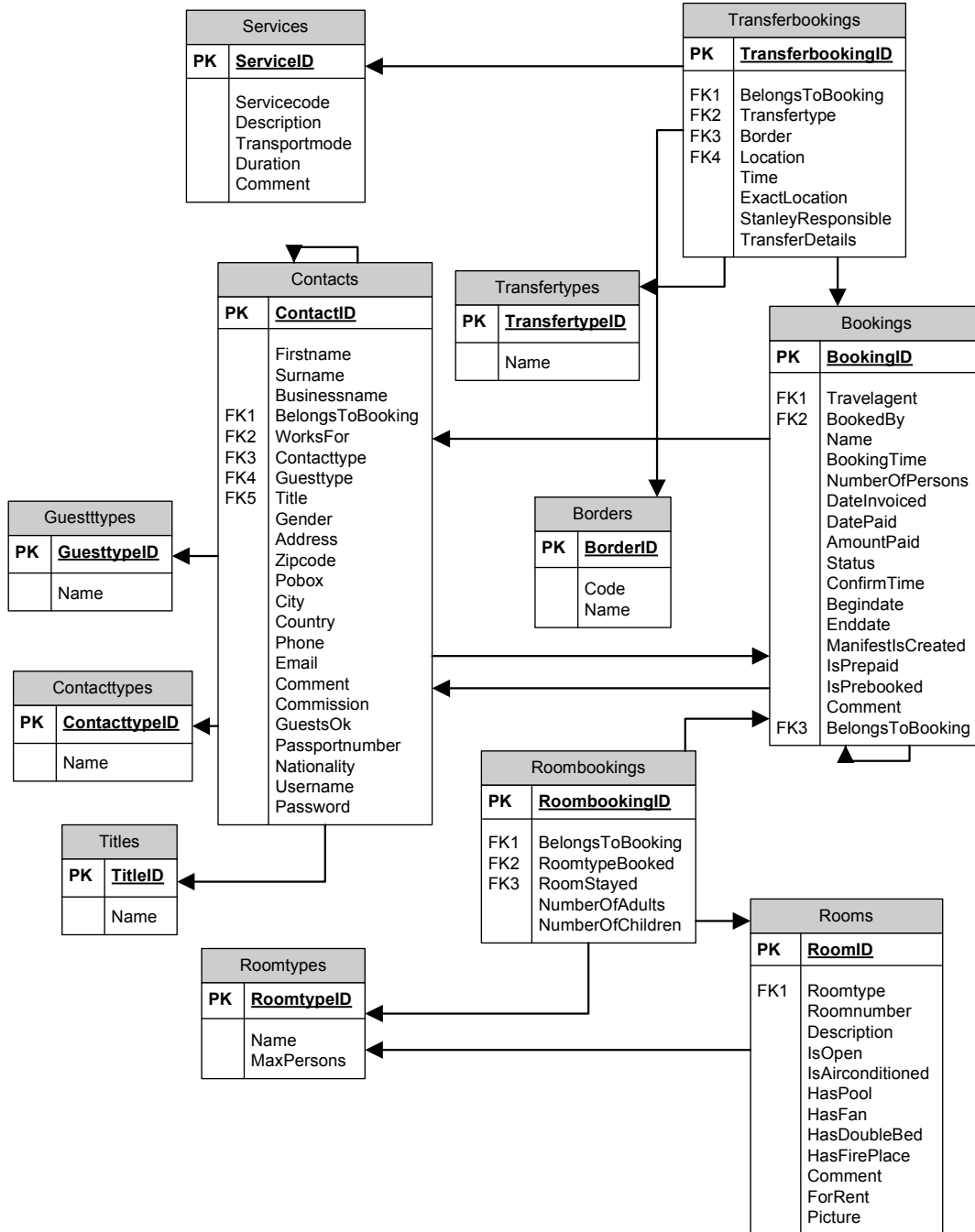
De volgende stap in deze fase is het maken van een databaseontwerp. Veel mensen verwarren het ontwerpen van een database met het maken van een klassediagram. Dit is op zich niet vreemd, ze hebben immers veel van elkaar weg en ze vormen beide de basis van het systeem. Zo wordt er vaak gezegd dat elke klasse gelijk is aan een tabel, echter zoals uit mijn klassediagram al blijkt heb ik een aantal subklassen gemaakt, zoals ‘Contact_Person_User’ en ‘Contact_Person_Guest’. Deze zullen niet worden verdeeld in twee tabellen, maar gewoon opgenomen worden in één tabel, bijvoorbeeld ‘contacts’. Het verschil tussen ‘Contact_Person_User’ en ‘Contact_Person_Guest’ wordt dan weergegeven door bijvoorbeeld een extra veld in de database op nemen als ‘IsUser’.

Een ander groot verschil met een databaseontwerp is dat een klassediagram naast attributen ook operaties bevat. Operaties worden, zoals algemeen bekend, niet opgenomen in een database. Een database is een *base* voor puur en alleen *data*. Een operatie zegt iets over de werking van het systeem en doet dus een bewerking aan de hand van de attributen.

Het ‘vertalen’ van de overervingen (uit het klassediagram) naar een juist relationeel databaseontwerp kon op verschillende manieren. Ik heb een manier gekozen, waardoor zo min mogelijk redundantie optreedt. Dus heb om alle subklassen van een superklasse in één tabel op te slaan. Bijvoorbeeld de superklasse Contact. Alle subklassen daarvan heb ik opgeslagen in één tabel ‘contacts’. Het veld ‘contacttype’ geeft dan aan tot welke subklasse het record behoort.

Naast de gangbare attributen bevat een klassediagram afgeleide attributen welke bij een database ‘not done’ zijn omdat dat redundantie van gegevens veroorzaakt. Een voorbeeld van een afgeleid attribuut in het klassediagram is: *NumberOfRooms*. Deze is een afgeleid attribuut van de som van *roombookings* die aan de boeking is gekoppeld.

Op de volgende pagina ziet u de het databaseontwerp, zoals deze is geïmplementeerd.



4.4 IMPLEMENTEREN DATABASE

Het implementeren van de database heb ik gedaan via phpMyAdmin. Dit is een stuk gemakkelijker dan zelf het implementatiemodel schrijven. Vanuit het databasemodel uit de vorige paragraaf heb ik de database geïmplementeerd. Voor het definiëren van de typen heb ik één standaard gebruikt. Namelijk:

Primary key = int(4)
Short String = varchar(50)
Medium String = varchar(100)
Long String = varchar(255)
Boolean = tinyint(1) met 1 is true en 0 is false
Commentfields altijd longtext

4.4 OBJECTGEORIËNTEERDE DATABASELAAG

Na het implementeren van de domeinklassen kunnen wijzigingen in database komen. Veldnamen kunnen veranderen, veldnamen kunnen verwijderd worden of er kunnen velden worden toegevoegd. Zelfs tabelnamen kunnen om één of andere reden gewijzigd worden. Door de SQL-queries in de domeinklassen op te nemen kan het veel tijd vergen om alle queries te herschrijven als er een veldnaam of een tabelnaam gewijzigd wordt. Om te vermijden dat hele queries herschreven moeten worden heb ik zelf een objectgeoriënteerde databaselaag geschreven, waardoor je op een gemakkelijke wijze tabellen uit een database kunt benaderen. Hieronder staan een aantal voorbeelden voor het gebruik van de databaselaag. Respectievelijk wordt weergegeven hoe een record kan worden toegevoegd, een record kan worden bewerkt en een record kan worden verwijderd.


```
1  <?php
2
3      /**
4       * Add a contact to the table `contacts`
5       */
6      $Contact = new mwhTable('contacts');
7      $Contact->setValue('Firstname', 'Kees') ;
8      $Contact->setValue('Surname', 'Kist') ;
9      $Contact->write();
10
11     /**
12      * Edit a record in table `contacts` with ContactID 65
13      */
14     $Contacts = new mwhTable('contacts');
15     $Contact->setValue('ContactID', 65);
16     $Contact->setValue('Firstname', 'Piet') ;
17     $Contact->setValue('Surname', 'Schrijvers') ;
18     $Contact->write();
19
20     /**
21      * Delete a record in table `contacts` with ContactID 65
22      */
23     $Contacts = new mwhTable('contacts');
24     $Contact->setValue('ContactID', 65);
25     $Contact->delete();
26  ?>
```

Overerving door domeinklasse

De eigenschappen en operaties van de databaselaag kunnen worden overgeërfd door een domeinklasse. Elke domeinklasse van de boekingsapplicatie, die direct gekoppeld is aan een tabel uit de database kan dus gebruik maken van de databaseklasse. Dit kan op de volgende eenvoudige manier bewerkstelligd worden.

Op deze manier kan op gemakkelijk wijze data uit de tabel gelezen en beschreven worden. Het enige wat ik nu moest doen om de gegevens van de klasse Contact_Person_User op te vragen was de functies zoals in figuur staan aangegeven aan te roepen.

Het voordeel om de domeinklasse de databaseklasse te laten overerven is dat er nog functies en attributen toegevoegd konden worden.

```
1  <?php
2
3  /**
4   * Contact_Person_User is a subclass of the databaselayer mwhTable
5   */
6  class Contact_Person_User extends mwhTable
7  {
8      /**
9       * Constructor, which initializes the class and connects it to
10      * the table 'bk_contacts'.
11      */
12      function Contact_Person_User()
13      {
14          $this->init('bk_contacts');
15      }
16  }
17
18  ?>
19
```

Voorbeeld gebruik van de databaselaag in een klasse

Associaties

Ook kunnen associaties op een eenvoudige wijze worden toegevoegd. Onderstaand is een code te zien welke aangeeft hoe gemakkelijk dit is.

```
8      $this->addAssoc('Guesttype', 'Guesttype', 'Guesttype', false);
9      $this->addAssoc('Title', 'Title', 'Title', false);
```

Voorbeeld voor het toevoegen van een associatie bij een klasse

Door middel van de functie addAssoc worden de associaties Guesttype en Title opgenomen in de klasse Contact_Person_Guest. De parameters die aan deze functie meegegeven moeten worden zijn: De naam van de associatie (Guesttype), de naam van de geassocieerde klasse (Guesttype) en de naam van de vreemde sleutel (eveneens Guesttype). Daarnaast kan (optioneel) nog een boolean meegegeven worden als parameter.

Extra tabellen

De databaseklasse kan automatisch een selectbox genereren op basis van de primary key en een variabel veld van een tabel. Omdat ik deze ‘feature’ heb ingebouwd heb ik een aantal extra tabellen toegevoegd, zodat ik daar gemakkelijk automatisch selectboxen van kan laten genereren. De tabellen zijn:

- Contacttypes: Manager, User, Guest, Vendor etc.
- Titles: Miss, Ms, Mr etc.
- Guesttypes: Adult, Child
- Transfertypes: IN, OUT, Activity

4.5 BIJWERKEN PROJECTDOCUMENTEN

Zoals ik in het hoofdstuk ‘Inception phase’ al heb aangegeven zijn er in RUP een aantal ‘levende’ documenten. Hieronder vallen het eisendocument, het risicodocument, het

visiedocument en het projectplan. Deze dienen tijdens of na elke fase waar nodig bijgewerkt te worden.

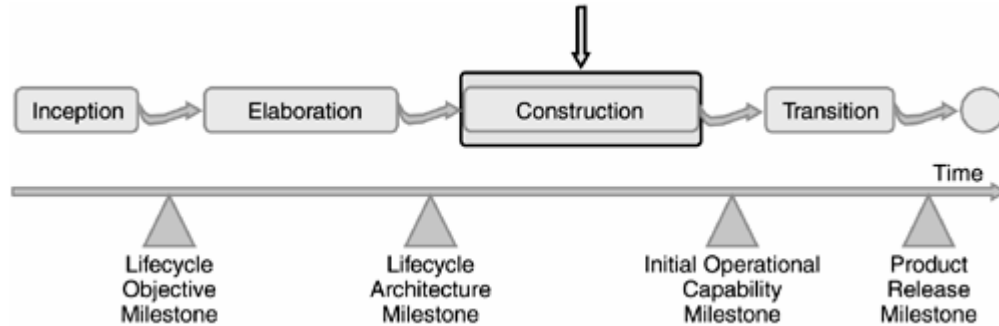
De documenten die ik tijdens het project heb onderhouden zijn het eisendocument en het risicodocument. Telkens als er een nieuwe eis tegen het licht kwam of dat er een nieuw risico op kwam, noteerde ik die in het desbetreffende document. Bij elk risico of eis staat een datum aangegeven wanneer deze is toegevoegd.

Documenten die ik aan het einde van deze fase heb bijgewerkt zijn het projectplan en het visiedocument.

Het projectplan moest bijgewerkt worden omdat deze fase, de elaboration phase, is uitgelopen, waardoor ik later ben begonnen met de construction phase. Het visiedocument is minimaal gewijzigd. De use case ‘helpfunctie’ is verwijderd, omdat die minder belangrijk wordt geacht. Het is belangrijk om helpteksten toe te voegen aan de applicatie, maar veel formulier in het boekingssysteem zullen voor zich spreken. Naast deze wijziging zijn de overzichten gedetailleerder beschreven.

5. DOORLOPEN CONSTRUCTION PHASE

Dit hoofdstuk gaat over de fase Construction phase binnen de Rational Unified Proces. Onderstaan figuur geeft aan ‘waar’ ik me tijdens het project bevond. Zoals de pijl aangeeft zit ik hier in de Construction Phase, de Engelstalige naam voor de construction phase.



Het algemene doel van de construction phase is dus de ‘Initial Operational Capability Milestone’.

Doelstellingen construction phase

Het doel van de construction phase is het uitbrengen van een eerste operationele versie van het systeem door de overgebleven use cases en eisen te beschrijven, de implementatie verzorgen en de software te testen.

5.1 ITERATIEPLAN CONSTRUCTION PHASE

Het iteratieplan heb ik gemaakt op basis van mijn ervaring in het programmeren in PHP en het objectgeoriënteerde programmeren.

Volgens het initiële projectplan zou de constructie op 16 december 2004 van start gaan en zou tot 2 februari duren. Echter omdat de elaboration phase 5 dagen was uitgelopen, begon en eindigde de construction phase ook 5 dagen later.


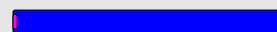


Timeboxing

De planning van de construction phase heb ik bepaald aan de hand van belangrijkheid van de use cases, welke ik samen met de opdrachtgever heb bepaald. De prioriteitstelling heb ik gedaan aan de hand van het MoSCoW-principe. De belangrijkste (Minimal Usable Set, de M en C) heb ik opgenomen in de eerste iteratie. De rest heb ik opgenomen in de tweede en derde iteratie.

Use case	Prioriteit	Iteratie
Inloggen	M	I
Controleren beschikbaarheid	M	I
Algemene gegevens	M	I
Transfers	M	I
Gasten	M	I
Boeking zoeken	S	II
Defragmenteren	S	II
Overzicht beschikbaarheid	C	III

en status		
Overzicht ‘white board’	C	III
Activiteiten	W	IV
Facturatie	W	IV

De planning van de construction phase op basis van de bovenstaande artefacten ziet er als volgt uit:

ID	Task Name	Dec 2004				Jan 2005				Fet 2005				Mar 2005		
		12/5	12/12	12/19	12/26	1/2	1/9	1/16	1/23	1/30	2/6	2/13	2/20	2/27	3/6	3/13
1	Constructiefase															
2	1 Boekinger															
3	2 Manager															
4	3 Overzichter (Facturen manifests)															

Verder in dit hoofdstuk wordt per paragraaf elke activiteit besproken die in deze fase uitgevoerd is. Allereerst in de volgende paragraaf een korte toelichting over UML waarin ik aangeef welke diagrammen ik ga gebruiken en welke niet.

5.2 PRODUCTDOCUMENTATIE

Voor de documentatie heb ik gebruik gemaakt van de gratis verkrijgbare PhpDocumenter via pear.php.net. Hiermee kan, als de wordt voldaan aan de codeconventies die ik in het projectplan heb aangegeven, automatische documentatie van de programmacode worden gegenereerd. De documentatie kan op deze manier in de vorm van een PDF-document of een HTML-website verkregen worden.

6. ITERATIE I - TRAVELAGENTS

Travelagents zijn de reisbureaus die voor hun eigen cliënten (dus de toekomstige gasten van Stanley) een kamer boeken bij de lodge. Reisbureaus kunnen zich over de gehele wereld bevinden, over het algemeen bevinden ze zich in Zuid-Afrika, Europa en Noord-Amerika. Reisbureau's krijgen per boeking die zij maken een commissie variërend van 15 tot 25 procent. Zo krijg bijvoorbeeld MilesTravel in Eindhoven, waar ik in december op bezoek ben geweest, 20 procent van het totale boekingsbedrag.

Uiteraard krijgt niet elk reisbureau een gebruikersnaam en wachtwoord. Alleen reisbureau's die geregeld bij Stanley boeken met een minimum van ongeveer twee keer per twee weken. Het lijkt erop dat er dan weinig van de applicatie gebruik gemaakt gaat worden, echter bij een boeking die via e-mail gemaakt wordt kunnen soms wel twintig e-mailberichten horen, wat de gehele conversatie lang en onoverzichtelijk maakt.

Het doel van deze iteratie is om te komen tot het belangrijkste gedeelte van het boekingssysteem – het aanmaken en bewerken van een boeking, zodat reisbureaus via een website in kunnen loggen en een boeking kunnen aanmaken en bewerken.

Het artefact dat bij deze iteratie opgeleverd wordt is een groot gedeelte van het boekingssysteem. Daarom heb ik besloten om deze iteraties op te delen in kleine ‘mini-iteraties’:

- Inloggen
- Boekingen
 - Controleren beschikbaarheid
 - Algemene gegevens boekingen
- Transfers
- Gasten
- Confirmatie

Ik zal nu per mini-iteratie uitleggen wat mijn werkzaamheden hiervoor waren. De werkzaamheden per mini-iteratie zullen bij sommigen bestaan uit een use case, een sequence-diagram, een schermontwerp, de implementatie van de code en de schermen. Het use-case-diagram toont hoe het systeem gebruikt kan worden door externe entiteiten zoals menselijke gebruikers. Ik begin expres met dit type diagram, aangezien RUP een ‘use-case-driven’ ontwikkelmethode is. Deze zijn vooral in de inception phase van belang. Het spreekt dus voor zich dat ik hier gebruik van maak.

Het sequencediagram toont de volgorde in tijd van de boodschappen die in het systeem verstuurd en ontvangen worden. Dit was zeker van belang bij het implementeren van de verschillende use cases.

Ik zal ze van bovenstaande volgorde langslopen. Ik begin dus met de use case ‘Inloggen’

6.1 USE CASE INLOGGEN

Om het boekingssysteem goed afgeschermd te houden van onbevoegden moest er eerst ingelogd kunnen worden. De eerste kleine iteratie in behelst dus het inloggen in het systeem. Van de use case inloggen heb ik use case template en sequentiediagram gemaakt die u kunt vinden in bijlage A en bijlage B.

Ontwerp user interface

Het ontwerpen van een userinterface voor een loginscherm is gemakkelijk. Het belangrijkste is dat het een twee velden, voor de gebruikersnaam en het wachtwoord, bevat en een knop om de ingevoerde gegevens te verzenden.

Implementatie code

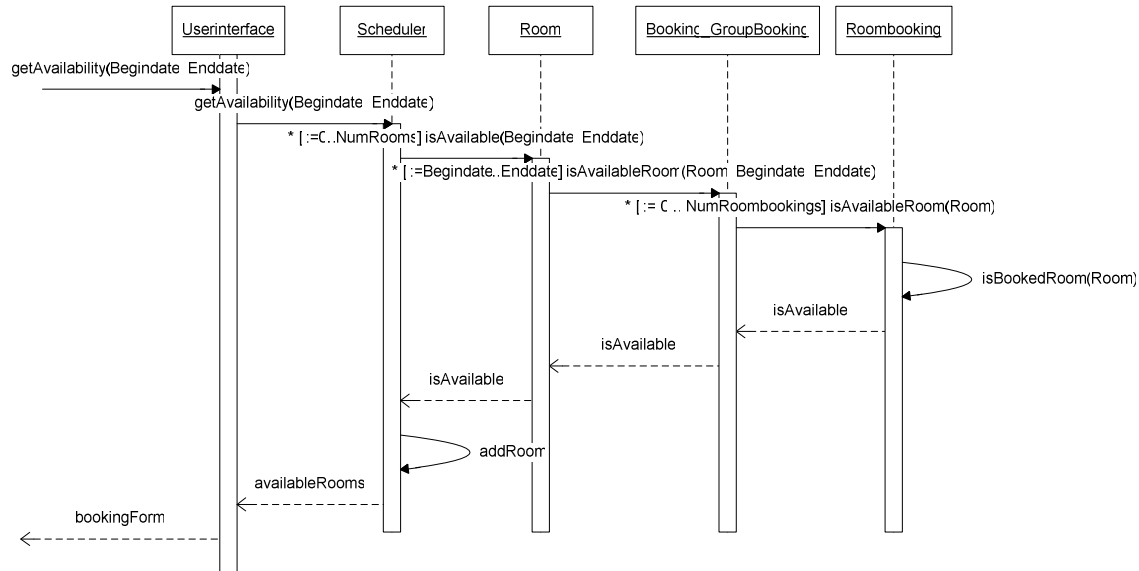
Aan de hand van scripts die ik via het Internet kon downloaden heb ik een eigen loginscript gemaakt om zelf ook iets te leren over sessies en beveiliging in PHP. De scriptingtaal PHP, die ik gebruik voor het schrijven van het boekingssysteem bleek goede ondersteuning te hebben voor het beveiligen van een PHP-pagina's. Zoals al uit het klassediagram blijkt is er een klasse `Contact_Person_User`, waar de gebruikersnaam en het wachtwoord van de gebruiker staan. De gebruikersnamen en wachtwoorden staan dus opgeslagen in de tabel `contacts`.

6.2 USE CASE CONTROLLEREN BESCHIKBAARHEID

Het eerste wat een medewerker van een reisagent doet (gebruiker) voordat hij (of zij) een boeking gaat aanmaken is het controleren van de beschikbaarheid van de kamers voor de periode die hij in gedachte had. De use case template hiervan ziet er als volgt uit:

Titel: Controleren beschikbaarheid	
Samenvatting	Controleren van de beschikbaarheid van kamers
Actor	Reisagent, manager (gebruiker)
Precondities	De gebruiker is ingelogd
Postcondities	Het systeem toont een boekingsformulier
Scenario	<ol style="list-style-type: none">1. De gebruiker voert een begindatum en einddatum in2. Het systeem controleert de ingevoerde data op correctheid3. Het systeem controleert de beschikbaarheid4. Het systeem toont een boekingsformulier met daarin de aangegeven beschikbaarheid.

En het sequentiediagram als volgt:



Uiteraard kan het resultaat bookingForm dat ‘teruggegeven’ wordt aan gebruiker ook een melding zijn dat er geen beschikbare kamers zijn voor de aangegeven periode of dat de ingegeven data niet correct zijn.

Implementatie

Zoals uit het sequencediagram blijkt wordt er per kamertype gekeken hoeveel er beschikbaar zijn op een bepaalde datum. Mochten er geen kamers vrij zijn in de aangegeven periode dan krijgt de userinterface een errorstring terug met bijvoorbeeld ‘Sorry er zijn geen kamers vrij in de door u aangegeven periode.’. Verder worden er nog een aantal controles, zoals in de use cases al beschreven, uitgevoerd op de ingevoerde data. Een einddatum mag bijvoorbeeld niet eerder zijn dan een begindatum (we kunnen immers nog steeds niet in de tijd reizen!).

6.3 USE CASE ALGEMENE GEGEVENS BOEKINGEN

Voor het systeem bestaan er twee verschillende gebruikers: de manager en de medewerker van het reisbureau, oftewel normale gebruiker. Beide soorten gebruikers kunnen een nieuwe boeking invoeren, echter een manager zal meer informatie moeten invoeren dan de normale gebruiker.

Na het controleren van de beschikbaarheid kan er een boeking ingevoerd worden

Onder algemene gegevens wordt verstaan:

- de boekingnaam
- begindatum en einddatum, welke reeds ingegeven zijn bij het controleren van de beschikbaarheid.
- Origineel reisbureau (een reisbureau kan via een ander reisbureau een kamer boeken, zodat een reisbureau als een soort tussenpersoon kan optreden) (alleen manager)
- Initiële boekingsstatus (alleen manager)
- Aantal personen

- En het aantal kamers van de beschikbare kamertypen.
- Eventueel commentaar (prebooked activities of eventuele vegetariërs)

Van deze use case heb ik ook geen use case template en sequentiediagram gemaakt, omdat het hier alleen het invoeren en verifiëren van gegevens betreft.

6.4 USE CASE TRANSFERS

Als er nieuwe gasten aankomen op de lodge moeten ze natuurlijk van het vliegveld (of een andere lodge, locatie) naar Stanley komen. Dit gaat niet vanzelf. Stanley biedt de mogelijkheid om dit voor hen te regelen. Zo kunnen gasten vanaf bijvoorbeeld het vliegveld, vanaf de grens met Zimbabwe of Botswana opgehaald worden. Daarvoor moeten Stanley wel weten of zij ook daadwerkelijk verantwoordelijk is voor de transfer, waar zij de gasten moet ophalen en vooral hoelaat zij de gasten op moet halen. Ook is het belangrijk om te weten vanuit welke grenspost de gasten Zambia binnenkomen omdat Stanley voor het dat de visa regelt. Als er niet is aangegeven waar de gasten het land binnenkomen kan Stanley geen visa voor hen regelen.

Implementatie

Invoeren is eigenlijk niet het juiste woord voor deze use case. Het gaat in werkelijkheid om het wijzigen van de transfergegevens. Zoals al bij het sequencediagram voor het aanmaken van een nieuwe boeking blijkt wordt bij het invoeren van een nieuwe boeking worden tegelijkertijd twee records toegevoegd aan de tabel transferbookings. De reden hiervoor is dat de gebruiker bij het invoeren van een nieuwe boeking reeds aangeeft of Stanley Safari Lodge verantwoordelijk is voor de transfer of dat een ander bedrijf daarvoor verantwoordelijk is. Als Stanley Safari Lodge verantwoordelijk is zijn er andere gegeven vereist dan wanneer een ander bedrijf verantwoordelijk is voor de transfer. Als Stanley Safari Lodge verantwoordelijk is voor de transfer dan is de exacte locatie nodig waar de gasten opgehaald moeten worden, als Stanley Safari Lodge niet verantwoordelijk is dan is alleen grenspost nodig waar de gasten Zambia binnenkomen voor het aanmaken van de manifesten. De transfers worden dus in een aparte tabel opgeslagen. Er kunnen namelijk meerdere soorten transfers plaatsvinden dan alleen een transfer-in of een transfer-out. Gasten kunnen een transfer boeken naar bijvoorbeeld de Victoria Falls, naar de stad of een andere locatie. De kosten voor deze transfers worden wel in rekening gebracht en komen aan het einde van het verblijf bij de lodge op de factuur te staan. Bij elkaar zijn er dus drie soorten transfers die mogelijk zijn – transfer-out, transfer-in en transfer-activity (voor de overige transfers, welke onbepaald mogelijk zijn). In de database worden deze transfertypen in de tabel transfertypen opgeslagen onder de namen IN, OUT en Activity.

6.5 USE CASE GASTEN

Uiteraard behoren er gastgegevens bij een boeking. De klasse Contact_Person_Guest is, zoals al uit de naam blijkt een subklasse van Contact_Person, welke weer een subklasse is van Contact. Alle attributen die bij de klasse Contact_Person en Contact horen, behoren dus ook bij de Contact_Person_Guest (dus gast). Echter de belangrijkste gegevens van de gasten zijn: Title, Firstname, Surname, Nationality, Passportnummer voor het aanvragen van de visa en of het een Child of een Adult betreft voor de facturering. Kinderen krijgen namelijk vijftig procent korting als zij bij hun ouders in een kamer verblijven. Een use case schrijven voor alleen het invoeren of wijzigen van een aantal gegevens is nutteloos omdat het alleen de gegevens ingeven is en opslaan.

7. ITERATIE II - MANAGER

Deze iteratie richt zich op de functionaliteiten van het boekingssysteem ten behoeve van de operationeel manager van de lodge. Veel use cases die door de travelagent gemaakt zijn komen grotendeels overeen met de use cases die voor de manager gemaakt moet worden. Echter voor de manager zijn een aantal overzichten van belang. Bovendien zal de manager voor het invoeren van een boeking andere gegevens in moeten voeren. Daarnaast komt het onderhoud van de kamers, de kamertypen, de gebruikers en de reisbureaus er nog bij. Dit heb ik wel op ‘een laag pitje’ gezet, omdat de opdrachtgever heeft aangegeven dit ook het programma phpMyAdmin te kunnen doen. De opdrachtgever kan met dit programma overweg en zodoende de gegevens van de kamers, kamertypen etc. wijzigen.

In deze iteraties worden de volgende use cases behandeld:

- Inloggen
- Boekingen
- Boekingsstatus wijzigen
- Boeking zoeken

In dit hoofdstuk zal ik per use case aangeven hoe ik deze ontworpen en geïmplementeerd heb.

7.1 USE CASE INLOGGEN

De use case inloggen in deze iteratie is een eenvoudig. Immers deze use cases is inmiddels al geïmplementeerd in de vorige iteratie. Ik hoefde alleen wat aan de code aan te passen om ervoor te zorgen dat de managers naar een ander pagina worden doorgestuurd dan de normale reisagenten.

7.2 USE CASE BOEKINGEN

De manager kan ook telefonisch of via e-mail directe boeking aannemen van gasten. Daarnaast zijn er nog een aantal reisbureau’s die bijvoorbeeld slechts eenmaal per jaar een kamer bij de lodge boeken en geen gebruikersnaam en wachtwoord nodig hebben. De manager moet dus ook de mogelijkheid hebben om boekingen in te voeren.

De use case boekingen komt wel grotendeels overeen met die van de travelagents. Het enige wat verschilt is dat de manager andere gegevens in moet voeren. Extra gegevens als boekingsstatus, boekingsagent en travelagent moeten aangegeven worden. Boekingsagent is er als extra veld tijdens de construction phase bijgekomen (toevoeging aan eisendocument) en geeft aan door welk originele reisbureau de kamer is geboekt. Reisbureau Janssen uit Groningen kan bijvoorbeeld een reis boeken bij Miles Travel uit Den Bosch. Reisbureau Janssens is dan het originele reisbureau. Deze gegevens zouden interessant kunnen zijn voor marketingactiviteiten.

Afgezien van het boekingoverzicht komt een boekingen aanmaken en bewerken overeen met boekingen aanmaken en bewerken voor een travelagent. Het boekingsoverzicht toont alle boekingen van alle reisbureaus van een maand geleden en later. De manager heeft de mogelijkheid om via een selectbox alle boekingen te selecteren of de boeking per reisbureau te selecteren.

7.3 USE CASE BOEKINGSSTATUS WIJZIGEN

In het visiedocument heeft u kunnen lezen dat dit een belangrijke use case is. In de vorige paragraaf staat dat de manager als extra gegeven de boekingsstatus van een boeking moet ingeven. Daarmee is deze use cases dus eigenlijk ook al gedekt.

7.4 USE CASE BOEKING ZOEKEN

Tijdens het implementeer van de overige use cases bleek het zinvol om er een zoekfunctie bij te bouwen, omdat het na het toevoegen van een bepaalde boeking moeilijk was deze terug te vinden. De use case template van deze use case kunt u vinden in bijlage, omdat het een dermate eenvoudige functie was. Het scherm ziet is eenvoudig van opzet zoals u hieronder kan zien:



De gebruiker kan zoeken op bookingname en op boekingen die zijn geboekt tijdens een bepaalde periode.

8. ITERATIE III - OVERZICHTEN

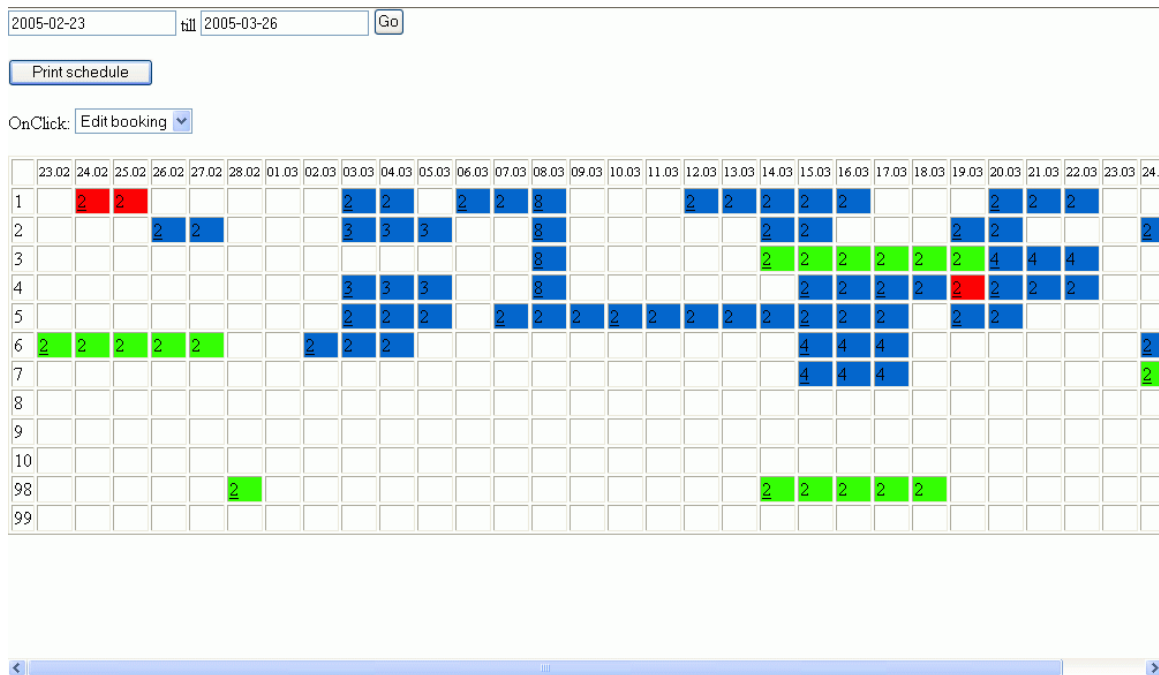
De laatste iteratie richt zich op de operationele overzichten die nuttig kunnen zijn voor de manager en waarmee snel de beschikbaarheid van kamers voor een bepaalde periode gecontroleerd kan worden.

In dit hoofdstuk zullen de volgende use cases behandeld worden:

- Overzicht beschikbaarheid en status
- Overzicht ‘White board’
- Defragmenteren

8.1 USE CASE OVERZICHT BESCHIKBAARHEID EN STATUS

Om in één oogopslag te kunnen zien wat de beschikbaarheid van de kamers voor een bepaalde periode is, had de manager behoefte aan een overzicht die deze behoefte dekte. Het overzicht moest er ongeveer net zo uitzien als het overzicht die het Excelsheet, zoals weergegeven in hoofdstuk 2, liet zien. Hieronder ziet u een afbeelding met dat het overzicht weergeeft.



Groen geeft een ‘Provisonal’ boeking aan, Blauw geeft een boeking aan waarbij de confirmatie al is aangevraagd (‘Confirmation requested’) en rood geeft een geconfirmeerde boeking aan (‘Confirmed’), waarbij al gefactureerd is naar het reisbureau. De nummers in het overzicht betekenen het aantal personen dat bij de boeking hoort. De getallen aan de linkerkant van het overzicht stellen de kamernummers voor. Kamers 1 t/m 7 zijn in gebruik. Kamers 8 t/m 10 zijn in aanbouw en kamernummers 98 en 99 zijn wachtkamers en kunnen tijdelijk gebruikt worden tijdens de defragmentatie.

8.2 USE CASE OVERZICHT ‘WHITE BOARD’

Zoals in hoofdstuk 2 al aangegeven werkt het personeel bij Stanley Safari lodge met een wit bord met de daarop de kamerbezetting, zodat elk personeel weet wanneer en in welke kamer er hoeveel gasten zitten.

Het zou dus erg goed van pas komen als het schema dat op het bord komt direct uit het systeem kom ‘rollen’, zodat de manager alle gegeven letterlijk kan overschrijven.

Hiervoor heb ik een php-pagina gemaakt, welke het desbetreffende overzicht genereert.

Het use case en sequentiediagram zijn te vinden in bijlage A en B. Het overzicht komt er zoals onderstaand figuur uit te zien:

2005-02-24

2005-03-03

Go

Print schedule

	Thu 02-24 ⁴	Fri 02-25 ⁴	Sat 02-26 ³	Sun 02-27 ²	Mon 02-28 ²	Tue 03-01	Wed 03-02 ²
1	IN: 1 BBKC(BBK Airport) 11h00 (ex Muchenje) Uhe x 2 (difficult guests, 24: Cruise o...)	Uhe x 2 (difficult guests, 24: Cruise o...)	Uhe x 2 (difficult guests, 24: Cruise o...) OUT: 1 LVI(LVI Airport) around 12h (to Rovos Air (before 2 Tigermo...))				
2			IN: 0 BBK by ASL (ex Impalila by ASL) Muir x 2	Muir x 2	Muir x 2 OUT: 1 LVI(LVI Airport) 13h35 (ce203)		
3							
4							
5							
6							IN: 1 LV10 Mcintyre x 2 (Mr M. McIntyre and Ms C Aymler)
7							
8							
9							
10							

Internet

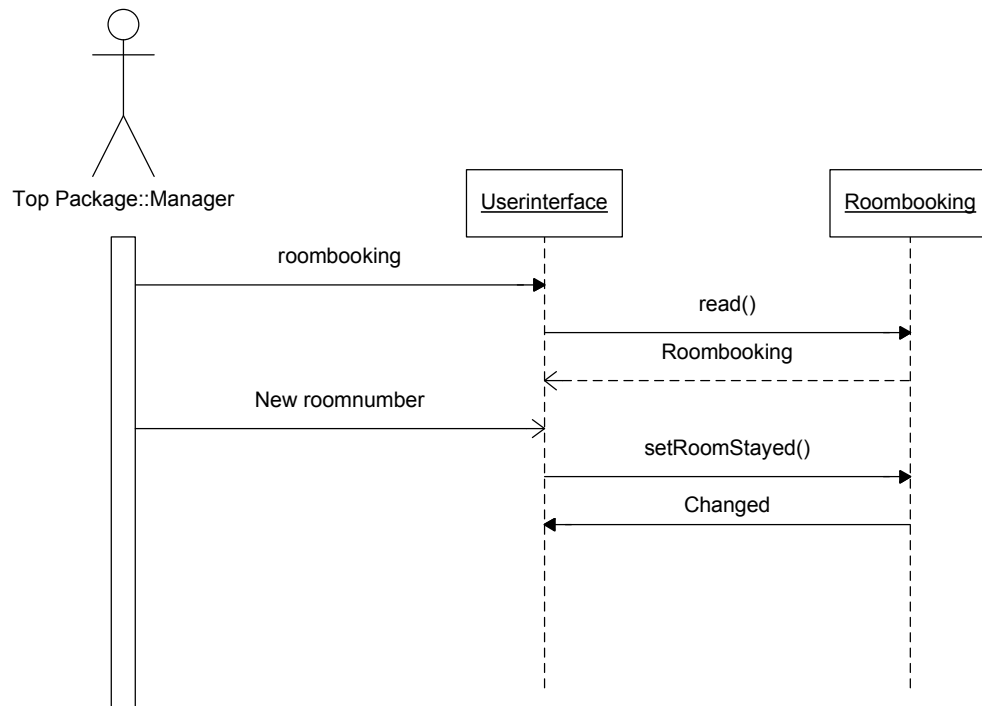
8.3 USE CASE DEFRAGMENTEREN

Via het schedule die ik reeds gemaakt heb kan de operationeel manager op een boeking klikken en hierin de kamer wijzigen waar de gasten in verblijven wijzigen. Echter, het is niet mogelijk om het kamertype waarin de gasten verblijven te wijzigen. Dat mag niet, omdat dat van invloed is op de prijs die de gast betaald. Een gast betaalt bijvoorbeeld meer voor een suite dan voor een normale kamer.

De use case template voor het defragmenteren ziet er zo uit:

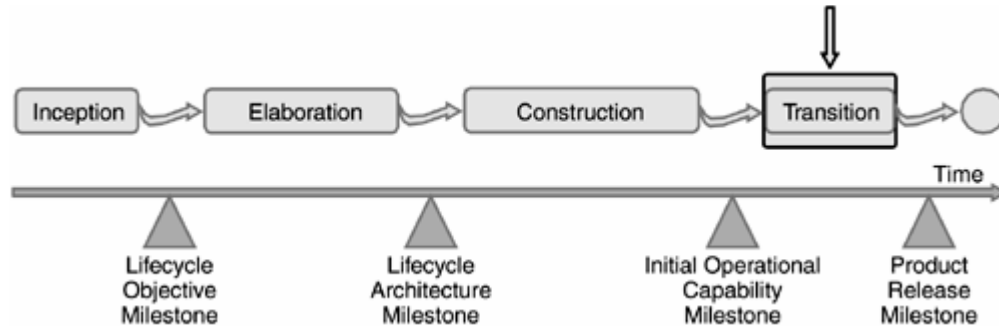
Titel: Defragmenteren	
Samenvatting	Verplaatsen van gasten naar andere kamers zodat de boekingen beter op elkaar aansluiten. (defragmenteren)
Actor	Manager
Precondities	De manager is ingelogd.
Postcondities	Bezettingsgraad is gewijzigd.
Scenario	<ol style="list-style-type: none">1. Het systeem toont een boekingsoverzicht.2. De manager selecteert een boekingsitem.3. Het systeem toont de gegevens behorende bij het boekingsitem.4. De manager verandert het kamernummer.5. Het systeem controleert of het kamernummer op de reeds ingevoerde data vrij is.6. Het systeem registreert de wijzigingen en maakt melding van de wijziging. [A1]7. De manager gaat terug naar het boekingsoverzicht.8. Het systeem toont het nieuwe boekingsoverzicht.
Alternatieven	[A1] Als de kamer niet vrij is maakt het systeem daar melding van en moet teruggegaan worden naar 3.

En het sequentiediagram:



9. TRANSITION PHASE

De laatste fase van RUP is de transition phase. Onderstaand figuur geeft aan waar me op dat moment bevond tijdens het project.



Het algemene doel van de transition phase is dus de ‘Product Release Milestone’. De mijlpaal voor de definitieve oplevering van het product.

Doelstellingen transition phase

De doelstellingen van transition phase zijn:

- het testen van het systeem
- het implementeren van het systeem
- het in gebruik nemen van het systeem

9.1 TESTEN

Het testen hebben hoofdzakelijk ik en de opdrachtgever gedaan. Dit hebben we gedaan aan de hand van de use case templates. Hierin stond vastgelegd wat het systeem moest doen en wat de gebruiker kon verwachten. Als dit niet klopte noteerde ik of de opdrachtgever dit.

Voor het testen heb ik eigenlijk een nieuw RUP-artefact geïntroduceerd. Namelijk het bugdocument. Deze is ongeveer hetzelfde als het eisendocument, alleen in plaats van eisen worden er bugs in genoteerd. Gelukkig is het geen groot document geworden. Tijdens het testen zijn er ook kleine extra eisen bijgekomen. Deze heb ik gewoon toegevoegd aan het eisendocument. Dit zijn echter kleine eisen die in dit tijdsbestek nog geïmplementeerd konden worden. Hieronder ziet u een fragment uit dit bugdocument:

9.2 DEBUGGEN EN NIEUWE EISEN IMPLEMENTEREN

Na het testen heb ik de bugs eruit gehaald, die in het bugdocument stonden. Ook heb ik de nieuwe eisen die aan het eisendocument zijn toegevoegd geïmplementeerd.

9.3 INVOEREN NIEUWE SYSTEEM

Nadat de bugs en de nieuwe in het systeem waren geïmplementeerd heb ik het systeem ingevoerd. De opdrachtgever wist inmiddels hoe het systeem werkte en heeft dit uitgelegd aan medewerkster die over de boekingen gaat. Ik heb Clara, de assistent-manager van de lodge wegwijs gemaakt in het systeem. Het enige wat voor haar belangrijk is, is het ‘White

board’, aangezien zij geen boeking invoert. Zij moet alleen weten welke transfers er zijn en welke gasten er in de lodge zijn.

9.4 ONDERHOUDSPLAN

Ik heb samen met de opdrachtgever een mondeling informeel onderhoudsplan opgesteld welke globaal het volgende inhoudt:

Gedurende de eerste weken van gebruik van de applicatie zullen er wellicht nog een aantal bugs naar boven komen. Ook zullen de reisagenten enige ondersteuning verlangen bij het adopteren van het nieuwe systeem. Daarom geef ik de eerste drie weken, als vraagbaak, ondersteuning bij het gebruik van het nieuwe systeem en zal ik, waar nodig, de bugs uit het systeem verwijderen. Reisagenten moeten zich na die drie weken wenden tot de opdrachtgever.

De managers kunnen ten alle tijden contact met mij opnemen voor vragen over het systeem. Dus zij zijn niet beperkt tot de eerste drie weken.

Mochten er nieuwe functionaliteiten aan het systeem toegevoegd worden, zal daar een gepaste vergoeding tegenover staan.

10. PROJECTEVALUATIE

10.1 PROCES

Ik denk dat ik tijdens het project iets te veel hooi op de vork heb genomen. Ik heb het bouwen van een boekingssysteem wat onderschat. Doordat ik dit heb gedaan, heb ik de opdrachtgever aangeboden om er ook een facturatiemodule en activiteitenmodule aan te hangen. Dit had ik eigenlijk niet moeten doen. Ik had dus beter moeten inschatten wat mijn eigen kwaliteiten waren.

Over de aanpak van RUP ben ik redelijk tevreden. Hoewel het pas de tweede keer was dat ik RUP toepaste denk ik dat ik in de toepassing vrij goed geslaagd ben. RUP is wel een erg moeilijke methode, er bestaat veel documentatie over, zodat ik af en toe door de bomen het bos niet meer kon zien.

Desalniettemin is mijn ‘eerste indruk’ over RUP een goede. Met name dat RUP use case driven is spreekt me erg aan en dat er snel producten kunnen worden opgeleverd. Echter, wil ik RUP goed beheersen heb ik nog wel een lange weg te gaan.

Het objectgeoriënteerd programmeren in PHP was voor mij ook een redelijk nieuwe ervaring. Ik had het al een aantal keren geprobeerd, nu heb ik eigenlijk echt kennis meegemaakt. Objectgeoriënteerd programmeren heeft duidelijk zijn voordelen, maar ik vond het ook erg moeilijk, ook omdat je veel problemen op verschillende manier kunt oplossen. PHP is niet echt een objectgeoriënteerde programmeertaal en dan merk je toch de beperkingen van PHP.

10.2 PRODUCT

Al met al, ben ik wel tevreden met het product, ik heb op tijd kunnen inschatten dat de facturatie- en activiteitenmodule niet meer binnen de gestelde termijn van 20 weken gerealiseerd kon worden. Daar heb ik de opdrachtgever toen ook over ingelicht en is daarmee zonder meer akkoord gegaan. Overigens heb ik de opdrachtgever aan het begin wel laten weten dat het niet honderd procent zeker is dat de facturatie- en activiteitenmodule gerealiseerd konden worden. Ik heb gezegd als ik tijd over heb en er ruimte voor is zal ik er zeker aan werken. De facturatie- en activiteitenmodule was dus al optioneel.

Een groot voordeel is wel dat ik er tijdens het ontwerp (klassediagram, databaseontwerp en –implementatie) al wel rekening mee gehouden hebt. Eigenlijk het enige wat moet gedaan worden is het bouwen van schermen en het schrijven van programmacode.

Het systeem is redelijk robuust, de betrouwbaarheid is hoog. Het systeem is, dankzij de objectgeoriënteerde aanpak uitbreidbaar en makkelijke te onderhouden. Veel code kan hergebruikt worden. Met name de code van de databaselaag is kan heel veel tijd besparen bij toekomstige projecten.

LITERATUURLIJST

Warmer, J., Kleppe, A., Praktisch UML, 2e editie, **1999**, 266, *ISBN 9043004944*

Kroll, P., Kruchten, P., The Rational Unified Process made easy, **2003**, *ISBN 0-321-16609-4 (alk. paper)*

Kruchten, P., The Rational Unified Proces: An introduction, **2003** *ISBN 0-321-19770-4*

G. Pollice, L. Augustine, C. Lowe, J. Madhur, Software Engineering for small projects, 2004 *ISBN : 0-321-19950-2*

www.php.net - Officiële PHP-website

www.phpfreakz.nl – Website met veel PHP-scripts

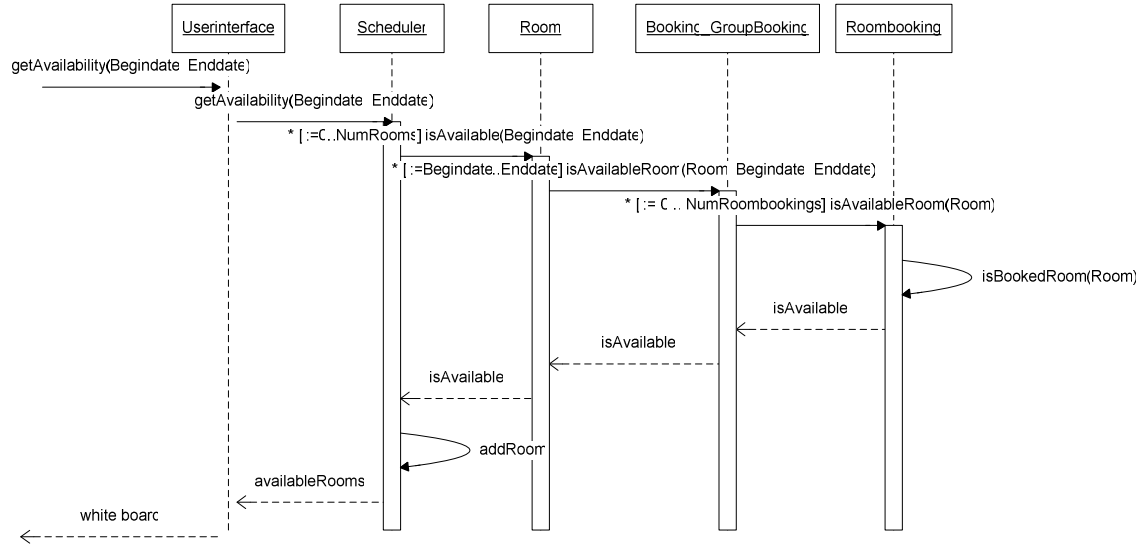
www.eprogrammeur.nl – forum voor programmeurs

BIJLAGE A – USE CASE TEMPLATES

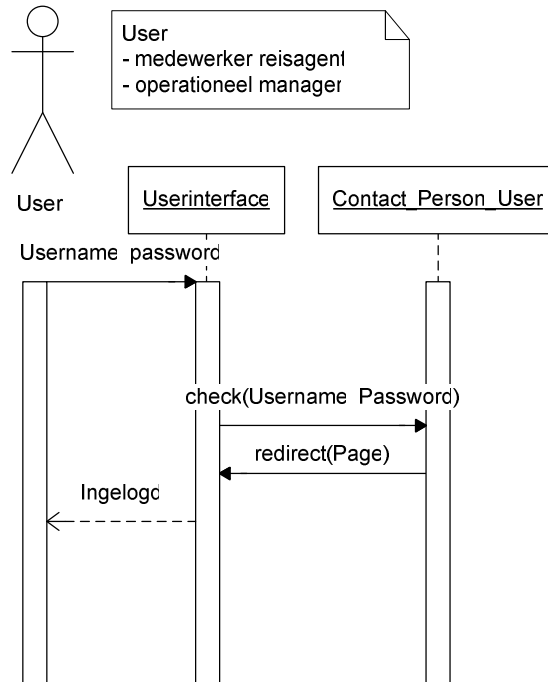
Titel: Inloggen	
Samenvatting	Inloggen op het system
Actor	Reisagent, manager
Precondities	
Postcondities	De reisagent is ingelogd.
Scenario	<ol style="list-style-type: none">1. Het systeem toont een inlogformulier5. De reisagent voert gebruikersnaam en wachtwoord in.6. Het systeem controleert de gebruikersnaam en wachtwoord.7. De reisagent heeft toegang tot het systeem. [A1]
Alternatieven	[A1] Er zijn geen juiste gegevens ingevoerd. Het systeem toont opnieuw het inlogformulier.

Titel: Boeking zoeken	
Samenvatting	De gebruiker kan een boeking zoeken op naam en boekingsdatum.
Actor	Manager
Precondities	Manager is ingelogd.
Postcondities	Het systeem toont een lijst van gevonden boekingen.
Scenario	<ol style="list-style-type: none">1. De gebruiker voert een (gedeelte van een) boekingnaam in. [A1]2. Het systeem toont een lijst met gevonden boekingen.
Alternatieven	[A1] Een boekingnaam kan vervangen worden door een begindatum en einddatum als er op boekingsdatum gezocht gaat worden.
Uitzonderingen	

BIJLAGE B – SEQUENTIEDIAGRAMMEN



Sequentie diagram use case 'White board'



Sequentiediagram use case 'Inloggen'

BIJLAGE C – VOORLOPIGE PLANNING

IC	Task Name	Stan	Finish	Duration	Nov 2004		Dec 2004					Jan 2005					Feb 2005				
					11/2	11/28	12/5	12/12	12/19	12/26	1/2	1/9	1/16	1/23	1/30	2/6	2/13	2/20	2/27		
1	Initiele fase	11/15/2004	11/26/2004	10d																	
2	Interviewen opdrachtgever	11/15/2004	11/15/2004	1d																	
3	Opstellen plan van aanpak	11/16/2004	11/18/2004	3d																	
4	Opstellen globale eisen	11/19/2004	11/22/2004	2d																	
5	Formuleren reikwijdte van het project	11/23/2004	11/23/2004	1d																	
6	Opstellen essentiële use cases	11/24/2004	11/25/2004	2d																	
7	Samenstellen architectuur	11/26/2004	11/26/2004	1d																	
8	Uitwerkingsfase	11/29/2004	12/15/2004	13d																	
9	Specificeren van de eiser	11/29/2004	11/29/2004	1d																	
10	Opstellen basisarchitectuur	11/30/2004	12/2/2004	3d																	
11	Opstellen databaseontwerp	12/3/2004	12/6/2004	2d																	
12	Opstellen gedetailleerde planning	12/7/2004	12/8/2004	2d																	
13	Aanleggen systeemdocumentatie	12/9/2004	12/10/2004	2d																	
14	Voorbereiding constructiefase	12/13/2004	12/15/2004	3d																	
15	Constructiefase	12/16/2004	2/1/2005	34d																	
16	Implementeren schermer	12/16/2004	12/2/2004	4d																	
17	Implementeren database	12/22/2004	12/23/2004	2d																	
18	Opstellen overige use cases	12/24/2004	12/28/2004	3d																	
19	Implementeren programmacode	12/29/2004	1/25/2005	20d																	
20	Aanleggen systeemdocumentatie	1/26/2005	1/27/2005	2d																	
21	Voorbereiden transitiefase	1/28/2005	2/1/2005	3d																	
22	Transitiefase	2/2/2005	2/18/2005	13d																	
23	Uitvoeren acceptatietest	2/2/2005	2/3/2005	2d																	
24	Voorbereiden oplevering	2/4/2005	2/8/2005	3d																	
25	Opstellen onderhoudspln	2/9/2005	2/11/2005	3d																	
26	Opleveren applicatie	2/14/2005	2/14/2005	1d																	
27	Opleveren systeemdocumentatie	2/15/2005	2/15/2005	1d																	
28	Opstellen gebruikershandleiding	2/16/2005	2/17/2005	2d																	
29	Opleveren gebruikershandleiding	2/18/2005	2/18/2005	1d																	
30	Uitloop	2/2/2005	3/4/2005	10d																	

BIJLAGE D – KANDIDAATKLASSELIJST

Kandidaatklasse	Beslissing
Reisagent	Is klasse
Manager	Attribuut (IsManager)
System	Implementatie
Gebruikersnaam	Attribuut
Wachtwoord	Attribuut
Toegang	Te vaag
Overzicht	Te vaag
Gast	Is klasse
Registratie	Te vaag
Lijst	Te vaag
Boeking	Ja
Wijziging	Te vaag
Controle	Te vaag
Tijdslimiet	Attribuut
Melding	Implementatie
Onderdeel	Implementatie
Boekingsoverzicht	Associatie
Bezettingsgraad	Associatie
Kamer	Is klasse
Defragmentatie	Irrelevant
Boekingsitem	Is klasse
Kamernummer	Attribuut
Data	Te vaag
Formulier	Implementatie
Datum	Attribuut
Factuur	Is klasse
Klasse na brainstormsessie met oog op uitbreiding.	
Activiteit	Is klasse
Product	Is klasse
Agenda	Te vaag
Entrypoint	Is klasse
Rate	Is klasse
Roomtype	Is klasse
Vendor	Is klasse
Contact	Is klasse



Stanley Safari Lodge

Victoria Falls, Zambia

EXTERNE BIJLAGEN

door

M.W. Henneman, 99005151

Ter afronding van de studie
Informatica en Informatiekunde
IVIT aan de

Haagse Hogeschool te 's
Gravenhage

24 maart 2005

Opdrachtgever: R. De Gruijter, Stanley Safari Lodge

Afstudeeropdracht: Sobsys, Stanley's Online Bookingssystem

INHOUDSOPGAVE

1. INLEIDING	- 4 -
2. OPDRACHTSOMSCHRIJVING.....	- 5 -
2.1 Omschrijving afstudeeropdracht	- 5 -
2.1.1 Inleiding (<i>organisatorische omgeving, kader, historie</i>).....	- 5 -
2.1.2 Probleemstelling.....	- 5 -
2.2 Doelstelling van de opdracht.....	- 6 -
2.3 Uitgangssituatie.....	- 6 -
2.3.1 Benodigde software.....	- 6 -
2.3.2 Benodigde hardware.....	- 6 -
2.3.3 Beschikbare rapporten	- 6 -
2.3.4 Aanwezige ideeën	- 7 -
2.4 Concrete werkzaamheden	- 7 -
2.4.1 Uit te voeren activiteiten.....	- 7 -
2.4.2 Te hanteren methodieken	- 8 -
2.4.3 Te gebruiken technieken	- 8 -
2.4.4 Planning.....	- 8 -
2.5 Te vermelden nadrukken	- 10 -
2.6 Resultaten voor de opdrachtgever	- 10 -
2.7 Relatie blok 6 en 7	- 10 -
3. VISIEDOCUMENT	- 11 -
3.1 Voordelen en mogelijkheden.....	- 11 -
3.2 De problemen die opgelost zullen worden	- 11 -
3.3 Betrokkenen.....	- 12 -
3.4 Voornaamste use cases.....	- 12 -
3.5 Voornaamste niet-functionele eisen	- 12 -
4. PROJECTPLAN.....	- 13 -
4.1 Projectomgeving.....	- 13 -
4.2 Kosten.....	- 14 -
4.3 Winstgevendheid	- 14 -
4.4 Kwaliteitsborging	- 14 -
<i>Specifiek</i>	- 14 -
<i>Meetbaar</i>	- 14 -
<i>Aanwijsbaar</i>	- 14 -
<i>Realistisch</i>	- 15 -
<i>Tijdgerelateerd</i>	- 15 -
4.5 Organisatie	- 15 -
<i>M.W. Henneman: afstudeerkandidaat</i>	- 15 -
<i>R. de Gruijter: bedrijfsmentor en opdrachtgever</i>	- 15 -
<i>MilesTravel: reisagent</i>	- 15 -
<i>Clara: operationeel manager</i>	- 16 -
4.6 Projectdoelstelling.....	- 16 -
4.7 Te gebruiken methoden en technieken.....	- 16 -
4.8 Risicomanagement	- 16 -
4.9 Cruciale Succesfactoren.....	- 17 -

4.10 Eisen en beperkingen	- 17 -
4.11 Faciliteiten	- 18 -
<i>Hardware</i>	- 18 -
<i>Software</i>	- 18 -
<i>Internet</i>	- 18 -
4.12 Informatie.....	- 18 -
4.13 Op te leveren producten	- 18 -
4.14 Activiteitenplan.....	- 19 -
4.15 Tijdsplanning	- 20 -
5. EISENDOCUMENT	- 21 -
6. RISICODOCUMENT	- 23 -

1. INLEIDING

Dit is het artefactenrapport gemaakt door Mark Henneman. De artefacten in dit rapport hebben betrekking op het afstudeerproject SOBSYS, Stanley's Online Bookingsystem in opdracht van Stanley Safari Lodge in Livingstone, Zambia.

2. OPDRACHTSOMSCHRIJVING

2.1 OMSCHRIJVING AFSTUDEEROPDRACHT

2.1.1 Inleiding (*organisatorische omgeving, kader, historie*)

Stanley Safari Lodge is een exclusieve kleinschalige lodge met slechts 7 kamers/suites, enkele kilometers van de Victoria Falls in Zambia. De gasten komen hoofdzakelijk uit Europa en de Verenigde Staten.

Het merendeel van de gasten verblijft enkele dagen aldaar en combineert de Victoria Falls met een safari in het Botswaanse Okavango Delta, het Zambiaanse South Luangwa National Park of een bezoek aan Zuid-Afrika. De gasten hoeven het verblijf zelf niet te regelen. Dat regelen de reisagentschappen en touroperators via veelal e-mail, de telefoon en per fax met de manager van de lodge.

2.1.2 Probleemstelling

De volgende problemen doen zich voor bij het huidige boekingsproces:

- Reisagenten kunnen niet direct de beschikbaarheid van de kamers controleren.
- In Zambia komt het voor dat de stroom of het Internet niet werkt, waardoor er nauwelijks communicatie mogelijk is met de buitenwereld. Aangezien de boekingen gescheiden via e-mail, fax en telefoon kan het voorkomen dat het enkele dagen kan duren voordat een reisagent de beschikbaarheid van een kamer weet.
- De opdrachtgever wil, ook als hij niet op de lodge verblijft, toegang hebben tot het boekingssysteem. Tevens kan hij geen goede overzichten maken op de huidige manier van werken.

2.2 DOELSTELLING VAN DE OPDRACHT

De doelstelling van de opdracht is:

- Het ontwikkelen van een informatiesysteem dat "any place, any time" beschikbaar is, met de volgende functionaliteiten:
 - het uitvoeren van een beschikbaarheidscontrole;
 - het boeken van een kamer;
 - het opvragen van de status van de boeking;
 - het genereren van overzichten;
- het aanleggen van bijbehorende systeemdocumentatie
- het opstellen van bijbehorende gebruikershandleiding

2.3 UITGANGSSITUATIE

2.3.1 Benodigde software

- Microsoft Windows XP
- Microsoft Office 2003 of 2000
- Macromedia Homesite (voor het gebruik bij het coderen in HTML/PHP etc.)
- Macromedia Dreamweaver (idem)
- PHPMyAdmin (webbased applicatie om MySQL-databases te beheren)
- PHPDocumenter (webbased applicatie om perfecte systeemdocumentatie te maken)
- Apache Webserver (voor het testen van het systeem)
- MySQL (DBMS)
- PHP 4.x

2.3.2 Benodigde hardware

- Een PC werkstation die doorgaans gebruikt wordt bij de ontwikkeling van een internetapplicatie. Dit kan een notebook of een desktop zijn.
- Een afdrukapparaat (printer) voor het afdrukken van modellen, verslagen etc.

2.3.3 Beschikbare rapporten

- Excel-sheets die de opdrachtgever gebruikt ter ondersteuning van het huidige boekingsproces.

2.3.4 Aanwezige ideeën

De opdrachtgever had zelf het idee er een online boekingssysteem van te maken door gebruik te maken van open source tools.

2.4 CONCRETE WERKZAAMHEDEN

2.4.1 Uit te voeren activiteiten

Initiele fase

- Interviewen opdrachtgever
- Opstellen plan van aanpak
- Opstellen globale eisen (oa. prototype GUI)
- Formuleren van de reikwijdte van het project
- Opstellen van essentiële use-cases
- Samenstellen architectuur

Uitwerkingsfase

- Specificeren van de eisen
- Opstellen van de basisarchitectuur
- Opstellen databaseontwerp
- Opstellen van gedetailleerdere planning
- Aanleggen systeemdokumentatie
- Voorbereiding op constructiefase (benodigde software installeren etc.)

Constructiefase

- Implementeren schermen
- Implementeren database
- Opstellen van overige use-cases en eisen
- Implementeren programmacode
- Aanleggen systeemdokumentatie

- Voorbereiden overgangsfase

Overgangsfase

- Uitvoeren acceptatietest
- Voorbereiden van oplevering van producten

- Regelen onderhoud en ondersteuning
- Opleveren applicatie
- Opleveren systeemdokumentatie
- Opleveren gebruikershandleiding

2.4.2 *Te hanteren methodieken*

Tijdens het project zal de ontwikkelmethode Rational Unified Proces (RUP) gebruikt worden.

2.4.3 *Te gebruiken technieken*

- Unified Modelling Language
- ERD-schematechniek
- Interviewtechniek
- Gantt-diagram

2.4.4 *Planning*

Activiteit	Aantal dagen
Initiele fase	
Interviewen opdrachtgever	1
Opstellen plan van aanpak	3
Opstellen globale eisen	2
Formuleren van de reikwijdte van het project	1
Opstellen van essentiële use-cases	2
Samenstellen architectuur	1
	10
Uitwerkingsfase	
Specificeren van de eisen	1
Opstellen van de basisarchitectuur	3
Opstellen databaseontwerp	2
Opstellen gedetailleerdere planning	2
Aanleggen systeemdokumentatie	2
Vorbereiding op constructiefase	3

	13
Constructiefase	
Implementeren schermen	4
Implementeren database	2
Opstellen overige use-cases en eisen	3
Implementeren programmacode	20
Aanleggen systeemdocumentatie	2
Voorbereiden overgangsfase	3
	34
Overgangsfase	
Uitvoeren acceptatietest	2
Voorbereiden van oplevering van de producten	3
Regelen onderhoud en ondersteuning	3
Opleveren applicatie	1
Opleveren systeemdocumentatie	1
Maken gebruikershandleiding	2
Opleveren gebruikershandleiding	1
	13
Subtotaal	70
Uitloop	5
Totaal	75

2.5 TE VERMELDEN NADRUKKEN

De nadruk ligt bij het deel dat voor de reisagenten zichtbaar is. Deze moet gebruikersvriendelijk zijn en perfect aan de functionele eisen voldoen. Uiteindelijk is dit hetgeen waar het om draait.

2.6 RESULTATEN VOOR DE OPDRACHTGEVER

- Plan van aanpak
- Eisenlijst
- Visie document
- Prototype GUI
- Online boekingssysteem
- Systeemdokumentatie
 - ERD-diagram
 - Klassediagram
 - Use-case-diagram
 - Use-cases
 - Toestandsdiagrammen
- Gebruikershandleiding
- Onderhoudsplan

2.7 RELATIE BLOK 6 EN 7

Bij het voorafgaande Integrerend Practicum (IP-03) heeft de afstudeerkandidaat ervaring opgedaan met RUP. MySQL en PHP werden al beheerst, maar het programmeren heeft toch een bijdrage geleverd voor de kennis van het objectgeoriënteerde programmeren. Objectgeoriënteerd programmeren werd ook aangeleerd in het vak SW-10. UML werd aangeleerd in het vak SO-08. Verder heeft de student met verschillende softwareontwikkeling- en procesbeheersingmethode kennisgemaakt tijdens het vak SO-07, waardoor RUP gemakkelijker te begrijpen en te hanteren is. Daarnaast zal de kennis die opgedaan is tijdens DB-01, DB-02, DB-03 en DB-05 erg goed van pas komen.

3. VISIEDOCUMENT

Een visiedocument wordt gemaakt om op globaal vast te leggen wat het systeem moet gaan doen, wat de voordelen ervan zijn en welke mensen bij het project betrokken zijn. Het doel is om ervoor te zorgen dat alle projectdeelnemers dezelfde visie hebben en “met de neus dezelfde richting in staan”.

3.1 VOORDELEN EN MOGELIJKHEDEN

De voordelen en mogelijkheden die het nieuwe systeem biedt zijn:

- Het boekingsproces verloopt sneller en efficiënter
- Minder kans op fouten bij het maken van een boeking
- De manager heeft meer zijn handen vrij voor andere werkzaamheden
- Er kunnen betere overzichten gegenereerd worden
- Er kan “any place, any time” toegang verkregen tot het boekingssysteem

3.2 DE PROBLEMEN DIE OPGELOST ZULLEN WORDEN

De volgende problemen zal door het te bouwen systeem opgelost moeten worden.

- Reisagenten kunnen niet direct de beschikbaarheid van de kamers controleren.
In Zambia komt het voor dat de stroom of het Internet niet werkt, waardoor er nauwelijks communicatie mogelijk is met de buitenwereld. Aangezien de boekingen geschieden via e-mail, fax en telefoon kan het voorkomen dat het enkele dagen kan duren voordat een reisagent de beschikbaarheid van een kamer weet.
- Zoals al aangegeven verlopen de boekingen via onder andere e-mail. Doordat een reisagent weleens vergeet om de juiste informatie in de e-mail te geven kan het wel eens lang duren voordat alle benodigde informatie (paspoortgegevens, aantal personen etc.) bij Stanley Safari aanwezig is. Ook komt het wel eens voor dat een boeking meerder malen wordt aangepast. Het continu aanpassen van boekingen kost de lodgemanager veel tijd, welke hij graag in andere werkzaamheden zou willen steken. Ook maakt de lodgemanager (bij het continu aanpassen van een boeking) soms fouten. Dit kan tot ontevredenheid leiden van de reisagenten.

- De opdrachtgever wil, ook als hij niet op de lodge verblijft, toegang hebben tot het boekingssysteem. Tevens kan hij geen goede overzichten maken op de huidige manier van werken.

3.3 BETROKKENEN

De betrokkenen bij dit project zijn:

- Operationeel lodgemanager (R. De Gruijter of iemand anders)
- Reisagent (Miles Travel, te Den Bosch)
- Productontwikkelaar (M.W. Henneman)
- Begeleiders vanuit school (A. Van der Molen, P.B. van der Sluys)

3.4 VOORNAAMSTE USE CASES

De voornaamste use cases zijn:

- Boeking
- Inloggen
- Wijzigen boekingsstatus
- Wijzigen kamerverdeling (defragmentatie)
- Overzichten (boekingsoverzicht per kamer per bepaalde periode, boekingsoverzicht per kamer per bepaalde periode met daarin duidelijk aangegeven wat de status van de boeking is.)

3.5 VOORNAAMSTE NIET-FUNCTIONELE EISEN

De voornaamste niet-functionele eisen zijn:

- De applicatie zal gebruik maken van een MySQL-database
- De applicatie zal geschreven worden in PHP, HTML en eventueel Javascript en CSS.

Bovenstaande zijn open-source tools. Er zijn geen kosten verbonden aan het gebruik van deze gereedschappen.

4. PROJECTPLAN

Het doel van dit projectplan is het plannen en voorbereiden van een zakelijke rechtvaardiging. Risicomanagement, organisatie, en overwegingen voor kosten/tijdsplanning/winstgevendheid worden hierin beschreven. De infrastructuur (faciliteiten, methoden en technieken), die voldoende is voor de ondersteuning van de levenscyclus ontwikkelingstaken, wordt bepaald.

4.1 PROJECTOMGEVING

Stanley Safari Lodge is een exclusieve kleinschalige lodge met slechts 7 kamers/suites, enkele kilometers van de Victoria Falls in Zambia. De gasten komen hoofdzakelijk uit Europa en de Verenigde Staten.

Het merendeel van de gasten verblijft enkele dagen aldaar en combineert de Victoria Falls met een safari in het Botswaanse Okavango Delta, het Zambiaanse South Luangwa National Park of een bezoek aan Zuid-Afrika. De gasten hoeven het verblijf zelf niet te regelen. Dat regelen de reisagentschappen en touroperators via veelal e-mail, de telefoon en per fax met de manager van de lodge.

Het project zal in drie verschillende landen worden uitgevoerd; Nederland, België en Zambia. De eerste maand zal er in Brussel, België gewerkt worden. Er wordt daar een appartement door de opdrachtgever ter beschikking gesteld. In dit appartementencomplex zal gedurende die maand ook werkruimte gereed worden gemaakt.

Na een maand gaat de werknemer even terug naar Nederland daar nog wat te regelen voor het vertrek naar Zambia. Vanuit Brussel gaan we naar Zambia en wordt er op locatie gewerkt. Er wordt op de lodge een kamer ter beschikking gesteld waar de afstudeerkandidaat kan overnachten. Daarnaast is er op de lodge ook een kantoor aanwezig.

4.2 KOSTEN

De kosten voor het project zijn als volgt opgebouwd. De kosten zijn een schatting en afgerond op hele bedragen.

Omschrijving	Kosten (in euro's)
Reiskosten	1100
Eten & drinken	500
Ziektekosten	200
Hardware/software	1500
Verzekeringen	200
Totaal	3500

4.3 WINSTGEVENDHEID

De heer De Gruijter heeft aangegeven dat hij zes halve dagen per week besteed aan het factureren, boeken (grootste gedeelte), opstellen van manifesten. Met het oog op het nieuwe boekingssysteem zal geschat worden dat dit één uur. Dit levert de manager een tijdswinst op van ongeveer twee uur per dag. Dit betekent dat dat per week een tijdswinst oplevert van twaalf uur. De manager kan zich zo meer bezig houden met de core-business en andere belangrijke werkzaamheden.

4.4 KWALITEITSBORGING

De kwaliteit wordt middels SMART-doelen geformuleerd:

Specifiek

Het wordt een boekingssysteem ontwikkeld dat minimaal aan de belangrijkste eisen voldoet die in paragraaf “Eisen en beperkingen” worden opgesomd.

Meetbaar

Er is een boekingssysteem ontwikkeld die aan de gestelde eisen voldoet. De gebruikers kunnen zelfstandig met het systeem werken.

Aanwijsbaar

M.W. Henneman zal de ontwikkeling van het boekingssysteem op zich nemen. Daarnaast zal de heer R. De Gruijter ook enige inbreng hebben.

Realistisch

Het boekingssysteem bevat minimaal de gestelde eisen als in “Eisen en beperkingen”. Daarnaast zal zoveel mogelijk aan de andere eisen voldaan worden.

Tijdgerelateerd

Het project begint 15 november 2004 en zal uiterlijk tot 4 maart 2005 duren.

4.5 ORGANISATIE

De volgende personen/instanties zijn bij het project betrokken:

M.W. Henneman: afstudeerkandidaat

De belangrijkste persoon van het project en vervult de rollen analyst, systeemontwikkelaar en programmeur.

Stroomerlaan 2

1861 TJ Bergen NH

Nederland

T. +31654700066 (tijdens verblijf in Zambia, zie contactgegevens R. De Gruijter)

E. mark@bacoba.nl

R. de Gruijter: bedrijfsmentor en opdrachtgever

De bedrijfsmentor, tevens opdrachtgever en eigenaar/manager van Stanley Safari Lodge. Ook zal hij de rol vervullen als gebruiker en tester.

Rue du Bourgemestre 44

1050 Bruxelles

Belgie

T. +32496672040

Stanley Safari Lodge

P.O. Box 60439

Livingstone, Zambia

T. +26097848615(Botswana)/+26391262042(Zambia)

E. info@stanleysafaris.com

MilesTravel: reisagent

Reisagent dat reizen boekt naar Zuidelijk Afrika. Medewerkers zijn de onder andere de eindgebruikers van het systeem. Zij zullen aan het begin van het traject

geïnterviewd worden om eisen in kaart te brengen. Tijdens en tegen het einde van het project zullen zij het systeem testen.

MilesTravel
Hinthamerstraat 100
5211 MS Den Bosch

Clara: operationeel manager

Zij zal net als R. De Gruijter haar eisen kenbaar maken en als één van de testers fungeren. Voor contactgegevens zie de contactgegevens van de lodge.

4.6 PROJECTDOELSTELLING

De doelstelling van het project is het ontwikkelen van een informatiesysteem dat "any place, any time" beschikbaar is, met de volgende functionaliteiten:

- het uitvoeren van een beschikbaarheidscontrole;
- het boeken van een kamer;
- het opvragen van de status van de boeking;
- het genereren van overzichten;

En:

- het aanleggen van bijbehorende systeemdocumentatie
- het opstellen van bijbehorende gebruikershandleiding

4.7 TE GEBRUIKEN METHODEN EN TECHNIEKEN

Tijdens het project zal de ontwikkelmethode Rational Unified Proces (RUP) gebruikt worden. De volgende technieken zullen worden toegepast:

- Unified Modelling Language
- Datamodellering door middel van Visio
- Interviewtechniek
- Gantt-diagram

4.8 RISICOMANAGEMENT

De risico's zullen beheerst worden door middel van het bijhouden van een risicodocument. Per risico worden de volgende punten bijgehouden:

- De omschrijving van het risico
- De actie die nodig is om het risico zo klein mogelijk te houden (tegenactie)

- Een zwaarteindex (1-5) om aan te geven hoe groot de kans van het risico is
- De datum wanneer het risico is toegevoegd

4.9 CRUCIALE SUCCESFACTOREN

Cruciale succesfactoren geven aan welke factoren essentieel zijn voor het succesvol afronden van het project. De volgende factoren dragen bij aan het succes van het project:

- Opdrachtsomschrijving; vooraf moet duidelijk zijn wat we gaan doen
- Visie; iedereen moet de juiste visie en doel voor ogen hebben
- Plan van aanpak;
- Risicomanagent; de risico's moeten zo klein mogelijk gehouden worden.
- Eisenmanagent; de eisen waaraan het systeem moet voldoen moeten zo goed mogelijk in kaart worden gebracht. Betrokkenheid van de gebruikers is hier een onderdeel van.
- Architectuur; de architectuur dient als basis voor het systeem. Deze moet zo goed mogelijk opgezet worden om problemen in de toekomst te vermijden

4.10 EISEN EN BEPERKINGEN

Het systeem moet de volgende onderdelen bevatten:

- Het aanmaken van nieuwe boekingen door reisagenten
- Het wijzigen van reeds aangemaakte boekingen
- Diverse operationele overzichten voor de manager
- Authenticatie van de gebruiker middels gebruikersnaam en wachtwoord

Van minder belang zijn de volgende eisen:

- Het genereren van facturen
- Het genereren van manifesten (aanvraag van Zambiaans visum van gasten)
- Het wijzigen van de bezettingsgraad
- Het boeken van activiteiten
- Kameronderhoud (toevoegen/wijzigen)
- Onderhoud reisagenten (toevoegen/wijzigen)

Verder zal de applicatie in PHP moeten geschreven worden. De database moet een MySQL-database zijn.

4.11 FACILITEITEN

Voor het project worden de volgende faciliteiten gebruikt:

Hardware

- Laptop – Toshiba Satellite Pro 4600
- HP Printer
- HP Scanner

Software

- Besturingssysteem: Windows XP
- Server: database(management), PHP-parser: Appserv 2.4.1
 - phpMyAdmin databasemanager version 2.6.0-rc1
 - PHP Information version 4.3.8
 - Perl Version 5.8.4 (niet van belang bij dit project)
 - Apache Webserver version 1.3.31
 - PHP Script Language version 4.3.8
 - MySQL database version 4.0.20
 - Zend Optimizer version 2.5.3 (optimaliseert PHP-code)
- Planning: MS Project, MS Word
- Modellering: MS Visio, DB Designer
- Implementatie: Macromedia Dreamweaver MX, Homesite 5.0, Topstyle
- Documentatie: MS Word
- Eisen-, versie- en risicomanagement: MS Word

Internet

Op alle locaties zal Internet beschikbaar zijn.

4.12 INFORMATIE

Informatie zal vergaard worden via:

- Vergaderingen/discussies
- Interviews
- Internet en andere bibliografische hulpmiddelen

4.13 OP TE LEVEREN PRODUCTEN

De op te leveren producten (in RUP artefacten) zijn:

- Projectplan
- Visiedocument
- Eisendocument
- Prototype Graphical User Interface
- Online boekingssysteem
- Systeemdokumentatie (databasemodel, klassediagram, use case model, use cases, toestandsdiagrammen, collaboratiediagrammen)
- Gebruikershandleiding
- Onderhoudsplan

4.14 ACTIVITEITENPLAN

Hieronder in tabelvorm de activiteiten met daarnaast het geschatte aantal dagen dat daaraan beesteed gaat worden.

Activiteit	Aantal dagen
Initiele fase	10
Interviewen opdrachtgever	1
Opstellen visiedocument	2
Opstellen plan van aanpak	3
Opstellen eisenlijst	1
Opstellen essentiële use cases	2
Samenstellen architectuur	1
Uitwerkingsfase	13
Specificeren eisen	1
Opstellen basisarchitectuur	3
Opstellen databaseontwerp	2
Opstellen gedetailleerde planning	2
Aanleggen systeemdokumentatie	2
Voorbereiding constructiefase	3
Constructiefase	34
Implementeren schermen	4
Implementeren database	2
Opstellen overige use cases	3
Implementeren programmacode	20
Aanleggen systeemdokumentatie	2
Voorbereiden transitiefase	3
Transitiefase	13
Uitvoeren acceptatietest	2

Voorbereiden van oplevering	3
Opstellen onderhoudsplan	3
Opleveren applicatie	1
Opleveren systeemdokumentatie	1
Maken gebruikershandleiding	2
Opleveren gebruikershandleiding	1
Uitloop	5

4.15 TIJDSPLANNING

Aan de hand van de vorige paragraaf (activiteitenplan) heb ik een GANTT-diagram gemaakt. Hieronder een schematisch overzicht van de planning.

ID	Task Name	Start	Finish	Duration	Nov 2004	Dec 2004				Jan 2005				Feb 2005			
						1/21	1/28	12/5	12/12	12/19	12/26	1/2	1/9	1/16	1/23	1/30	2/6
1	Initiele fase	11/15/2004	11/26/2004	10d													
2	Interviewen opdrachtgever	11/15/2004	11/15/2004	1c													
3	Opstellen plan van aanpak	11/16/2004	11/16/2004	1c													
4	Opstellen globale eisen	11/15/2004	11/22/2004	2c													
5	Formuleren reikwijdte van het project	11/23/2004	11/23/2004	1c													
6	Opstellen essentiële use cases	11/24/2004	11/25/2004	2c													
7	Samenstellen architectuur	11/26/2004	11/26/2004	1c													
8	Uitwerkingsfase	11/29/2004	12/15/2004	13d													
9	Specificeren van de eisen	11/29/2004	11/29/2004	1c													
10	Opstellen basisarchitectuur	11/30/2004	12/2/2004	3c													
11	Opstellen databaseontwerp	12/3/2004	12/6/2004	2c													
12	Opstellen gedetailleerde planning	12/7/2004	12/9/2004	2c													
13	Aanleggen systeemdokumentatie	12/9/2004	12/10/2004	2c													
14	Voorbereiding constructiefase	12/13/2004	12/15/2004	3c													
15	Constructiefase	12/16/2004	2/1/2005	34d													
16	Implementeren scherm	12/16/2004	12/21/2004	4c													
17	Implementeren database	12/22/2004	12/23/2004	2c													
18	Opstellen overige use cases	12/24/2004	12/28/2004	3c													
19	Implementeren programmacode	12/29/2004	1/25/2005	20c													
20	Aanleggen systeemdokumentatie	1/26/2005	1/27/2005	2c													
21	Voorbereiden transitiefase	1/28/2005	2/1/2005	3c													
22	Transitiefase	2/2/2005	2/18/2005	13d													
23	Uitvoeren acceptatietest	2/2/2005	2/3/2005	2c													
24	Voorbereiden oplevering	2/4/2005	2/6/2005	3c													
25	Opstellen onderhoudsplan	2/6/2005	2/11/2005	3c													
26	Opleveren applicatie	2/14/2005	2/14/2005	1c													
27	Opleveren systeemdokumentatie	2/15/2005	2/15/2005	1c													
28	Opstellen gebruikershandleiding	2/16/2005	2/17/2005	2c													
29	Opleveren gebruikershandleiding	2/18/2005	2/18/2005	1c													
30	Uitloop	2/2/2005	3/4/2005	10c													

5. EISENDOCUMENT

Omschrijving	Prior.	Mo.	Status	Datum
De tijdslimiet voor boekingen te wijzigen < 10 dagen	M	2	OK	15/11
Er moet een helpfunctie beschikbaar zijn.	W	1		15/11
Reisagenten moeten online een beschikbaarheid van de kamers controleren.	M	5	OK	15/11
Bij een boeking moet gelijk aangegeven kunnen worden welke activiteiten men wil gaan doen.	W	5		18/11
Een reisagent krijgt per boeking een bepaald percentage commissie.	M	1	OK	15/11
De reisagent bepaalt zelf hoeveel kamers er nodig zijn, dat doet niet het systeem.	M	1	OK	15/11
De prijzen zijn per persoon per kamer en afhankelijk van het aantal mensen per kamer en van het seizoen.	M	2	OK	15/11
Tijdens kerst en nieuwjaar worden hoogseizoenprijzen gerekend.	M	1	OK	15/11
Als er slechts één persoon in een kamer verblijft worden daar extra kosten voor berekend. (single supplement)	M	3	OK	15/11
Wanneer kinderen een kamer delen met twee ouders krijgen de kinderen 50% korting.	C	4		15/11
Transfers van/naar Livingstone Airport zijn inbegrepen. Behalve als er slechts één overnachting is.	C	4		15/11
Voor een transfer geldt dat er een uitvoerder en een verantwoordelijke is.	M	3	OK	15/11
Één gebruikersnaam en wachtwoord per medewerker per reisbureau	M	3	OK	15/11
De manager moet een boekingoverzicht kunnen laten genereren.	M	2	OK	15/11
De manager moet aan kunnen geven dat de reisagent wel of geen leuke gasten aanlevert.	C	1	OK	15/11
De manager moet de status van een boeking kunnen wijzigen.	M	2	OK	15/11
De manager moet manifesten kunnen laten genereren.	W	3		15/11
De manager moet gasten kunnen verplaatsten van de ene kamer naar een andere kamer. (verhogen van de bezettingsgraad)	M	5	OK	17/11
Er moeten kamers toegevoegd/bewerkt/verwijderd kunnen worden.	W	2		15/11
Er moet zoveel mogelijk rekening worden gehouden met de uitbreiding van het systeem.	S	5	OK	15/11
De programmacode moet voorzien zijn van duidelijk commentaar in het Engels.	S	2	OK	15/11
Het is een Engelstalige applicatie	M	2	OK	15/11
Zo min mogelijke gebruik van grote afbeeldingen.	S	1	OK	15/11

Van tevoren zijn codestandaarden opgesteld. Daar dient zoveel mogelijk aan worden gehouden.	M	2	OK	15/11
Het boekingsoverzicht per reisbureau moet boekingen weergeven van maximaal een maand geleden	S	2	OK	08/01
Een boeking dient door de manager bevestigd te worden	M	2	OK	18/11
Elke boekingsstatus in het overzicht heeft z'n eigen kleur	S	2	OK	09/01
Een reisbureau kan alleen kiezen tussen kamertypen	S	3	OK	15/11
Op het overzicht 'White board' moet per dag het aantal personen weergegeven worden	S	1	OK	05/02
Het overzicht 'White board' moet in dezelfde kleuren als het echte bord (IN rood, OUT blauw, rest zwart)	S	3	OK	05/02
Weekend moeten duidelijk zichtbaar zijn op het 'White board'	S	2	OK	05/02
Het 'White board' moet uitgeprint kunnen worden	S	1	OK	05/02
Er moet op het 'White board' aangegeven worden of Stanley verantwoordelijk is voor de transfers	S	2	OK	05/02
Boeking dient aan volgende eisen te voldoen: <i>Bookingname x numpax</i>	S	1	OK	05/02
De layout moet zo compact mogelijk zijn	S	2	OK	05/02
Vanuit het availability-scherm moet een boeking bewerkt kunnen worden (niet vanuit het 'white board')	M	3	OK	08/01
Vanuit het availability-scherm moet een kamerboeking 'gedefragmenteerd' kunnen worden.	M	5	OK	08/01
Vanuit het availability-scherm moet een e-mail naar de reisagent gestuurd kunnen worden (wanneer bijv. Als de status te lang op Provisional staat)	W	2	OK	05/02

6. RISICODOCUMENT

Te weinig ervaring met of kennis van RUP		Gemiddeld	23/11
	Niet weten wat je in één van de fases moet doen.		
	Boeken gekocht en gedownload over RUP. Voor het project goed inlezen en voor mezelf noteren wat de belangrijkste activiteiten en artefacten per fase zijn.		

Een té groots systeem voor ogen hebben		Hoog	23/11
	Steeds maar weer nieuwe functies toevoegen en door blijven denken over nieuwe functionaliteiten.		
	Duidelijke het doel van het systeem afbakenen. Goed met de opdrachtgever overleggen wat per se nodig is. Goed het eisendocument gebruiken en MoSCoW gebruiken.		

Te veel focus op details		Hoog	23/11
	Te lang op één punt blijven hangen. Je maakt stukken minder vooruitgang dan anders. Je hebt het gevoel dat je doodloopt.		
	Vraag jezelf bij het ontwikkelen van de applicatie af. Is het echt nodig? Waarom doe ik het? Heeft de opdrachtgever er duidelijk om gevraagd?		

Flexibele werktijden		Laag	23/11
	De werktijden zijn flexibel en door mijzelf in te delen.		
	Meer een kwestie van zelfdiscipline. Zorg ervoor dat je ongeveer 8 uur per dag bezig bent met het project en 5 dagen per week. Nog belangrijker, houd de planning goed in de gaten, wanneer doe je wat en wanneer moeten bepaalde artefacten opgeleverd worden.		

Het buitenland, gebrek aan interesse		Laag	23/11
	Je verliest de aandacht voor het project en richt je meer op het 'toeristische' karakter van de stage. Gebrek aan interesse kan gebeuren als het even tegenzit.		
	Vraag je elke dag af waarom je er bent. Houdt je hoofd bij het project. Ontspan wanneer het kan. Zorg ook voor genoeg afwisseling.		

Versieconflict MySQL en PHP		Laag	23/11
	De versies van mysql en php komen niet overeen als op de server waarop de		

	applicatie getest is.
	Bij het selecteren van een hostingbedrijf controleren welke versies zij gebruiken. Dit afstemmen op de versie die gebruikt wordt bij het testen van het systeem.

Te strict aan RUP houden		Gemiddeld	23/11
	Op het moment dat je eigenlijk niet weet waarom je een artefact maakt en je het alleen maakt omdat een bepaald boek dat verteld.		
	Bedenk bij elke activiteit dat je doet, waarom je het doet en welk nut het heeft om het te doen.		

Het systeem wordt niet door de gebruiker geaccepteerd (mn. door de reisagent)		Laag	23/11
	De reisagent maakt geen gebruik van de applicatie, maar blijft boekingen doorgeven via e-mail, fax of telefoon.		
	Houd goed in de gaten wat de reisagent wilt en houdt goed contact met de reisagent, zodat ze zich betrokken voelen bij het project. Maak voortijdig bekend dat er een nieuw systeem ingevoerd gaat worden. Zorg voor een goede introductie en helpfunctie.		

Een fase te ver denken		Hoog	23/11
	Je zit bijvoorbeeld al in de initiële fase al aan een databaseontwerp te denken.		
	Besef goed in welke fase je zit. Houdt de planning nauwkeurig bij.		

Te veel nieuwe dingen proberen		Laag	23/11
	Je probeert te veel dingen te leren, waardoor je in tijdnood komt.		
	Als je iets nieuws wilt proberen, kijk dan eerst of daar ruimte en tijd voor is.		

RUP is iteratief		Hoog	23/11
	Je bent gewend om met SDM te werken en bent bezig met de fases uit de conventionele systeemontwikkeling.		
	Bibliografische hulpmiddelen goed bestuderen.		

Geen goede architectuur		Hoog	23/11
	Begin niet te snel aan de constructiefase, als je nog niet helemaal zeker bent van de architectuur.		

	Begin pas aan de constructiefase als je zeker bent van de architectuur.
--	---

RUP is geen doel, maar een hulpmiddel	Gemiddeld	23/11
	Als je teveel bezig bent met het goed uitvoeren van RUP	
	Vergeet niet dat je een systeem moet ontwikkelen.	

Te veel zelf beslissingen nemen	Hoog	23/11
	Als je zelf veel beslissing neemt over de functionaliteit van het systeem, die je zelf wel leuk/handig lijkt.	
	Duidelijk doel afbakenen en eisen formuleren van de opdrachtgever.	