

People matter, results count.

# Mobiel inspectie management

*De ontwikkeling van een mobiele applicatie op het Sybase Unwired Platform ter ondersteuning van het inspectie proces in SAP.*

Naam: Jerry van Holst

Studentnummer: 08024146

Datum: 24 september 2012

Versie: 1.0



# Referaat

## People matter, results count.

Dit verslag beschrijft het proces dat is doorlopen bij het uitvoeren van de afstudeeropdracht “Mobiël inspectie management” bij Capgemini in Utrecht.

Afstudeerder:	Jerry van Holst
Studentnummer:	08024146
Onderwijsinstelling:	Haagse Hogeschool
Opleiding:	Informatica
Duur:	06-05-2012 tot 24-09-2012
Organisatie:	Capgemini
Bedrijfsmentor:	Femke Hoekstra
Opdrachtgever:	Menno Barkmeijer
Descriptoren:	Sybase Unwired Platform Inspectie Management SAP Mobile Development JAVA



# Voorwoord

Dit afstudeerverslag is geschreven in het kader van mijn afstudeeropdracht die is uitgevoerd bij Capgemini Nederland. In dit project is een mobiele applicatie ontwikkeld die het uitvoeren van inspecties in het software pakket SAP ondersteund

Hierbij wil ik graag iedereen bedanken die mij geholpen heeft met dit project. En in het bijzonder mijn bedrijfsmentoren Femke Hoekstra en Manja Kerstholt voor het begeleiden en ondersteunen van het proces dat ik doorlopen heb. Ook wil ik Menno Barkmeijer bedanken voor de ondersteuning met het functionele gedeelte van inspectie management en SAP. Tot slot wil ik ook graag de examinatoren Alwine Lousberg-Orbons en Okan Zor bedanken voor de begeleiding vanuit school.

Jerry van Holst

Utrecht, 23 september 2012

## Inhoud

<b>1. Inleiding .....</b>	<b>5</b>
<b>2. Capgemini.....</b>	<b>6</b>
2.1 Organisatie structuur .....	7
2.2 Internationale Setting .....	8
<b>3. Opdracht omschrijving .....</b>	<b>9</b>
3.1 Probleemstelling .....	9
3.2 Aanleiding .....	9
3.3 Doelstelling .....	11
3.4 Resultaat.....	11
<b>4. Aanpak.....</b>	<b>12</b>
4.1 Onderzoek aanpak .....	12
4.2 Ontwikkelmethode .....	13
4.3 Plan van Aanpak.....	14
<b>5. Onderzoek naar Sybase Unwired Platform .....</b>	<b>16</b>
5.1 Wat is Sybase Unwired Platform? .....	17
5.2 Wat is een Mobile Enterprise Application Platform? .....	19
5.3 Wat zijn de voordelen van MEAP? .....	20
5.4 Wat zijn de nadelen van MEAP? .....	22
5.5 Hoe kun je Sybase Unwired Platform koppelen met een SAP back-end? .....	23
5.6 Hoe kun je applicaties ontwikkelen met Sybase Unwired Platform? .....	24
5.7 Welke architectuur keuzes moeten er gemaakt worden. ....	26
5.8 Conclusie .....	29
<b>6. Inspectie Management.....</b>	<b>30</b>
<b>7. SAP back-end systeem.....</b>	<b>32</b>
7.1 SAP .....	32
7.2 SAP systemen .....	32
7.3 Back-end systeem keuze.....	33
7.4 Inrichten van het back-end systeem.....	34
<b>8. Definitie van requirements .....</b>	<b>35</b>
8.1 Mobiel inspectie management .....	35
8.2 De applicatie .....	38

<b>9. Het ontwerp .....</b>	<b>39</b>
<b>10. Architectuur.....</b>	<b>41</b>
<b>11. Wijziging van de doelstelling .....</b>	<b>42</b>
11.1 Probleem .....	42
11.2 Doelstelling .....	42
11.3 Resultaat.....	42
<b>12. Ontwikkeling van de applicatie .....</b>	<b>43</b>
12.1 Gereed maken van het systeem.....	43
12.2 Ontwikkelen applicatie .....	49
<b>13. Advies .....</b>	<b>54</b>
<b>14. Evaluatie .....</b>	<b>55</b>
14.1 Proces evaluatie .....	55
14.2 Product evaluatie .....	56
<b>15. Aantonen beroepstaken .....</b>	<b>58</b>
15.1 Voorbereiden en opstarten softwareontwikkeltraject.....	58
15.2 Uitvoeren analyse door definitie van requirements .....	58
15.3 Ontwerpen software architectuur.....	58
15.4 Bouwen applicatie.....	58
<b>16. Literatuurlijst .....</b>	<b>59</b>

## Bijlagen

Bijlage A	Afstudeerplan
Bijlage B	Plan van aanpak
Bijlage C	Inspectie proces beschrijving
Bijlage D	Product Backlog
Bijlage E	Ontwerp

# 1. Inleiding

Dit verslag is gericht aan de beoordelende docenten van de Haagse Hogeschool: Alwine Lousberg-Orbons en Okan Zor. Daarnaast is dit verslag ook gericht aan de externe gecommitteerde en mijn bedrijfsbegeleider.

In februari 2012 ben ik begonnen met een afstudeeropdracht bij Capgemini. Helaas is die opdracht door omstandigheden niet goed gegaan en ben ik gestopt. Omdat ik het erg naar mij zin had bij Capgemini en zij mij graag een nieuwe afstudeerplek wilden geven heb ik besloten om in mei 2012 een nieuw afstudeer project te starten bij Capgemini. Ik heb zelf het initiatief genomen om een nieuwe opdracht te vinden en mede dankzij het netwerk dat ik in die tijd heb opgebouwd resulteerde dit in deze opdracht.

Vanaf mei 2012 heb ik gewerkt aan de ontwikkeling van een mobiele applicatie voor het ondersteunen van het inspectie proces in SAP. Hierbij heb ik gebruik gemaakt van het Sybase Unwired Platform om de koppeling met SAP te maken. Het doel van de applicatie is om het uitvoeren en registreren van inspecties sneller en efficiënter te maken.

In hoofdstuk twee wordt eerst de omgeving waarin ik de afstudeeropdracht uitgevoerd heb beschreven. Vervolgens beschrijft hoofdstuk drie het probleem, de aanleiding, de doelstelling en het beoogde resultaat van de afstudeer opdracht. In hoofdstuk vier beschrijf ik de project aanpak..

Omdat voor de opdracht het Sybase Unwired Platform is gebruikt heb ik eerst een onderzoek gedaan naar wat SUP precies is en hoe je er applicaties mee kunt ontwikkelen. Dit onderzoek en de resultaten hiervan worden beschreven in hoofdstuk vijf. In hoofdstuk zes wordt inspectie management in SAP beschreven en hoofdstuk zeven geeft een globale beschrijving van SAP en hoe ik een geschikt back-end heb gekozen voor de applicatie.

Hoofdstuk acht beschrijft hoe ik de requirements heb gedefinieerd en in hoofdstuk negen tot welk ontwerp dit geleid heeft. In hoofdstuk tien worden de gemaakte architectuur keuzes uitgelegd.

Tijdens het onderzoek en het definiëren van de requirements ben ik erachter gekomen dat de doelstelling van de opdracht gewijzigd moest worden. Deze wijziging licht ik toe in hoofdstuk elf.

Hoofdstuk twaalf beschrijft de sprints waarin ik de applicatie ontwerp en bouw. Ook beschrijf ik hier alle problemen en obstakels die ik ben tegengekomen en hoe ik deze heb opgelost. Daarnaast heb ik ook het testen van de applicatie beschreven in dit hoofdstuk.

In hoofdstuk dertien geef ik een advies naar aanleiding van de nieuwe doelstelling. In hoofdstuk veertien zal ik het proces en het product evalueren. Hierin beschrijf ik wat goed is gegaan, wat niet goed is gegaan en wat ik meeneem voor in de toekomst. Tot slot komen in hoofdstuk vijftien de beroepstaken aan bod. In mijn afstudeerplan heb ik aangegeven met deze opdracht te willen aantonen dat ik een aantal beroepstaken op het gewenste niveau heb bereikt. In dit hoofdstuk zal ik toelichten hoe ik dit bereikt heb.



## 2. Capgemini

Capgemini is een consultancy bedrijf en dienstverlener op het gebied van innovatie en technologie. De organisatie is in 1967 ontstaan toen Serge Kampf het Franse bedrijf Sogeti oprichtte en in 1975 de acquisitie van IT organisaties CAP en Gemini Computer Systems uitvoerde. De organisatie is inmiddels uitgegroeid tot één van de top vijf ICT dienstverleners wereldwijd met meer dan 115.000 werknemers in bijna 40 landen. Het hoofdkantoor bevindt zich in Parijs, met regionale kantoren in Noord, West en Zuid Europa, Noord en Zuid Amerika en het Pacifische gebied.

**Missie: Het mogelijk maken van transformatie.**

Capgemini stelt haar cliënten in staat om te transformeren en presteren door technologieën.

**Visie: Leveren van vrijheid**

Capgemini voorziet haar cliënten van inzichten en mogelijkheden die de vrijheid leveren om de hoogst mogelijke resultaten te kunnen behalen

Capgemini ontwerpt en integreert technologische oplossingen, ontwikkelt innovaties en transformeert de technische omgevingen van klanten. Deze diensten concentreren zich met name op systeemarchitectuur, -integratie en -infrastructuur. Capgemini werkt daarbij samen met strategische partners om de issues en uitdagingen van klanten slagvaardig en met alle benodigde expertises te kunnen aanpakken.

De opdracht wordt uitgevoerd binnen de divisie SAP solutions. Capgemini investeert proactief in de ontwikkeling van nieuwe dienstverlening en samen met SAP is Capgemini ontwikkelaar van nieuwe producten en methoden. Capgemini is al vijftien jaar SAP-partner en heeft meer dan 3000 SAP-projecten voor 1500 klanten uitgevoerd.

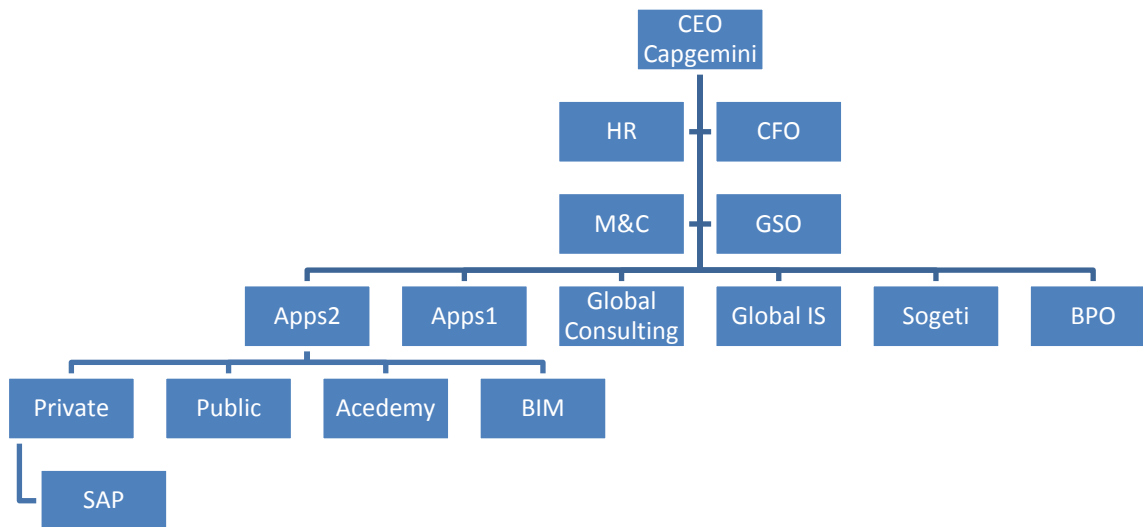


**Figuur 2.1: Hoofdkantoor Capgemini Nederland**

## 2.1 Organisatie structuur

In onderstaand overzicht is de divisiestructuur van de organisatie uitgebeeld. De organisatie telt zes verschillende divisies, deze worden binnen de organisatie ook wel Strategic Business Units (SBU) genoemd:

- Apps 2: Application services voor Overheid & Commerciële instellingen
- Apps 1: Application services voor Financiële instellingen
- Global Consulting
- Global IS: Global Infrastructure
- Sogeti, dochteronderneming
- BPO: Business Process Outsourcing



**Figuur 2.2: Positie binnen Capgemini**

De SBU's zijn alle onderverdeeld in practices, subdivisions welke een specifieke dienst of branche betreffen. Elk land heeft een eigen Countryboard (CB) welke voor alle SBU's behalve BPO, de regionale organisatie en verantwoordelijkheid (waar)neemt.

De opdracht wordt uitgevoerd binnen de SAP divisie, dit valt onder de Private sector in Apps 2, zoals te zien in het schema. De SAP divisie is opgedeeld in drie verschillende clusters:

- SAP Logistiek
- SAP Financiën
- SAP Technologie

Dit project is een combinatie van twee clusters. Het bouwen van een mobiele applicatie valt onder technologie en het inspectie management is een onderdeel van het cluster logistiek.

De gehele SAP divisie telt ongeveer 250 werknemers, ruim 200 van deze werknemers werkt dagelijks bij klanten. De overige werknemers zitten "op de bank" en zijn regelmatig aanwezig op het hoofdkantoor in Utrecht. Zij volgen trainingen of zijn op zoek naar een opdracht.



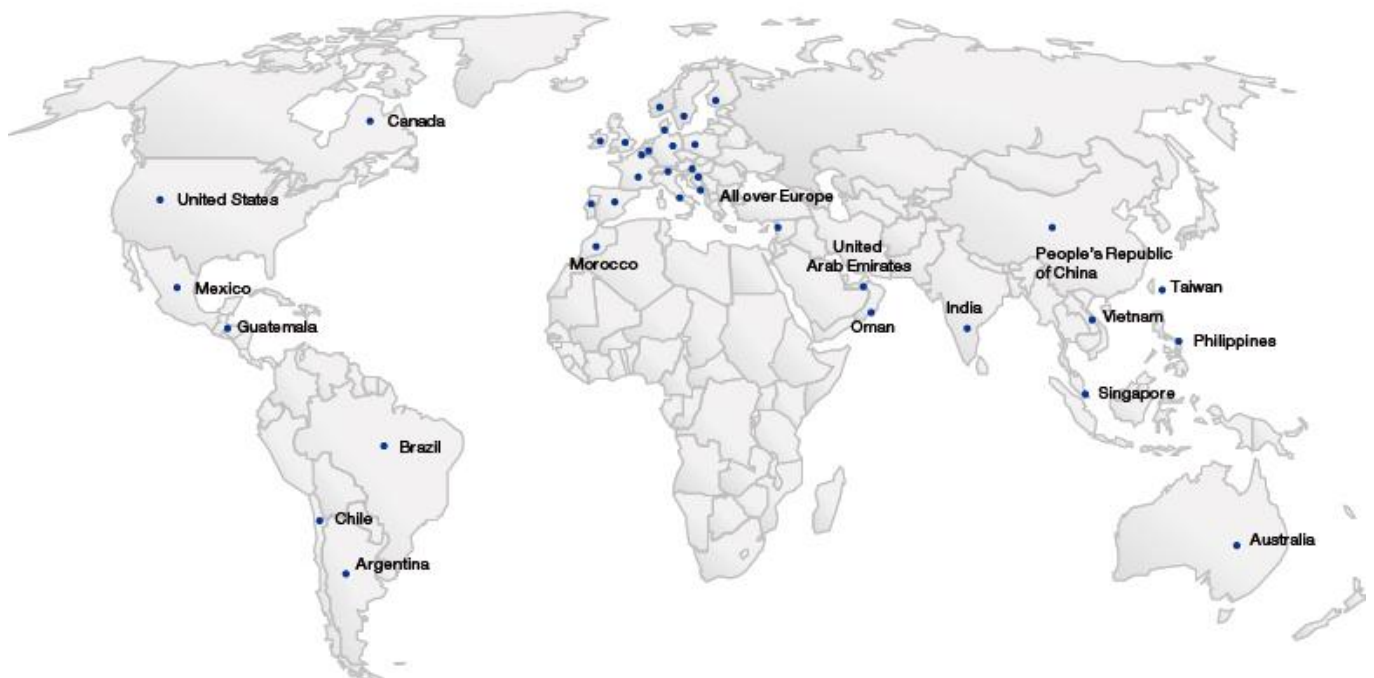
## 2.2 Internationale setting

Capgemini is een groot Internationaal bedrijf en heeft locaties over heel de wereld. Tegenwoordig komt het vaak voor dat bedrijven (IT-gerelateerd) werk laten uitbesteden in lage loon landen zoals India en China. In het geval van Capgemini gaat dit om eigen vestigingen in deze landen, vooral India. Dit wordt ook wel offshore outsourcing genoemd.

Het voordeel hiervan is dat er vaak flink bespaart kan worden op kosten, dit komt voornamelijk door de lage lonen in deze landen. Op een gemiddeld project kan vaak al zo'n 35% aan kosten bespaard worden door deze te outsourcen. Daarnaast komt het de kwaliteit van de software vaak ten goede omdat het ontwikkelen van software in Nederland vaak als tussenstap gezien wordt en bij offshore als einddoel. De kwaliteit van de programmeurs is daarom hoger en 90% van de offshore projecten blijft binnen de tijd en budget.

Het nadeel hiervan is dat het in dit soort landen vaak ontbreekt aan deskundige visie en het benodigde management. Door de grote cultuur verschillen gaat het communiceren met westerse klanten niet altijd soepel. Dit soort werk wordt dan ook vaak nog in Nederland gedaan.

Omdat dit project een groot aantal technisch aspecten heeft zal er veel contact zijn met collega's uit India. Zij hebben vaak al de ervaring met dit soort projecten en hebben meer technische kennis dan collega's uit Nederland. Het is dan ook een uitdaging om de juiste mensen te vinden in India en te leren communiceren met mensen uit de Indiase cultuur.



Figuur 2.3: Capgemini wereldwijd

meer dan 115 000 werknemers in 40 landen

## 3. Opdracht omschrijving

### 3.1 Probleemstelling

In de huidige SAP systemen kan een Quality Management module ingericht worden. Met behulp van deze module is het mogelijk om de uitvoer van inspecties te administreren. Bij inspecties moet gedacht worden aan het controleren van grote machines op eventuele gebreken. Een inspecteur bekijkt het inspectieplan, voert de inspectie uit en noteert de resultaten op papier. Later moeten deze resultaten ingevoerd worden in het systeem. Voor elk product zijn er van te voren kwaliteitseisen vastgesteld waar het product aan moet voldoen. Met behulp van deze inspecties wordt bepaald of het product aan die eisen voldoet.

Het probleem dat men op dit moment heeft is dat de inspecteur de machine controleert, en deze gegevens noteert op papier. Nadat de inspectie voltooid is gaat de inspecteur naar een computer met toegang tot het systeem en moet vervolgens deze informatie invoeren. Hij moet de resultaten dus dubbel registreren. Hierbij gaat veel tijd verloren en is een verhoogde kans op fouten door verkeerd overnemen.

Een ander probleem is dat wanneer de inspectieresultaten afwijken van de norm er een reparatie melding gedaan moet worden. Op dit moment gebeurt dit nog handmatig, en moet alles overgetypt worden in het SAP systeem.

### 3.2 Aanleiding

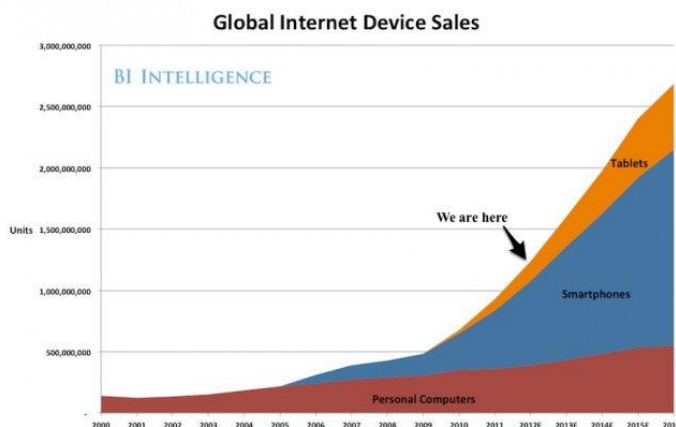
Tegenwoordig lijkt het heel normaal dat iedereen op straat een smartphone met internet heeft, maar 10 jaar geleden wist niemand wat het precies was. Zo snel is het gegaan met de ontwikkeling van smartphones.

In 1990 heeft Bill Gates al een presentatie gehouden met de titel "Information at Your Fingertips". Hiermee doelde hij op de "ouderwetse" PC en had hij misschien beter de titel kunnen geven "Information in Your Hand", maar hiermee gaf hij toen al aan wat de voordelen zijn als de (benodigde) informatie dicht bij de hand is.

#### 3.2.1 Opkomst van mobiele apparaten

Al in 1999 werd er voorspeld dat binnen enkele jaren de focus van ontwikkelingen op mobiel gebied vooral op de mobiele applicaties zou komen te liggen. Dit heeft iets langer geduurd dan voorspeld. De reden dat dit zo lang heeft geduurd ligt aan het feit dat er eerst een bepaald aantal gebruikers van mobiele apparaten nodig is, voordat de bedrijven en ontwikkelaars geld zien in deze bedrijfstak. Het verkrijgen van dit aantal gebruikers is pas echt begonnen met het uitbrengen van de eerste iPhone in 2007. Op dat moment is het aantal gebruikers hoog genoeg geworden om deze kettingreactie in gang te zetten. De verwachting is dat door de ontwikkeling van veel verschillende soorten mobiele applicaties het aantal gebruikers alleen nog maar zal toenemen.

Sinds de komst van de iPad in 2010 is er een stijgende lijn te zien in het aantal gebruikers van tablets. De ontwikkeling van deze tablets, die doorgaans sneller zijn en grotere schermen hebben, heeft de ontwikkelaars de kans geboden om betere applicaties te bouwen met meer functionaliteiten.



Figuur 3.1: Opkomst mobiele apparaten

### 3.2.2 Verschillende mobiele platformen

Net zoals er op het gebied van computers verschillende besturingssystemen zijn, zijn er ook bij de smartphones verschillende besturingssystemen. De meest voorkomende mobiele besturingssystemen in 2012 zijn:

- Google Android
- Apple iOS
- Microsoft Windows Phone

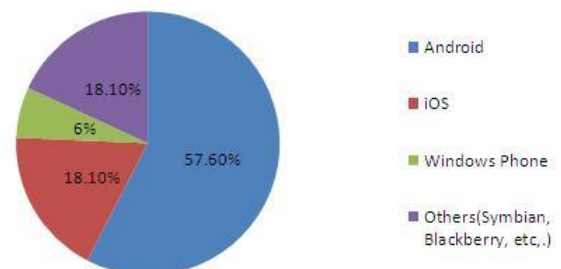
Zoals in figuur 3.2 te zien is heeft Android in 2012 het grootste marktaandeel in de smartphone wereld. Microsoft heeft voor het einde van het jaar een release van de nieuwe Windows 8 op de agenda staan. Men verwacht dat ze hiermee een grote slag gaan slaan.

Met het ontwikkelen van een mobiele applicatie moet er rekening worden gehouden voor welk mobiele besturingssysteem het ontwikkeld moet worden. Elk

besturingssysteem heeft weer zijn eigen programmeertaal en ontwikkelomgeving. Het ontwikkelen voor een specifiek besturingssysteem dat lokaal op een mobiel apparaat draait wordt ook wel een native applicatie genoemd.

Naast een native applicatie is het ook mogelijk om een web-based applicatie te ontwikkelen. Het voordeel hiervan is dat het te gebruiken is op vrijwel alle besturingssystemen. Het nadeel is dat het nog wel beperkt is wat betreft de functionaliteiten die toe te passen zijn.

**Smartphone Marketshare 2012 - Digitimes Research**



**Figuur 3.2: Smartphone markt 2012**

### 3.2.3 Enterprise mobility

Tegenwoordig hebben steeds meer mensen een smartphone en een trend op dit moment is dat ook steeds meer werkgevers hun medewerkers stimuleren om een smartphone aan te schaffen. Soms krijgen ze er een van de werkgever of is er de mogelijkheid om een smartphone te declareren. Hierdoor zijn medewerkers beter bereikbaar en kunnen ze overal hun mail lezen.

Een ander bijkomend voordeel is dat er op smartphones de mogelijkheid is om mobiele applicaties te installeren. Het is voor bedrijven dan ook erg interessant om te kijken of er applicaties zijn of ontwikkeld kunnen worden die hun bedrijfsprocessen kunnen ondersteunen. Applicaties die geïntegreerd zijn met de bedrijfsprocessen, servers en databases. Mensen van InsiderResearch hebben recent (2012) onderzoek gedaan naar mobility binnen grote organisaties en kwamen met de volgende verwachtingen:

- Het percentage organisaties met een organisatie brede mobiele strategie zal binnen drie jaar verdubbelen van 32% naar 67%.
- Driekwart van de organisaties verwachten over drie jaar hun mobiele applicaties op verschillende platformen te kunnen draaien.
- 32% van de organisaties willen vijf of meer mobiele applicaties ontwikkelen in de komende 12 maanden

Waar het voorheen gebruikelijk was om één applicatie te bouwen met daarin alle benodigde functionaliteiten, bouwt men tegenwoordig juist meerdere kleinere applicaties, elk specifiek voor een rol binnen een bepaald proces. Hiermee wordt er voorkomen dat de gebruiker een grote applicatie moet downloaden terwijl er maar een klein deel van de applicatie daadwerkelijk gebruikt wordt. Hierdoor kunnen gebruikers efficiënter werken.

### 3.3 Doelstelling

Het doel van de opdracht is het ontwikkelen van een mobiele applicatie die het inspectie proces in SAP ondersteund. In de applicatie moet het mogelijk zijn voor inspecteurs om snel en eenvoudig de geplande inspecties in te zien en de resultaten hiervan invoeren in het huidige SAP back-end systeem. Door deze mobiele applicatie moet het inspectie proces efficiënter worden, sneller en met minder fouten.

Om de applicatie te laten samenwerken met SAP moet er een koppeling komen tussen het inspectie proces in SAP en het Sybase Unwired Platform (SUP). SUP is sinds kort onderdeel van SAP en dient om de zakelijke toepassing van mobiele oplossingen te managen. Het functioneert als een tussenlaag voor mobiele apparaten en het SAP systeem. Dit soort middleware wordt ook wel een Mobile Enterprise Application Platform (MEAP) genoemd.

Sybase Unwired platform is een nieuw product, er is dan ook nog weinig praktische kennis beschikbaar binnen Capgemini. Bij het bouwen en ontwerpen van de applicatie zullen er dan ook veel vragen optreden. Omdat de kennis er niet is, zal ik zelf naar oplossingen moeten zoeken. ik zal een onderzoek uitvoeren naar Sybase Unwired Platform om erachter te komen waar rekening mee gehouden moet worden en wat de valkuilen zijn. Het ontwikkelen van een applicatie met Sybase Unwired Platform zal een uitdaging worden en hier ligt ook een stukje complexiteit van de opdracht.

### 3.4 Resultaat

Het resultaat is een mobiele applicatie waarmee de inspecteur sneller en efficiënter zijn inspecties kan uitvoeren. De resultaten worden automatisch in het SAP systeem verwerkt en hoeft dus niet meer apart door een inspecteur gedaan te worden. Deze inspectie applicatie heeft een grote meerwaarde voor de gebruikers:

- Wettelijke bepalingen: Bij juridische geschillen moet er aangetoond kunnen worden dat bepaalde producten na de service- en onderhoudsbeurt correct zijn overgedragen;
- Veiligheid: Meerdere en gestandaardiseerde inspecties kunnen de veiligheid verhogen;
- Inspecteur efficiency: Metingen hoeven maar één keer ingevoerd te worden. Metingen noteren en later overtypen is verleden tijd. Fouten worden op deze manier voorkomen.
- Onderhoud efficiency: Door inspecties kan het preventieve onderhoud beter worden geanalyseerd en geoptimaliseerd.

Deze opdracht zal ook een meerwaarde zijn voor Capgemini. Er wordt onderzocht hoe er een applicatie ontwikkeld kan worden met behulp van het Sybase Unwired Platform. Het wordt ook duidelijk hoe er een koppeling realiseerbaar is tussen de mobiele applicatie, Sybase Unwired Platform en een SAP back-end. Bovendien wordt er praktische ervaring opgedaan op het gebied van Sybase Unwired platform. Het is dan ook belangrijk om dit goed te kunnen overdragen aan het einde van het project.

## 4. Aanpak

Om het project tot een goed einde te brengen is het van belang om van te voren goed na te denken over de aanpak. In dit hoofdstuk beschrijf ik hoe ik het onderzoek naar Inspectie management en het Sybase Unwired Platform gedaan heb en welke ontwikkelmethode ik heb gebruikt voor het ontwikkelen van de mobiele applicatie. Daarnaast zal ik ook de planning en de op te leveren producten beschrijven.

### 4.1 Onderzoek aanpak

Voordat ik de requirements kan gaan opstellen en de applicatie kan gaan ontwikkelen moet ik meer kennis hebben van inspectie management in SAP en Sybase Unwired Platform. Ik heb daarom eerst een onderzoek gedaan naar inspectie management in SAP en daarna wat het Sybase Unwired platform is en hoe er applicaties mee ontwikkeld kunnen worden.

Het onderzoek naar inspectie management heb ik gedaan met behulp van interne experts. Ik heb veel met deze experts gesproken en op deze manier het inspectie proces in SAP onderzocht. Hierdoor heb ik dit proces leren begrijpen en weet ik wat het doel van inspectie management is.

Het onderzoek naar Sybase Unwired Platform is een literatuuronderzoek, dit houdt in dat de informatie die gevonden is vooral uit bestaande literatuur komt. Dit kunnen bijvoorbeeld wetenschappelijke artikelen op internet, boeken of bestaande onderzoek verslagen zijn. Aan de hand van deze informatie zijn vervolgens eigen conclusies getrokken.

Voor het zoeken van informatie op internet is vooral gebruik gemaakt van de zoekmachine “Google”, de wetenschappelijke artikelen via zoekmachine “Google Scholar” en de technische documenten op de website van Sybase. Als zoekmethode heb ik gebruik gemaakt van de sneeuwbal methode. Dit houdt in dat je bij de inleidende tekst over het onderwerp kijkt in de literatuurlijst. Die artikelen zoek je op, en gebruik je waar nodig. Ook hier kijk je weer in de literatuur lijst en ook dit gebruik je waar nodig enzovoort. Hierdoor wordt de lijst van literatuur steeds groter (net als een sneeuwbal). Het is belangrijk om hier in de gaten te houden wat de hoofdvraag is omdat je snel geneigd bent om alles uit te zoeken.

De eerste stap is het opstellen van de onderzoeksvraag. Het doel van dit onderzoek is erachter komen wat het Sybase Unwired platform is en hoe er applicaties mee ontwikkeld kunnen worden. Ook hoe er een koppeling met een back-end systeem gemaakt kan worden en of het uiteindelijk een goede keuze is om het Sybase Unwired Platform te gebruiken voor het bouwen van deze applicatie. Vandaar dat de hoofdvraag luidt:

“Wat is Sybase Unwired Platform en hoe kun je er aan SAP gekoppelde mobiele applicaties mee ontwikkelen?”

Vervolgens heb ik een aantal deelvragen opgesteld. Aan de hand van deze deelvragen kan er antwoord worden gegeven op de hoofdvraag. De deelvragen zijn opgesteld door de hoofdvraag op te splitsen in drie delen. Het eerste deel gaat over SUP en wat het kan, het tweede deel over hoe ik de koppeling kan maken met SAP en in het derde deel hoe er een mobiele applicatie ontwikkeld kan worden.

De volgende deelvragen zijn opgesteld:

1. Wat is Sybase Unwired Platform?
2. Wat is een Mobile Enterprise Application Platform (MEAP)?
3. Wat zijn de voordelen van MEAP?
4. Wat zijn de nadelen van MEAP?
5. Hoe kun je Sybase Unwired Platform koppelen met een SAP back-end?
6. Hoe kunnen er applicaties ontwikkeld worden met Sybase Unwired Platform?
7. Welke architectuur keuzes moeten er gemaakt worden?

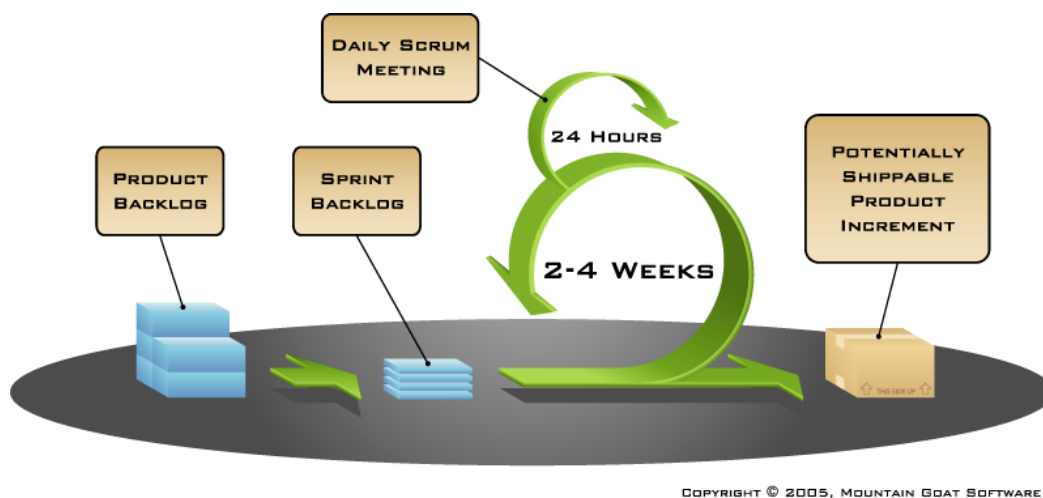
## 4.2 Ontwikkelmethode

Voor bouwen van de applicatie heb ik gebruik gemaakt van de SCRUM methode. Dit is een Agile methode waarbij in korte sprints werkende software wordt opgeleverd. Scrum is zeer goed geschikt voor projecten die niet van te voren helemaal uit te schrijven zijn. De wensen en eisen zijn namelijk gedurende het hele project nog aan te passen. Scrum is gekozen voor dit project omdat ik gewerkt heb met een vrij nieuw product en daarom wist ik van te voren nog niet precies wat ik allemaal tegen zou komen en wat de mogelijkheden zijn.

Bij scrum wordt er van te voren een product backlog gemaakt. Hierin staan alle functionele eisen van het systeem gesorteerd op prioriteit. De eis met de hoogste prioriteit staat bovenaan en zal als eerste ontwikkeld worden. Zodra alle eisen duidelijk zijn wordt er een sprint backlog gemaakt. Hierin staan alle eisen die in de betreffende sprint gerealiseerd zullen worden. Bij elke sprint zal er een werkend stukje software worden opgeleverd en gepresenteerd.

Om er achter te komen wat de functionele eisen zijn van het systeem heb ik eerst kennis ingewonnen van interne experts op het gebied van inspectie management. Zij weten waar de behoefte op de markt ligt en wat de bottlenecks zijn in het huidige inspectie proces. Deze experts heb ik zelf moeten zoeken en benaderen. Omdat Capgemini een detacheer bedrijf is zitten de experts vaak bij een klant voor een project. Het was daarom van belang om ruim van te voren afspraken te plannen met de experts.

Daarnaast heb ik een aantal architectuur keuzes gemaakt. Bijvoorbeeld de keuze of het een native of web-based applicatie gaat worden. Dit hangt af van de wensen van de opdrachtgever en deze keuze is dan ook in nauw overleg gemaakt. Deze keuzes zijn in het architectuur hoofdstuk verder beschreven.



Figuur 4.1: Scrum

Bij scrum ligt de focus vooral op het samenwerken maar omdat dit een individuele afstudeer opdracht was, is er niet gewerkt in een team. Wel waren er andere studenten op de afdeling aan een zelfstandig project bezig. Om toch een moment te creëren waarin de problemen besproken konden worden is er een wekelijkse scrum meeting gehouden. Hierin werd er dan kort verteld wat er gedaan was, wat je ging doen en of er nog problemen waren. Er was een project manager aangesteld vanuit Capgemini. Zij heeft de scrum meetings begeleid. Ook kon ik bij de project manager terecht voor afstudeer gerelateerde vragen.

Daarnaast was er wekelijks contact met mijn inhoudelijke begeleider om de vorderingen van het project te bespreken. Ook was dat het moment om problemen te bespreken en vragen te stellen. Tijdens deze afspraken werden ook de producten opgeleverd.



### 4.3 Plan van aanpak

De eerste week was de oriëntatiefase waarin ik onderzocht heb wat voor soort bedrijf Capgemini is en welke producten en diensten zij levert. Ook heb ik deze fase gebruikt om de juiste mensen te vinden die kunnen helpen met het project. Om het project verder op te kunnen starten is er een plan van aanpak opgesteld. In dit document wordt de aanpak van het project beschreven. Het is de definitie van het project en kan worden gebruikt als leidraad. In dit document worden onder andere het doel, de planning, de methodes, de op te leveren producten en de risico's van het project vastgesteld. Dit plan van aanpak zal worden besproken met de opdrachtgever zodat het voor beide partijen een duidelijk document is.

#### 4.3.1 Planning

Om te zorgen dat het project binnen 17 weken is afgerond heb ik een planning opgesteld. De weken die hierin aangegeven worden zijn de weken van het project. In het afstudeerplan is een lijst met alle op te leveren producten opgesteld. Aan de hand van deze lijst is de planning gemaakt. Ik heb beschreven welke activiteiten er gedaan worden en welke producten daaruit zullen komen. Deze planning is de meest recente versie van het project. De versie die ik aan het begin van het project gemaakt heb staat in het plan van aanpak.

Week →	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
Activiteit ↓																	
Oriëntatie																	
- Oriëntatiefase																	
- Plan van Aanpak																	
Back-end Systeem Kiezen																	
- Back-end systeem																	
Inlezen Inspectie management																	
- Beschrijving huidige situatie																	
- Beschrijving doelstelling																	
Onderzoek Sybase Unwired Platform																	
- Onderzoeksrapport																	
Opstellen Requirements																	
- Proces flow																	
- Requirements																	
- Use Cases																	
Opstellen Architectuur																	
- Architectuur																	
Sprint 1: gereed maken systeemdeel																	
- Ingericht back-end system																	
- Workspace																	
- Sybase server																	
Sprint 2: Bouwen basis applicatie																	
- Ontwerp																	
- applicatie																	
Afronden Scriptie																	
- Eindverslag																	
Schrijven Scriptie																	
- Eindverslag																	

### 4.3.2 Op te leveren producten

- **Plan van aanpak**

In het plan van aanpak is het doel en resultaat van dit project beschreven en volgens welke methodes ik dit ga aanpakken. Ook wat de risico's zijn waar rekening mee gehouden moet worden en welke randvoorwaarden er zijn om het project te kunnen starten.

- **Onderzoek rapport Sybase Unwired Platform**

In dit rapport wordt onderzocht en beschreven wat het Sybase Unwired Platform is, hoe er applicaties mee ontwikkeld kunnen worden en hoe er een koppeling gemaakt kan worden met een back-end SAP systeem.

- **Product backlog**

Hierin staan de requirements van het systeem beschreven, ook wel user stories genoemd. Deze staan op prioriteit gesorteerd. Elke requirement krijgt een geschatte waarde van moeilijkheid. Ook worden de uiteindelijke requirements beschreven in de vorm van Use Case beschrijvingen. Het gehele proces zal ook worden getekend in een proces flow.

- **Architectuur document**

Er zal beschreven worden hoe de architectuur van mijn applicatie eruit ziet, en welke keuzes er gemaakt zijn.

- **Ontwerp**

Het ontwerp van het systeem. hierin zal de structuur van de applicatie duidelijk worden door middel van een datamodel en een functioneel ontwerp.

- **Applicatie**

Aan het eind van dit project zal er een mobiele applicatie gebouwd en opgeleverd worden waarmee het inspectie proces van SAP ondersteund wordt

- **Testplan**

In het test plan wordt beschreven wat er getest moet worden en hoe dat gedaan wordt. Dit moet gebeuren om te bepalen of het systeem aan de kwaliteitseisen voldoet.

- **Testrapport**

Het systeem zal uiteindelijk getest worden volgens het testplan. De resultaten zullen beschreven worden in het testrapport.

## 5. Onderzoek naar Sybase Unwired Platform

Om een goed beeld te krijgen van Sybase Unwired Platform heb ik een onderzoek uitgevoerd. De hoofdvraag van dit onderzoek luidt:

“Wat is Sybase Unwired Platform en hoe kun je er mobiele applicaties mee ontwikkelen met een SAP koppeling?”

Het doel van dit onderzoek is om erachter te komen wat het Sybase Unwired platform is en hoe er applicaties mee ontwikkeld kunnen worden. Daarnaast laat dit onderzoek zien hoe er een koppeling met een back-end systeem gemaakt kan worden en wanneer het uiteindelijk een goede keuze is om het Sybase Unwired Platform te gebruiken voor het bouwen van mobiele applicaties.

Om deze hoofdvraag te kunnen beantwoorden zijn er de volgende deelvragen opgesteld:

### 1. Wat is Sybase Unwired Platform?

Deze deelvraag moet de lezer een beeld geven van wat het Sybase Unwired Platform precies is en wat ermee kan. Ook moet duidelijk worden uit welke onderdelen het platform bestaat

### 2. Wat is een Mobile Enterprise Application Platform (MEAP)?

In deze deelvraag wordt dieper ingegaan op het begrip “MEAP”. Wat is het precies en wat kun je ermee

### 3. Wat zijn de voordelen van MEAP?

Bij deze deelvraag worden de voordelen van MEAP besproken en waarom een bedrijf MEAP zou willen gebruiken.

### 4. Wat zijn de nadelen van MEAP?

Deze deelvraag beschrijft de nadelen van MEAP en hoe je het beste rekening kunt houden om risico's van de nadelen zo minimaal te laten zijn op de oplossing.

### 5. Hoe kun je Sybase Unwired Platform koppelen met een SAP back-end?

Het antwoord op deze deelvraag beschrijft hoe er een koppeling gemaakt kan worden tussen een SAP back-end en het Sybase Unwired Platform.

### 6. Hoe kunnen er applicaties ontwikkeld worden met Sybase Unwired Platform?

Door deze deelvraag te beantwoorden moet het duidelijk worden hoe er een applicatie ontwikkeld kan worden met behulp van SUP. Welke aspecten er bij komen kijken en waar rekening mee gehouden moet worden.

### 7. Welke architectuur keuzes moeten er gemaakt worden?

Bij het gebruiken van het Sybase Unwired Platform moeten er een aantal architectuur keuzes gemaakt worden. Bij het beantwoorden van deze deelvraag worden deze keuzes verder beschreven.

## 5.1 Wat is Sybase Unwired Platform?

De applicatie is gebouwd met behulp van het Sybase Unwired Platform(SUP). SUP is een onderdeel van SAP om de zakelijke toepassing van mobiele oplossingen te managen. Het functioneert als een tussenlaag voor mobiele apparaten en het SAP systeem. Om hier meer inzicht in te geven zal er in deze deelvraag antwoord worden gegeven over wat SUP precies is, waar het uit bestaat en wat je er mee kunt.

Het Sybase Unwired Platform is een Mobile Enterprise Application Platform(MEAP). Dit platform maakt het mogelijk om meerdere mobiele applicaties te ontwikkelen die werknemers snel en veilig toegang geven tot de back-end informatie van het bedrijf. Sybase ondersteunt de mobiele platformen van Android, Blackberry, Windows Phone en Apple IOS.

Het Sybase Unwired System is geschikt voor grote bedrijven met meerdere back-end systemen en waar veel verschillende mobiele apparaten gebruikt worden door een groot aantal medewerkers. Deze bedrijven hebben vaak een grote complexe infrastructuur. Bij bedrijven die dat nog niet hebben zal dat waarschijnlijk binnen korte tijd wel gebeuren. Wanneer er geen gebruik gemaakt wordt van een pakket als Sybase Unwired Platform zal het snel een onoverzichtelijk spinnenweb van verbindingen en datastromen worden.

Met Sybase Unwired Platform is het mogelijk om mobiele applicaties te:

- koppelen
- bouwen
- gebruiken
- beheren

Het koppelen van het Sybase Unwired Platform kan met traditionele back-end systemen zoals databases, web services en applicaties zoals SAP. Hierdoor is het binnen vrijwel ieder bedrijf goed te implementeren in de bestaande infrastructuur. Wel moet er rekening mee worden gehouden dat het Sybase Unwired Platform zich vooral richt op SAP systemen. Andere back-end systemen worden wel ondersteund maar niet zo goed als SAP systemen. Vaak selecteren bedrijven hier SUP ook op.

Het bouwen van applicaties kan redelijk eenvoudig door het ontwikkelen van simpele web based applicaties, of door meer complexere native applicaties.

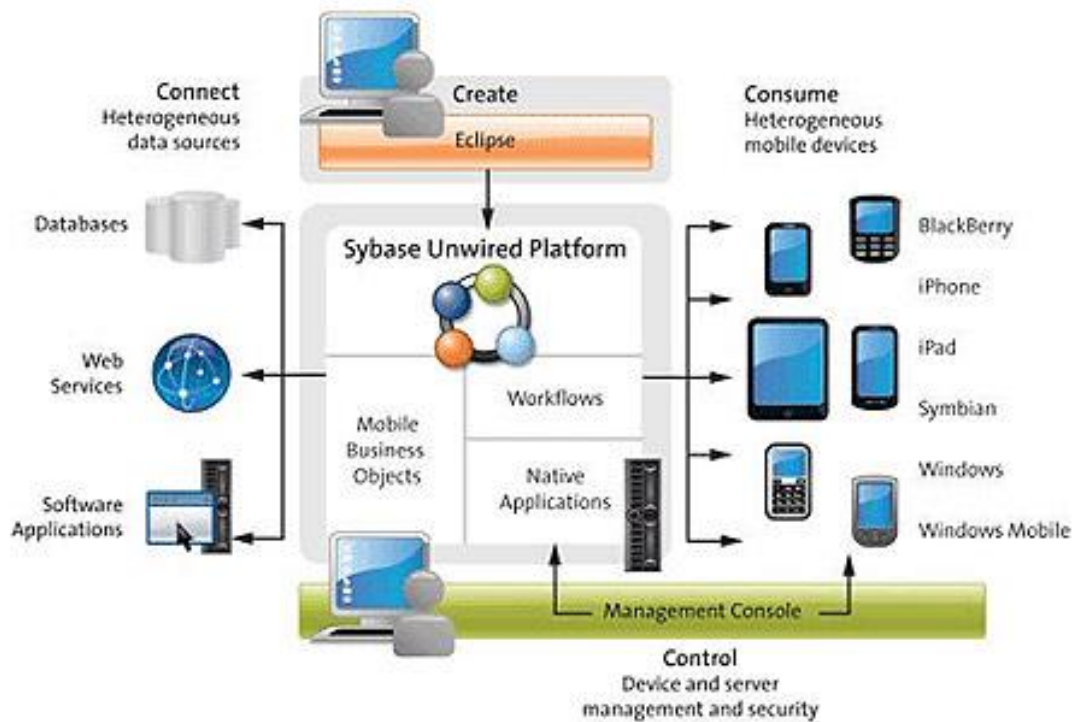
Een simpele web based applicatie wordt ook wel een workflow applicatie genoemd. Dit is een eenvoudige applicatie waarin simpele werkzaamheden gedaan kunnen worden, bijvoorbeeld het opvragen van lijsten en het eventueel invoeren van nieuwe data.

Een native applicatie is, zoals de naam al aangeeft, een applicatie die native draait op het mobiele apparaat. Hierbij is het dus noodzakelijk om voor elk platform een aparte applicatie te ontwikkelen. Het voordeel van een native applicatie is dat er veel complexere applicaties ontwikkeld kunnen worden die meer integratie met de mobiele apparaten nodig hebben.

De verbinding tussen de applicatie en de back-end wordt ook beheerd door het Sybase Unwired Platform. De applicatie maakt niet rechtstreeks verbinding met de back-end maar met de Unwired server, deze maakt weer verbinding met de back-end. Hiermee wordt voorkomen dat een back-end teveel aanvragen krijgt

Tot slot is het ook mogelijk om met SUP de applicaties te beheren. Hiermee wordt bedoeld het registreren van apparaten, het synchroniseren van de data en het beschikbaar maken van updates. Voor het beheer van de applicaties maakt SUP gebruik van Afaria. Dit is een onderdeel van Sybase voor het beheer, configuratie en beveiliging van mobiele apparaten. Afaria maakt het een administrator mogelijk om nieuwe applicaties te pushen naar mobiele apparaten en oude applicaties te vergrendelen of zelfs te verbergen. Zo is het mogelijk om met behulp van GPS een bepaald applicatie te verbergen waar deze niet gebruikt mag worden. Denk hierbij aan het vergrendelen van je camera applicatie wanneer je een bedrijf binnenloopt waar je niet mag fotograferen. Daarnaast kan de administrator ook connectie en configuratie instellingen te doen. Dit bespaart de helpdesk kosten en de kosten om nieuwe applicaties uit te rollen.

Onderstaand figuur geeft een totaal overzicht van Sybase Unwired Server structuur en de onderdelen waar het uit bestaat.



**Figuur 5.1: Architectuur Sybase Unwired Platform**

- **Sybase Unwired Server**  
Het platform zelf is een server die alle verbindingen reguleert tussen mobiel en back-end. Dit platform ondersteunt Workflows en native applicaties.
- **Back-end systeem**  
Met het Sybase Unwired Platform is het mogelijk om een bestaand bedrijfssysteem te koppelen aan mobiele applicaties. Dit kunnen zowel databases, webservices zijn als bestaande systemen zoals SAP.
- **Workspace**  
Dit is de Software Development Kit van het platform die wordt gebruikt voor bouwen van de applicaties. Deze Workspace is gebaseerd op Eclipse. Wanneer er workflow applicaties worden gebouwd maak je alleen maar gebruik van de Workspace. Wanneer het gaat om native applicaties heb je ook de ontwikkelomgeving van het betreffende mobiele platform nodig.
- **Mobiel apparaat**  
De applicatie zal uiteindelijk worden gebruikt op een mobiel apparaat. Dit kan zowel een smartphone als een tablet zijn op alle mogelijke platformen. SUP reguleert hierbij onder andere de verbindingen die lopen tussen apparaat en back-end.
- **Sybase Control Center**  
SCC is de Management console van het platform voor het beheren van de applicaties. In de SCC kun je real-time de server status, prestaties en beschikbaarheid monitoren. Ook kan het meldingen genereren op het moment dat er iets niet juist functioneert.

Daarnaast is het binnen de SCC mogelijk om alle gebruikeraccounts te beheren. Er kunnen nieuwe apparaten toegevoegd worden en je kunt bepaalde applicaties aan apparaten toewijzen. Ook updates van applicaties kun je via de SCC eenvoudig toepassen op alle verschillende apparaten.

## 5.2 Wat is een Mobile Enterprise Application Platform?

Voor het ontwikkelen van bedrijfsspecifieke mobiele applicaties die geïntegreerd kunnen worden met back-end bedrijfssystemen maakt men tegenwoordig gebruik van een Mobile Enterprise Application Platform (MEAP).

“Een Mobile Enterprise Application Platform is een pakket van producten en services die de ontwikkeling van mobiele applicaties mogelijk maken”.

Met behulp van MEAP is het eenvoudiger om mobiele applicaties te ontwikkelen die volledig integreren met bedrijfsprocessen, servers en databases. Deze applicaties kunnen gebouwd worden voor verschillende mobiele platformen, maar is niet te vergelijken met een cross platform. Het is mogelijk om met behulp van MEAP één onderliggende structuur te ontwikkelen voor applicaties die draaien op de verschillende mobiele platformen. Maar om de applicatie echt te kunnen gebruiken op de verschillende platformen zijn er vaak nog aanpassingen nodig.

Naast het ontwikkelen van applicatie kan MEAP ook gebruikt worden voor het beheren van de verschillende mobiele apparaten, netwerken en gebruikers. Het beheer gebeurt tijdens de hele lifecycle van een applicatie. Het start bij het uitrollen van een applicatie maar gaat ook door tijdens het gebruiken van de applicatie



Figuur 5.2: Principe van MEAP

Een MEAP wordt tussen de mobiele apparaten en het back-end systeem geplaatst, en is daarom een “middleware” oplossing.

Om te bepalen wanneer iets een MEAP genoemd mag worden wordt gebruik gemaakt van de “regel van drie”. Deze regel beschrijft wat het platform minimaal moet ondersteunen voordat het een MEAP genoemd mag worden.

MEAP moet minimaal:

- **drie mobiele applicaties ondersteunen**  
Wanneer er meer dan drie mobiele applicaties gebruikt worden gaat er heel veel verschillende data heen en weer tussen de mobiele apparaten en het back-end systeem. Het kan voorkomen dat verschillende applicaties gebruik maken van dezelfde data. Wanneer er gebruikt gemaakt wordt van een MEAP scheelt dit aanzienlijk in het aantal data dat over en weer wordt gestuurd, omdat deze data maar één keer uit het back-end gehaald hoeft te worden en wordt opgeslagen in de server.
- **drie mobiele besturing systemen ondersteunen**  
Als er meer dan drie verschillende mobiele besturingssystemen ondersteund worden door de applicatie komt er veel extra werk bij voor het beheren van deze applicaties. Ook heeft elke besturingssysteem weer een eigen manier van communiceren en synchroniseren. Door gebruik te maken van MEAP wordt het beheer, de koppeling en het synchroniseren gedaan door één platform. Daarnaast is kun je met MEAP veel code hergebruiken wanneer je een applicatie ontwikkeld voor verschillende mobiele besturing systemen
- **drie verschillende back-end systemen ondersteunen**  
Grote bedrijven hebben vaak meer dan één back-end systeem en medewerkers hebben allemaal toegang nodig tot een of meerdere back-end systemen. Wanneer dit door middel van een direct connectie gaat krijg je een onoverzichtelijke chaos aan verbindingen, en worden gegevens inefficiënt over en weer gestuurd. Met MEAP worden alle mobiele apparaten en back-end systemen verbonden met het platform. Hierdoor krijg je een veel overzichtelijker en efficiëntere omgeving.

In de praktijk zijn er genoeg voorbeelden te noemen waarbij niet helemaal aan deze regel wordt voldaan. Zo is er ook een bedrijf dat een SUP applicatie heeft ontwikkeld waarbij het mogelijk is om uren in te vullen op zowel het eigen back-end systeem als het back-end systeem van de klant. In totaal dus twee verschillende back-end systemen. Dit bedrijf noemt haar oplossing ook een MEAP terwijl het maar twee back-end systemen gebruikt. Het is dan ook niet zo zwart op wit als deze regel weergeeft. Je kunt de regel meer zien als een algemene situatie.



### 5.3 Wat zijn de voordelen van MEAP?

In dit hoofdstuk worden de voordelen van MEAP besproken, met vooral de focus op Sybase Unwired platform. Dit omdat hier het eindproduct mee ontwikkeld is. Tijdens en uitvoeren van dit onderzoek ben ik erachter gekomen dat Sybase zich vooral richt op twee grote voordelen van dit systeem. Dit is ten eerste de centrale toegang voor mobiele apparaten en back-end servers en ten tweede het hergebruik van code.

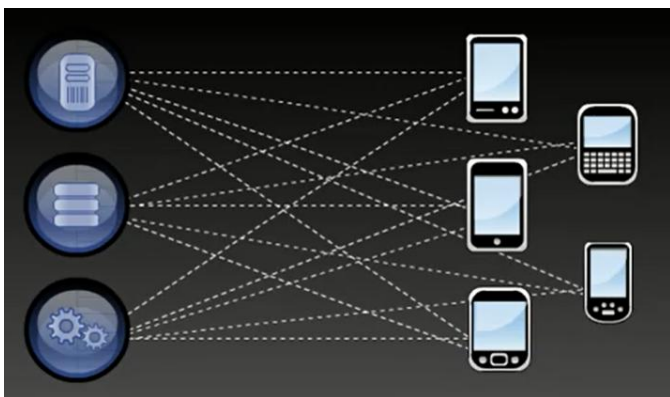
#### Centrale toegang

Een groot voordeel, en een van de speerpunten, van MEAP en ook Sybase Unwired Platform is dat alles gecentraliseerd wordt op een generieke manier. Dit zorgt voor een efficiënte en betrouwbare toegang tot alle data vanaf een mobiel apparaat, door het gezamenlijke management en de centrale beveiliging.

Elk component in de infrastructuur heeft één verbinding met de server, dus elke mobiel apparaat en elk back-end systeem. Dit zorgt er ook voor dat het eenvoudiger te beheren en overzien is welke data en hoeveel er gebruikt wordt, en welke verbindingen er zijn met welke apparaten. In tegenstelling tot andere mobiele oplossingen zoals bijvoorbeeld mobile point solution waarbij er een soort spinnenweb aan verbindingen ontstaat wanneer er veel gebruikers zijn.

Een ander voordeel van de centrale toegang is dat het hele platform goed schaalbaar wordt. Het is eenvoudig om een mobiele telefoon aan de server toe te voegen zonder dat de back-end server hier iets van merkt. Hierdoor kunnen bedrijven het platform mee laten groeien. Het enige limiet is dat een maximum aantal apparaten aan het platform gekoppeld kunnen worden. Wanneer je over dit limiet gaat is er de mogelijkheid om er een server bij te installeren. Je maakt dan gebruik van een clustered platform.

Zonder Sybase Unwired Platform



Met Sybase Unwired Platform



Figuur 5.3: Met en Zonder SUP

#### Hergebruik van code

Ook een groot voordeel is dat ontwikkelaars veel meer code kunnen hergebruiken dan normaal het geval is bij het ontwikkelen van applicatie. De ontwikkelaar bouwt één generieke basis waarin het de functionaliteiten vastlegt door middel van "mobile business objects". Dit functioneert dan als basis voor de applicatie voor alle back-end systemen en alle mobiele apparaten en applicaties die verbonden zijn met de Sybase server. Deze basis hoeft maar één keer ontwikkeld te worden voor een specifieke applicatie en is vervolgens verder aan te passen naar de specifieke verschillende platformafhankelijke mogelijkheden. Dit geldt voor zowel native als workflow applicaties. Een groot voordeel van deze basis is dat alle databases in de back-end versmolten kunnen zijn. Normaal gebeurt dit pas in de applicatie en kost dit veel resources, daarnaast levert het ook vertraging op. Met MEAP wordt alle data op de server al gecombineerd en als één pakket naar de mobiele applicatie verzonden.

Het hergebruik van de code, en de mogelijkheid om een goede basis te bouwen voor meerdere applicaties bespaart het bedrijf veel geld en tijd bij het ontwerpen en ontwikkelen van de applicatie.

## **Vermindering van data verkeer**

Sybase zorgt voor de verbinding tussen het back-end bedrijfssysteem en de mobiele apparaten. Dit is voor bijna alle bestaande back-end systemen mogelijk. Omdat Sybase hierbij functioneert als middleware pakket zorgt het voor vermindering van het aantal verbindingen met het back-end systeem. Hierdoor wordt het back-end systeem ontlast, zal het sneller werken en is er minder data verkeer nodig vanaf het back-end systeem.

## **Multi platform applicaties mogelijk**

Met Sybase is het mogelijk om native en workflow applicaties te ontwikkelen. Workflow applicaties zijn web applicaties die te gebruiken zijn op alle verschillende mobiele platformen. De applicatie is dus niet platform afhankelijk. Dit komt omdat de applicatie niet op het apparaat draait maar via het internet. Hierdoor heeft de applicatie wel altijd internet nodig. Het is mogelijk om een applicatie stand alone te gebruiken, dus zonder internet, alleen loop je dan tegen beperkte functionaliteiten en synchronisatie problemen aan. Het grote voordeel van workflow applicaties is dat er maar één applicatie ontwikkeld hoeft te worden voor zowel Android als IOS als Windows Mobile. Workflow applicaties zijn ideaal om standaard bedrijfsprocessen mobiel te ondersteunen waarbij je geen complexe berekeningen of mobiele functionaliteiten nodig hebt.

De reden dat Sybase heeft gekozen om deze workflow applicaties te ondersteunen is omdat het de snelste manier van applicaties ontwikkelen is. Het enige nadeel is dat het minder functionaliteiten kan hebben dan een native applicatie. Een bedrijf moet hier goed rekening mee houden bij het ontwikkelen. Wanneer een applicatie te snel of zonder genoeg functionaliteiten gebouwd wordt bestaat de kans dat de medewerkers de applicatie helemaal niet zullen gebruiken. Dit is zonde van de applicatie en kost het bedrijf alleen maar geld en tijd. Onderzoek heeft ook uitgewezen dat applicaties er aantrekkelijk uit moeten zien om gebruikt te worden. Maar dit moet natuurlijk niet ten koste gaan van de snelheid van de applicatie.

## **Centraal beheer**

Sybase Unwired Platform maakt het mogelijk om tijdens het gebruik van de applicaties deze te beheren. Denk hierbij aan nieuwe applicaties uitrollen, applicaties updaten en gebruikers beheren.

Wanneer er een nieuwe applicatie ontwikkeld is kun je op twee verschillende manieren de applicatie uitrollen. Je kunt via SUP de applicaties pushen naar de geregistreerde apparaten op de server. Het voordeel hiervan is dat niet alle gebruikers zelf de applicatie hoeven op te zoeken en te installeren. Daarnaast weet je zeker dat de gebruiker de juiste applicatie heeft. Hierdoor wordt de drempel kleiner voor de gebruiker om de applicatie ook daadwerkelijk te gaan gebruiken.

Het is ook mogelijk dat het bedrijf, in verband met privacy, kiest voor een eigen soort app store. Het is hiermee mogelijk om alle ontwikkelde applicaties voor een specifiek bedrijf in een soort app store te zetten waar medewerkers applicaties uit kunnen downloaden en installeren. Het voordeel hiervan is dat de gebruiker zelf controle heeft over welke applicaties er geïnstalleerd worden en het bedrijf weet welke gebruiker welke app geïnstalleerd heeft.

Met SUP kun je gebruik maken van alle soorten mobiele apparaten zoals smartphones, tablets en laptops. Het nadeel van mobiele applicaties is dat zij relatief vaak updates nodig hebben. Deze updates hebben voor alle apparaten weer een verschillende configuratie nodig. Het update van zo'n applicatie kan dan ook behoorlijk veel tijd kosten. Bij het gebruik van SUP heb je minder last van dit probleem omdat je gebruik maakt van veel hergebruikte code. Wanneer er een update nodig is hoeft er maar een klein deel van de applicatie aangepast te worden en deze updates kunnen automatisch bij de gebruikers geïnstalleerd worden.

## 5.4 Wat zijn de nadelen van MEAP?

MEAP heeft behalve veel voordelen ook een aantal nadelen. In dit hoofdstuk beschrijf ik de nadelen en hoe deze nadelen en de risico's beperkt kunnen worden.

### Extra kennis

Een voordeel van Sybase Unwired Platform is dat medewerkers minder kennis van de back-end systemen en processen nodig hebben, om bijvoorbeeld te testen. Daar staat tegenover dat er binnen het bedrijf wel extra kennis moet zijn over andere aspecten van het systeem, bijvoorbeeld voor het ontwikkelen van applicaties. Daarnaast moet het systeem gecontroleerd en beheerd worden. Een gemiddeld bedrijf dat gebruikt maakt van MEAP heeft minstens twee extra experts fulltime in dienst om deze taken uit te voeren. Wanneer je bijvoorbeeld voor een andere oplossing 3 mensen nodig voor het onderhoud heb je voor MEAP 5 mensen nodig.

### Hoge investeringskosten

Ondanks dat het bij ingebruikname van applicaties veel geld bespaart is een nadeel van MEAP is dat het hoge investeringskosten heeft. Het implementeren van het systeem kost veel geld. Ondanks dat Sybase zegt van niet, moet er bijna altijd bestaande infrastructuur aangepast worden om te kunnen integreren met het nieuwe systeem. Ook moet er in de meeste gevallen oude hardware vervangen worden. Om het systeem in gebruik te nemen zijn er consultants van de MEAP leverancier nodig. Ten slotte zijn er ook hogere kosten verbonden aan de ondersteuning en service. En is er nieuwe kennis nodig binnen het bedrijf waardoor er ook mensen geschoold moeten worden.

Dit nadeel kun je zoveel mogelijk beperken door de besparing bij het ingebruikname zo groot mogelijk te maken. Dit betekent dat er minimaal een aantal mobiele apparaten gebruikt moeten worden en ook verschillende soorten mobiele platformen. Hierdoor zorg je dat er hierop veel bespaard wordt en dit maakt het de grote investering waard. Uit onderzoek is gebleken dat dit minimale aantal gebruikers ligt op 150 mensen.

### Single point of failure

Een groot risico in het gebruiken van een MEAP als Sybase is dat het een single point of failure heeft. Dit houdt in dat heel het platform plat kan komen te liggen op het moment dat één van de vele onderdelen niet werkt. Dit komt omdat Sybase is opgebouwd is vanuit één centraal punt. Wanneer dit gebeurt is het dus onmogelijk om nog bij de data te komen.

Dit probleem is op te lossen door een cluster van platformen te implementeren. Als er dan één platform plat ligt nemen de andere het over. Grote bedrijven gebruiken deze oplossingen al om bijvoorbeeld het aantal gebruikers te verspreiden, Hiermee kunnen ze overload van de server voorkomen. Kleinere bedrijven hebben hierbij een nadeel, en moeten kiezen tussen het risico dat het systeem plat komt te liggen en de hoge kosten van meerdere platformen.

### Synchronisatie

Een vierde punt is dat de mobiele apparaten tegenwoordig overal en altijd verbinding kunnen maken met het internet. Ze kunnen ook eenvoudig wisselen met verbindingen of de verbinding even verbreken. Daarnaast is het mogelijk om met een gedeeltelijke kopie van de database te werken wanneer het apparaat offline is. Het nadeel hiervan is dat dit kan gebeuren tijdens een synchronisatie, wat weer corrupte data kan opleveren op het apparaat of op de server. Dit is een probleem waar alle mobiele applicaties mee te maken hebben, maar met bedrijfsinformatie is het een extra belangrijk om erop te letten dat er met de juiste data gewerkt wordt.

Dit risico kun je beperken door in zo'n geval de gebruiker een bericht te sturen met het verzoek om handmatig te updaten omdat er iets gewijzigd is op de server. Wanneer deze update gedaan is werkt de gebruiker weer met de juiste informatie. Sybase biedt de mogelijkheid om te controleren of de data op het apparaat hetzelfde is als de data op de server. In de back-end moet dan een algoritme worden ingebouwd dat kan omgaan met dit type synchronisatie.

Het probleem dat je krijgt bij synchroniseren is wanneer er meerdere gebruikers tegelijkertijd aan de data werken. Sybase kan hier nog niet goed mee omgaan, zelfs niet wanneer er een constante verbinding en synchronisatie is. Dit is nog een groot gebrek van het platform. Een oplossing hiervoor moet op dit moment nog in de back-end plaatsvinden.

Nu de voor en nadelen van Sybase Unwired bekeken zijn denk ik dat de voordelen wel opwegen tegen de nadelen. Wanneer je de nadelen van het platform weet kun je hier rekening mee houden tijdens het ontwikkelen en installeren. Op deze manier kunnen de meeste nadelen beperkt of zelfs voorkomen worden.

## 5.5 Hoe kun je Sybase Unwired Platform koppelen met een SAP back-end?

In 2011 is SAP eigenaar geworden van Sybase. Voor dit platform hebben ze dan ook een goede integratie met SAP systemen gerealiseerd. Het is redelijk eenvoudig om een connectie te leggen tussen de Sybase Workspace en een SAP systeem. Hiervoor is alleen een systeem ID, systeem host, Inlognaam en wachtwoord nodig. Bovendien kun je de back-end connectie eenvoudig aanmaken in de Workspace.

Als de connectie gelegd is kan Sybase alle Business Objecten ophalen uit de SAP back-end. Aan de meeste van deze business objecten zitten Business Application Programmaming Interfaces (BAPI's) gekoppeld. BAPI's zorgen ervoor dat externe programma's ontwikkeld door klanten, partners of SAP applicaties, zoals Sybase, gebruik kunnen maken van de business objecten via een interface. Het is hierdoor mogelijk om gegevens uit het back-end SAP systeem op te halen, te wijzigen en te verwijderen. Ook kunnen er met behulp van deze BAPI's nieuwe gegevens ingevoerd worden.

Niet alle business objecten die nodig zijn voor de applicatie hebben een BAPI beschikbaar in SAP. Deze BAPI's kunnen dan nog gebouwd worden in Advanced Business Application Programming (ABAP), de programmeer taal van SAP. Om een BAPI te kunnen bouwen moet er wel specifieke kennis zijn op het gebied van ABAP. Als er dus nieuwe BAPI's gebouwd moeten worden, zul je veel ervaring en kennis van ABAP nodig hebben.

Als de juiste BAPI's aanwezig zijn, is het mogelijk om hiervan Mobile Busines Objects(MBO's) te maken. Hiermee kun je in de applicatie bepaalde methodes uitvoeren. Bijvoorbeeld het opvragen van informatie, of het creëren van nieuwe records.



**Figuur 5.4: Mobile Busines Objects**

Wanneer je een workflow applicatie wil ontwikkelen zal er een gebruiker moeten worden aangemaakt in de Sybase Control Center(SCC). Deze gebruiker krijgt dan toegang tot een workflow applicatie. In SCC zal de gebruiker dan gekoppeld worden aan de gebruikers in SAP, waardoor er ook rekening wordt gehouden met de rechten die de gebruiker heeft in SAP.

Wat helaas nog niet mogelijk is binnen SAP is het automatisch overnemen van de verschillende users. De beste oplossing is dat de users die in SAP staan automatisch kunnen worden gekoppeld in Sybase. Een gedeeltelijke oplossing hiervoor is het gebruik van rollen. Je kunt verschillende rollen specificeren in Sybase en deze koppelen aan een gebruiker. Dit bespaart een hoop werk.

Bij een native applicatie moet de applicatie zelf connectie leggen met de Sybase server. Hier is een flink stuk programma code voor nodig en is voor elke server en apparaat moet er gesynchroniseerd worden. Dit houdt in dat de database op de Sybase server gesynchroniseerd wordt met de data op de smartphone. Het voordeel van een native applicatie is dat wel gebruik gemaakt kan worden van de SAP gebruikers. De gebruikers kunnen dan gewoon inloggen als zichzelf en hebben de zelfde rechten als in het SAP systeem. Het is dus niet noodzakelijk om extra gebruikers aan te maken in het SCC.

## 5.6 Hoe kun je applicaties ontwikkelen met Sybase Unwired Platform?

Met Sybase Unwired Workspace is het mogelijk om mobiele applicaties te bouwen. Sybase gebruikt hiervoor de Sybase Workspace. De Sybase Workspace is de ontwikkelomgeving voor bouwen van een mobiele applicatie.

Het is mogelijk om workflow en native applicaties te ontwikkelen, maar ook om de connecties te leggen met Sybase Unwired Server, SAP back-end, SOAP en REST services. De Workspace is een plug-in op Eclipse, de omgeving ziet er dan ook hetzelfde uit. Het voordeel hiervan is dat het een bekende omgeving is waar veel ontwikkelaars als eens mee gewerkt hebben.

### Mobile Business Object

Een applicatie wordt gebouwd op basis van Mobile Business Objects(MBO), dit zijn objecten die je voor de mobiele applicatie bouwt. Deze MBO's worden gebouwd vanuit de Business Objecten in het SAP back-end. Met behulp van BAPI's kun je deze MBO's aanmaken vanuit het SAP back-end systeem.

Een MBO kun je maken door de juiste BAPI op te vragen uit het back-end systeem en input en output parameters te geven. Bijvoorbeeld wanneer je een lijst met inspecties wilt opvragen uit het back-end systeem kan je als input parameter een type inspectie meegeven en als output alle benodigde gegevens van de gevonden inspecties. Deze input en output gegevens moet je al definiëren bij het aanmaken van een MBO. Hierdoor is het belangrijk dat er voor het ontwikkelen van een applicatie al goed wordt nagedacht welke MBO's je wilt gebruiken en hoe. Elke attribuut dat in een MBO zit wordt meegenomen met de synchronisatie. Wanneer deze dan niet gebruikt wordt is er onnodig extra data verzonden en opgeslagen, dit gaat ten koste van de efficiëntie.

Nadat alle MBO's zijn aangemaakt is het mogelijk om in de Workspace de verschillende relaties tussen deze MBO's te definiëren. Dit gebeurt visueel door een lijn trekken. Vervolgens kun je aangeven welke attributen de relatie hebben (Primary key en Foreign key). Bij het ontwikkelen van de applicatie kun je deze relaties gebruiken voor het ophalen van bijvoorbeeld de juiste resultaten bij een inspectie.

Wanneer er MBO's gemaakt en relaties gedefinieerd zijn is de vervolg stap om de applicatie te gaan bouwen. Op dit moment moet de keuze worden gemaakt of het een native of een workflow applicatie wordt.

### Workflow applicatie

Als je kiest voor een Workflow applicatie kun je blijven ontwikkelen in de Workspace, en is het mogelijk om met klikken en slepen schermen te maken op basis van de MBO's. Het platform kan ook zelf de benodigde velden aanmaken. Wanneer de benodigde schermen zijn gemaakt kun je de schermen koppelen en knoppen toevoegen. Je kunt de gegenereerde schermen zelf aanpassen naar je eigen behoeftes. Zo kun je bijvoorbeeld tekst velden bij plaatsten of juist verwijderen. Bij het ontwikkelen van een workflow applicatie hoeft er niet één regel geprogrammeerd te worden.

Wanneer je alle schermen hebt aangemaakt en de applicatie klaar is kun je het project uitrollen naar de Unwired Server. In het Sybase Control Center kun je dan aangeven welke geregistreerde gebruikers er gebruik kunnen maken van de applicatie.

Met een workflow applicatie is het niet mogelijk om hele ingewikkelde applicaties te ontwikkelen met veel functionaliteiten waarbij de processor van de telefoon of andere functionaliteiten van de telefoon zelf te gebruiken is. Ook kunnen er geen offline functionaliteiten toegevoegd worden aan de applicatie. Het is bijvoorbeeld niet mogelijk om een koppeling te maken tussen de agenda van de telefoon en de uitvoer datum van de inspecties.

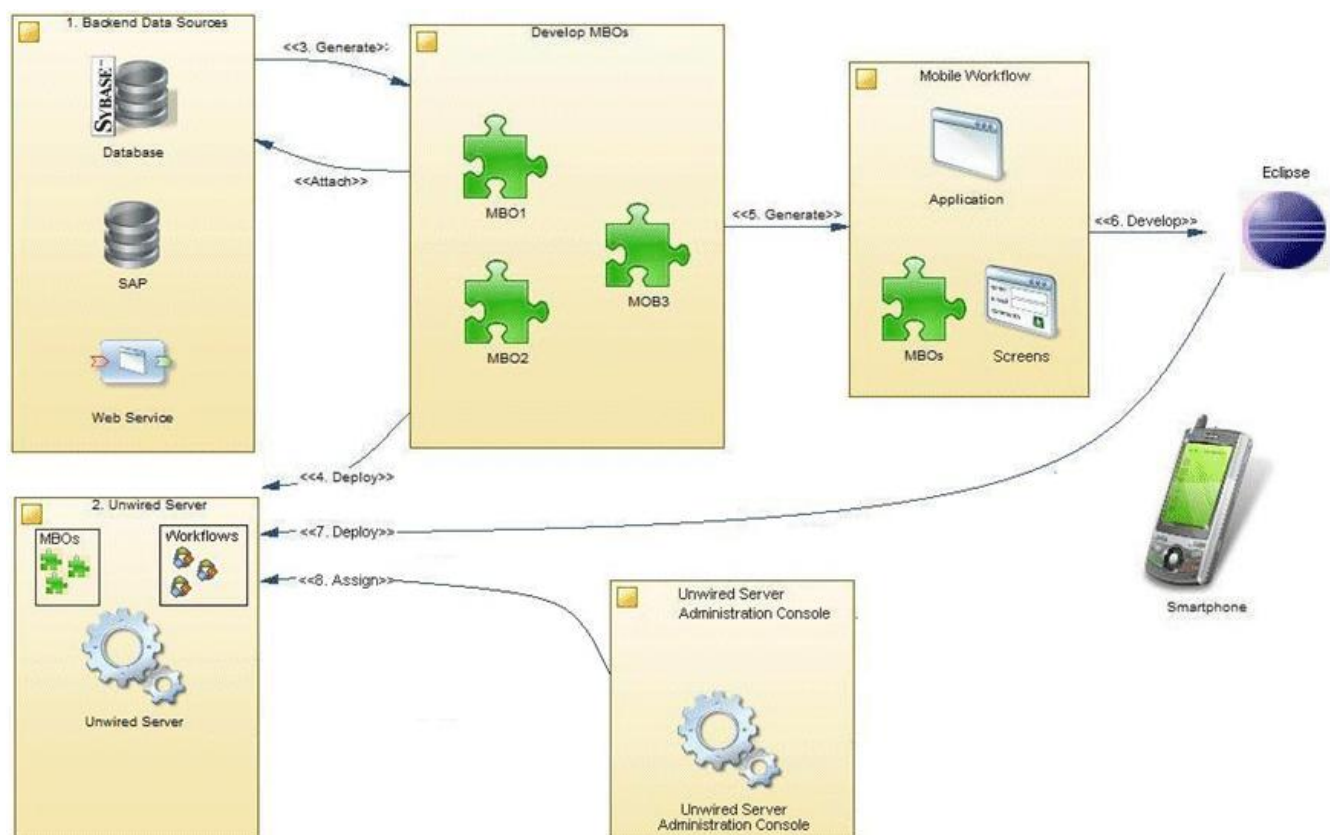
## Native applicatie

Een andere mogelijkheid is om een native applicatie te bouwen. De naam zegt het al, een applicatie die volledig draait op het mobiele apparaat zelf. Hierbij is het dus ook noodzakelijk om de applicatie platform specifiek te bouwen. Er kan dus niet één applicatie gebouwd worden voor zowel Android als IOS.

Wanneer er gekozen wordt voor een native applicatie kun je in de Workspace code laten genereren voor de basis structuur die je hebt gebouwd in de workspace. In de basis zijn alle MBO's en de relaties gedefinieerd, en alle verbindingen die er zijn in de basis applicatie zullen worden verwerkt in de code. Bij het genereren moet je de keuze maken op welk platform je wilt bouwen. Stel er wordt gekozen voor Android zal de Workspace een JAVA package genereren die kan worden gebruikt in het Android project.

Vervolgens kan de ontwikkelaar zelf verder bouwen op de basis. Dit is te vergelijken met het ontwikkelen van een normale mobiele applicatie. Er zijn dus verder geen beperkingen meer vanuit het Sybase platform. Het ontwikkelen van deze applicatie gebeurt ook in de ontwikkelomgeving van het gekozen platform. Een IOS applicatie zal dus worden gebouwd in de IOS SDK.

In de applicatie zelf moet je ervoor zorgen dat het apparaat geregistreerd wordt en verbinding legt met de Unwired Server. De server zorgt er vervolgens voor dat de gegevens gebruikt kunnen worden van het back-end en dat de verbindingen beveiligd zijn.



Figuur 5.5: Mobiele applicatie ontwikkelen in SUP



## 5.7 Welke architectuur keuzes moeten er gemaakt worden?

Sybase Unwired Platform heeft grotendeels al een vastgestelde architectuur. Bij het ontwikkelen van een applicatie moet er gekozen worden of er een native of een workflow applicatie ontwikkeld gaat worden. En welke synchronisatie techniek er gebruikt wordt. Dit is mede afhankelijk van het platform waarop gebouwd gaat worden.

### 5.7.1 Native of Workflow

Als een applicatie ontwikkeld gaat worden moet de ontwikkelaar kiezen of dit een native of een workflow applicatie wordt. Beide vormen hebben voor- en na- delen.

Een workflow applicatie, ook wel hybrid web container-based application genoemd, is geschikt voor voorbereiding of ondersteuning van een bestaand bedrijfsproces of workflows zoals bijvoorbeeld goedkeuringen of aanvragen.

Een native applicatie is geschikt als:

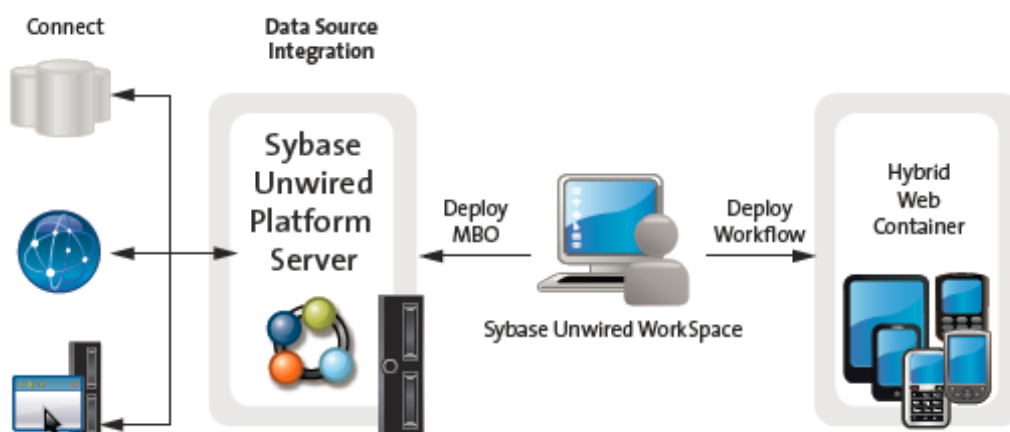
- Ingewikkelde business data nodig is
- Er apparaat functies geïntegreerd zijn in de applicatie(bijvoorbeeld een agenda)
- Gebruikersinterface afhankelijk is van de mogelijkheden van het apparaat
- Geen beperkingen aan de applicatie
- Mogelijkheid om samen te werken met het mobiele besturing systeem.
- Offline functionaliteiten wenselijk zijn

### 5.7.2 Workflow applicatie

In het geval er wordt gekozen om een workflow applicatie te ontwikkelen zal Sybase gebruik maken van de hybrid web container. Dit is een runtime applicatie die op elk device native moet zijn geïnstalleerd. Deze container functioneert als een browsers en hiermee kan dan de workflow applicatie geopend worden. De workflow applicatie is te vergelijken met een web-based applicatie. Er hoeft maar één native container per device te worden geïnstalleerd om meerdere workflow applicaties te kunnen draaien.

Het voordeel om een workflow applicatie te gebruiken is dat het gebruikt kan worden op elk platform, maar er wel controle is wie de applicatie kan gebruiken. Dit wordt namelijk geregistreerd en gecontroleerd in het SCC Daarnaast is het eenvoudig om een update uit te voeren, dit wordt automatisch op alle workflow applicaties toegepast. Hierdoor weet je zeker dat iedereen met de nieuwste versie werkt.

Wanneer een apparaat offline is zal de applicatie nog wel te gebruiken zijn. Alleen alle berichten die de applicatie naar de server stuurt zullen pas verstuurd worden zodra het apparaat weer online komt, Hetzelfde geldt voor de berichten die de server naar het apparaat stuurt. Iets wat niet werkt is het direct opvragen van gegevens uit de database, een zogenaamd online request. Dit zal niet lukken als het mobiele device offline is.



Figuur 5.6: Workflow applicatie

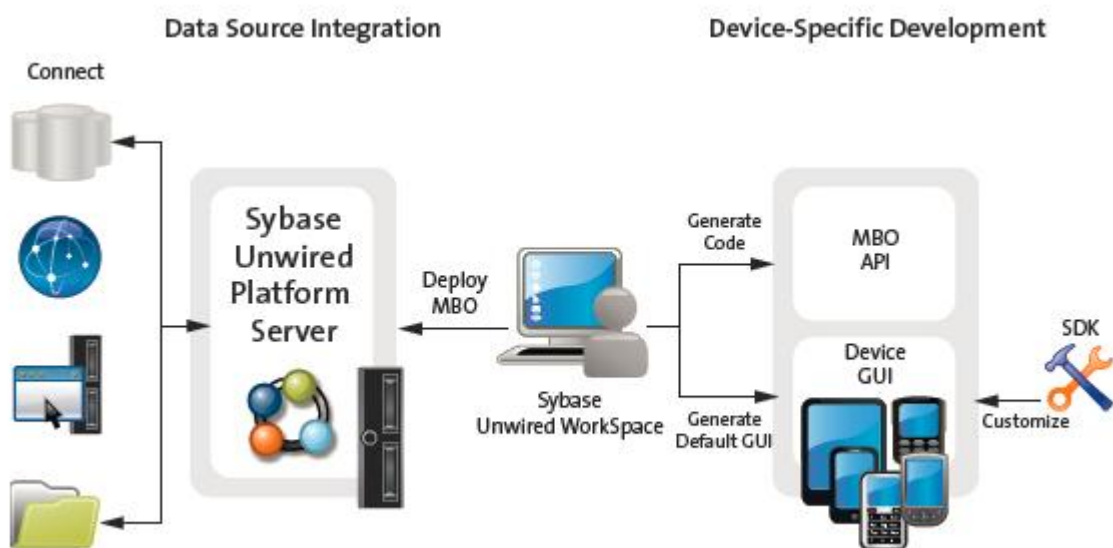
### 5.7.3 Native applicatie

Als er voor een native applicatie wordt gekozen zal dit gecompileerde code zijn die specifiek voor een apparaat gegenereerd is. Er zal dus voor elk platform een andere applicatie ontwikkeld moeten worden. Een native applicatie heeft de meeste flexibiliteit en maakt het beste gebruik van de mogelijkheden van een apparaat. Zo kan er gebruik gemaakt worden van de lokale opslagcapaciteit en de processor om berekeningen te maken. Ook is het mogelijk om bijvoorbeeld een applicatie te laten koppelen met een agenda.

Met een native applicatie is het mogelijk om offline te werken. Omdat de benodigde back-end gegevens al op het apparaat staan kan de applicatie gewoon gebruikt worden. Wel zal de gewijzigde informatie pas op het moment dat het apparaat weer online komt gesynchroniseerd worden.

Voor het ontwikkelen van een applicatie kan de ontwikkelaar wel gewoon kiezen voor de ontwikkelomgeving die bij het platform hoort. Dit is voor IOS bijvoorbeeld XCode, voor Windows mobile is dit Visual studio, en voor Blackberry en Android is dit een plug-in voor Eclipse.

Een nadeel van native applicaties is dat bij een update elk applicatie individueel geüpdate moet worden. En er dus met verschillende versies gewerkt kan worden.



Figuur 5.7: Native applicatie

### 5.7.4 Synchronisatie

Synchronisatie is een belangrijk onderdeel van Sybase Unwired platform. Het zorgt ervoor dat de mobiele applicatie kan werken met recente gegevens uit de database van het back-end systeem. Maar moet dusdanig efficiënt zijn dat dit niet ten koste gaat van de performance van het back-end systeem. Het over en weer sturen van data gebeurt via het HTTP protocol.

Er zijn twee manieren waarop de applicatie kan synchroniseren met database van Sybase. Welke optie dit wordt is afhankelijk van de applicatie die gebouwd wordt. De keuze van het soort applicatie en het platform is dus bepalend voor de manier van synchroniseren.

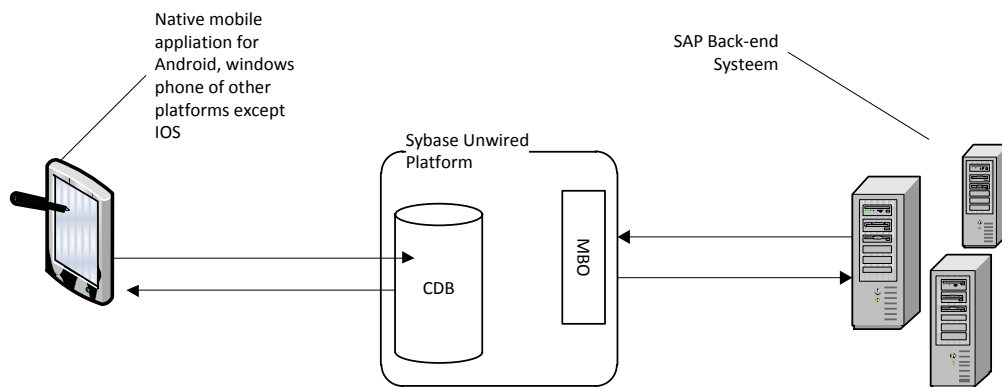
- Replication based
- Message based

### Replication Based

Replication based houdt in dat er een replicatie van de benodigde database gegevens in de Consolidated Database (CDB) van het de Sybase server staan. En dat de gegevens vanuit het CDB naar het device gestuurd wordt. Het CDB kan dan on demand of scheduled synchroniseren met de back-end server. Het CDB zal alleen gegevens synchroniseren die nodig zijn voor de mobiele applicaties. Dit wordt aangegeven bij de initiële synchronisatie vanuit de mobiele applicatie. Hierna doet het platform de rest van het werk.

Een voordeel hiervan is dat er minder connecties hoeven worden gelegd met de back-end server en zal deze dus minder belast worden. Een nadeel is dat het voor kan komen dat er iets verouderde data opgevraagd wordt omdat de data niet altijd direct uit het back-end systeem komt maar van de SUP server.

Replication Based Synchronisatie wordt gebruikt bij Native applicaties op elk platform behalve IOS. Bij Replication Based Synchronisatie heb je dus het nadeel dat wanneer twee mensen tegelijkertijd data aanpassen het platform er niet mee om kan gaan.

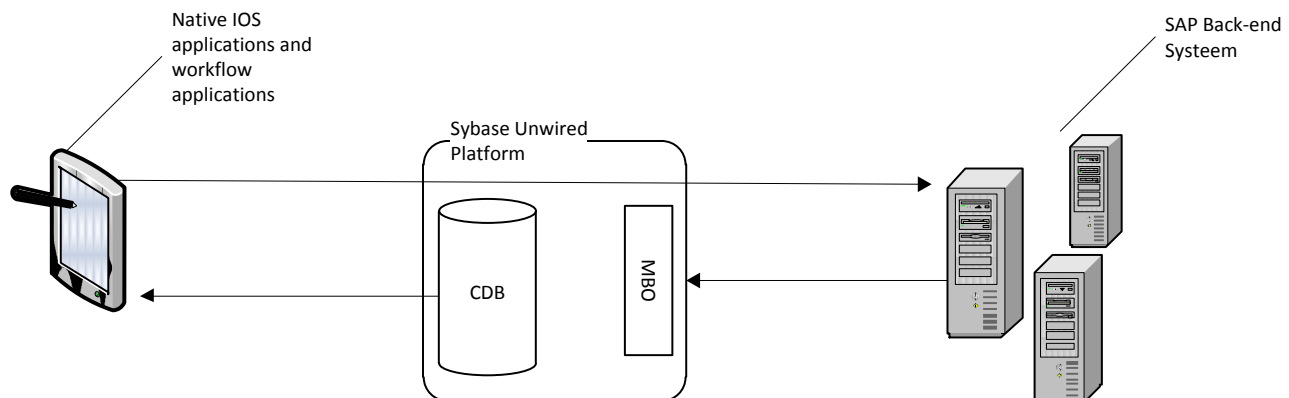


Figuur 5.8: Replication Based Synchronisation

### Message Based

Message based wordt gebruikt bij workflow applicaties en IOS. Dit houdt in dat bij elke aanvraag de data direct uit het back-end wordt opgevraagd en gepusht wordt naar het apparaat.

Voordeel is dat je zo goed als zeker weet dat je met de meest recente data werkt. Een nadeel is dan weer wel dat er veel aanvragen gedaan worden op het back-end systeem.



Figuur 5.9 Message Based

## 5.8 Conclusie

Sybase Unwired Platform is een Mobile Enterprise Application Platform. Dit is een pakket van producten en services die de ontwikkeling van mobiele applicaties mogelijk maken.

Over het algemeen wordt een platform een MEAP genoemd wanneer het meer dan drie applicaties ondersteunt, meer dan drie back-end systemen ondersteunt of meer dan drie mobiele platformen ondersteunt. Sybase is dus geschikt voor grote bedrijven en past daarom goed in de klantenkring van Capgemini.

De voordelen van MEAP zijn de centrale toegang, waardoor de beveiliging en management eenvoudiger wordt. Het hergebruik van code, waardoor er minder geprogrammeerd hoeft te worden en de ontwikkeling sneller gaat. De Vermindering van dataverkeer tussen back-end en mobiel apparaat waardoor het overbelasten van de back-end voorkomen wordt. De mogelijkheid om snel eenvoudige multiplatform workflow applicaties te ontwikkelen. Wanneer er goed nagedacht wordt over de applicatie die het bedrijf wil ontwikkelen en de functionaliteiten die erin komen kunnen zeer nuttige applicaties gebouwd worden.

Daarnaast zitten er een aantal nadelen aan het gebruik van MEAP. Deze risico's van deze nadelen zijn goed te beperken. Als je weet wat de nadelen zijn kun je er rekening mee houden tijdens het ontwikkelen en gebruiken van de applicaties. De kosten voor het implementeren van dit platform zijn vrij hoog. Het kost een bedrijf ook nog eens gemiddeld twee extra mensen fulltime in dienst om het platform te beheren en onderhouden. Het is dan ook aan te raden om minimaal 150 mensen in dienst te hebben of de verwachting om deze in dienst te krijgen. Bij dit aantal medewerkers is MEAP goedkoper dan alternatieven.

Sybase is onderdeel van SAP, hierdoor is het Unwired Platform zeer geschikt om applicaties te ontwikkelen die gekoppeld worden met een SAP back-end. SUP is dus ook een zeer geschikt platform om tijdens dit project te gebruiken. Welke applicaties er gebouwd kunnen worden is deels al vastgelegd. Dit komt omdat er gebruik gemaakt wordt van BAPI's. In SAP is al een set van te gebruiken BAPI's gedefinieerd en het is zonder specifieke ABAP kennis bijna onmogelijk om in korte tijd zelf BAPI's te bouwen.

Applicaties voor SUP kun je bouwen in de Workspace. Dit is een software development kit gebaseerd op de Eclipse omgeving voor JAVA. Deze omgeving is voor veel ontwikkelaars een bekend. In deze Workspace is het mogelijk om de basis van de applicatie te ontwikkelen. Hierin zijn de Mobile Business Objecten gemaakt en de relaties ertussen gedefinieerd. Wanneer er een native applicatie ontwikkeld wordt kan de Workspace hiervan code genereren voor een specifiek mobiel besturingssysteem. Dit kan bijvoorbeeld voor JAVA zijn voor Android of C# voor Windows Mobile. Vervolgens is het mogelijk om vanuit deze code een native applicatie te ontwikkelen in de ontwikkel omgevingen van het mobiele besturingssysteem. Er kunnen ook workflow applicaties ontwikkeld worden. Dit zijn web-based applicaties en kunnen direct in de workspace ontwikkeld worden. Deze applicaties zijn wat beperkt aan functionaliteiten maar relatief snel te ontwikkelen. Bovendien zijn ze te gebruiken op elk mobiel besturingssysteem.

Om een applicatie te ontwikkelen met Sybase Unwired Platform moeten er architectuur keuzes gemaakt worden. Er moet bijvoorbeeld gekozen worden of er een workflow of native applicatie ontwikkeld word en op welk platform. Met deze keuze bepaal je ook direct de manier van synchroniseren. De keuze voor native of workflow wordt grotendeels bepaald door de gewenste functionaliteiten die in de applicatie moeten komen.

Er moeten ook keuzes gemaakt worden tijdens het installeren van het platform. Dit is bijvoorbeeld of er een fysieke server komt of dat er een server gehost wordt bij SAP. Ook is er de mogelijkheid om een clustered, platform met meerdere gekoppelde servers, te nemen of non clustered, met één fysieke server. Deze keuze hangt af van het aantal gebruikers dat met het platform moet gaan werken. Omdat in dit project vooral gekeken wordt naar het ontwikkelen van een applicatie zullen deze keuzes niet van invloed zijn op het project.

Na dit onderzoek ben ik er achter gekomen dat het ontwikkelen van een applicatie minder eenvoudig is dan Sybase aangeeft in de meeste documentatie. Het ontwikkelen van een workflow applicatie is eenvoudig, alleen is het beperkte aantal functionaliteiten vaak genoeg reden om voor een native applicatie te kiezen. Bij het ontwikkelen van een native applicatie heb je veel SUP en programmeer kennis nodig. Bovendien kost dit vaak meer tijd omdat er voor elk mobiel besturingssysteem een aparte applicatie ontwikkeld moet worden. Wel gaat de tweede applicatie waarschijnlijk sneller omdat je veel code kunt hergebruiken.

## 6. Inspectie management

De applicatie die ik ga ontwikkelen heeft betrekking op het inspectie management proces. Voordat ik deze applicatie ontwikkeld heb, ben eerst gaan onderzoeken wat dit proces precies inhoud.

Inspectie management wil zeggen het managen van inspecties en/of onderhoud werkzaamheden van apparaten in het veld. Dit kunnen bijvoorbeeld koffie automaten zijn van Douwe Egberts die verspreid staan bij hun klanten. Maar ook pompen die olie rondpompen bij fabrieken van Shell. Deze machines worden in SAP termen equipment genoemd. De kennis die ik gebruikt heb voor dit onderzoek heb ik vooral vergaard door gesprekken met interne experts. Ook heb ik via deze experts toegang gekregen tot documenten die het proces beschrijven.

Een aantal inspectiepunten die veel voorkomen zijn:

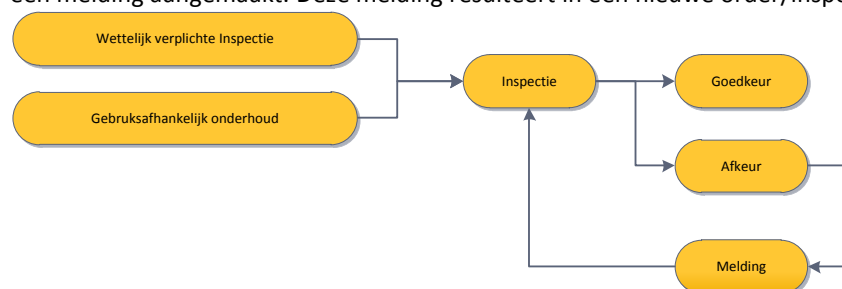
- Trillingsmetingen (detecteert lager slijtage)
- Ultrasoon onderzoek (detecteert scheurtjes in velgen)
- Smeerolie analyse (detecteert veroudering smeerolie)
- Gas chromatografie (duidt lekkage nucleaire installatie aan)
- Visuele inspectie (controleren op olielekage)
- Procesgegevens controleren (afwijkende temperatuur kan slijtage aanduiden)
- Dye checking (toont scheurtjes in metaaloppervlak aan)
- Röntgenonderzoek (detecteert fouten in lassen)
- Meggering (detecteert gebrekkige elektrische isolatie)

Er zijn twee soorten inspecties te onderscheiden, preventief en correctief. Correctieve inspecties worden uitgevoerd nadat er is geconstateerd dat een equipment defect is. Preventieve inspecties worden op basis van tijd of gebruik uitgevoerd en moeten voorkomen dat het equipment defect raakt. In dit project is vooral de focus gelegd op preventieve inspecties.

Er zijn twee redenen te beschrijven waarom een bedrijf preventieve inspecties zou willen uitvoeren en registreren. De eerste is omdat dit wettelijk verplicht is. Bedrijven die bijvoorbeeld met chemische stoffen werken moeten kunnen aantonen dat zij om de zoveel tijd inspecties uitvoeren op het equipment om te voorkomen dat er ongevallen plaatsvinden met chemicaliën. De tweede reden is vanuit bedrijfseconomisch oogpunt. Bedrijven doen hierbij preventief onderhoud zodat zij tijdig gebreken aan het equipment kunnen signaleren. Dit voorkomt dat het equipment op den duur helemaal stil komt te liggen. Hierbij is het belangrijk om te bepalen wanneer er inspecties uitgevoerd moeten worden. Als er te vaak geïnspecteerd wordt kost dit onnodig veel geld, en als er te weinig geïnspecteerd wordt kan dit veel geld kosten doordat het equipment onbruikbaar wordt.

Om deze inspecties op tijd uit te voeren wordt er een zogenaamd inspectie plan opgesteld. Hierin staan alle eigenschappen van de inspectie en ook hoe vaak deze uitgevoerd moet worden. Dit kan bijvoorbeeld gebeuren op basis van tijd(bijvoorbeeld elke maand), en op basis van gebruik(bijvoorbeeld na 1000 kopjes koffie). Wanneer er een inspectie gepland staat wordt er automatisch vanuit de SAP Plant Maintenance module een werkorder aangemaakt. Vanuit een werkorder wordt een Inspection Lot gecreëerd en hierin staan de meetpunten en taken voor de inspectie. Voor het uitvoeren van een inspectie is daarom voor de inspecteur ook alleen het Inspection Lot van belang. Een werkorder wordt vooral gebruikt voor het registreren van gewerkte uren, gebruikt materiaal etc.

Met het uitvoeren van een inspectie bedoel ik het verrichten van metingen en registreren van de resultaten. Aan de hand van deze resultaten wordt er besloten of het equipment goed of afgekeurd is. Zodra een equipment afgekeurd is, wordt er een melding aangemaakt. Deze melding resulteert in een nieuwe order/inspectie.



Figuur 6.1: Inspectie management overview

Inspectie management kan op verschillende manieren worden geïmplementeerd in SAP systemen. Drie veel voorkomende processen zijn:

- **Condition Monitoring**

Dit houdt in het uitvoeren van preventieve inspecties aan de hand van wettelijke verplichtingen of gebruiksafhankelijke inspecties. Bij het condition monitoring registreert de inspecteur de resultaten van de inspecties. Zodra deze inspectieresultaten binnen de norm vallen schrijft de inspecteur zijn uren en gebeurt er verder niets.

Wanneer de resultaten buiten de norm vallen moet er een usage decision worden gemaakt. Dit houdt in dat er een vervolgstap moet komen op de gemeten resultaten. Meestal resulteert dit in een melding waarin het defect beschreven wordt. Er moet dan opnieuw een inspecteur heen op dit defect te repareren.

- **Repair Maintenance**

Repair maintenance treedt op wanneer er een defect is geconstateerd op een equipment. Dit kan gebeuren door een gebruiker of door een preventieve inspectie(condition monitoring). Er is een melding gegenereerd met een beschrijving van het defect. Deze melding is geaccordeerd door een manager.

De inspecteur gaat naar het equipment en repareert het defect. Hierna voert hij weer een inspectie uit om te testen of het equipment daadwerkelijk gerepareerd is en of de inspectie resultaten nu wel binnen de norm vallen.

- **Calibration Inspection.**

Dit is een combinatie van de twee bovenstaande processen. Hierbij inspecteert de inspecteur het equipment op basis van wettelijke verplichting of gebruiksafhankelijk. Wanneer de resultaten binnen de norm vallen gebeurt er niets. Wanneer de resultaten erbuiten vallen zal de inspecteur direct het equipment repareren zodat de resultaten weer binnen de norm vallen.

In veel gevallen is dit proces onwenselijk omdat hierbij de beslissing om het defect te repareren alleen aan de inspecteur overgelaten wordt. Het komt regelmatig voor dat er een aantal kleine defecten opgespaard worden om hierna één inspecteur alles te laten repareren. Daarnaast moet er vaak goedkeuring gegeven worden door de klant om de werkzaamheden te verrichten aan het equipment.

Ik heb onderzoek gedaan naar inspectie management in SAP omdat ik voor dit proces een applicatie heb ontwikkeld. Voor dit onderzoek heb ik gebruik gemaakt van de kennis van interne experts bij Capgemini. Zo heb ik in het begin samen met verschillende experts het hele proces doorlopen op papier en aan de hand van online trainingen. Op deze manier kom veel te weten over het proces. Daarnaast kun je alle vragen direct stellen aan de experts. Ik ben zelfstandig op zoek gegaan naar experts binnen Capgemini en mensen die mij kunnen helpen met het beschrijven van het inspectie proces. Zoals al eerder beschreven is het van belang om op tijd goede afspraken te maken met de experts omdat zij vrijwel altijd ingezet zijn bij klanten.

Omdat ik het proces met verschillende experts besproken heb, heb ik het proces ook vanuit verschillende inzichten gezien en gehoord. Hieruit heb ik vastgesteld het beschrijven van één soort proces heel lastig is in SAP. Er zijn veel verschillende manieren van implementeren en iedere implementatie is wel weer anders. Het is dus van belang dat bij de ontwikkeling van de applicatie er goed nagedacht wordt over hoe het proces zo generiek mogelijk blijft.

Door een zo generiek mogelijk proces te ontwerpen kan de demo applicatie aan meer klanten getoond worden. Het proces is dan voor alle klanten duidelijk zodat het nut en de voordelen van de applicatie beter overgebracht kunnen worden.



## 7. SAP back-end systeem

Om een koppeling te realiseren met een back-end is er wel eerst een back-end systeem nodig. Het enige dat vast stond is dat het een SAP systeem moest zijn. Omdat van te voren niet vast stond welk SAP systeem gebruikt kan worden heb ik dit zelf onderzocht en een systeem gekozen. Omdat SAP een groot en complex pakket is zal ik eerst wat beschrijven over SAP in het algemeen en vervolgens hoe ik tot een keuze ben gekomen.

### 7.1 SAP

SAP is een Duits softwarebedrijf dat is opgericht in 1972 door een aantal ex IBM medewerkers. SAP staat voor Systeme Anwendungen und Produkte in der datenverarbeitung(systemen, applicaties en producten in gegevensverwerking). Oorspronkelijk bij de oprichting stond het voor SystemAnalyse und Programmentwicklung.

SAP staat vooral bekend om het maken van enterprise software voor het beheren van bedrijfsprocessen en klantrelaties. De bekendste pakketen zijn Enterprise Resource Planning(SAP ERP) en Customer Relationship Management ( SAP CRM) en nieuwe ontwikkelingen zijn Mobiele oplossingen (Sybase) en in memory computing (SAP HANA)



Figuur 7.1:SAP

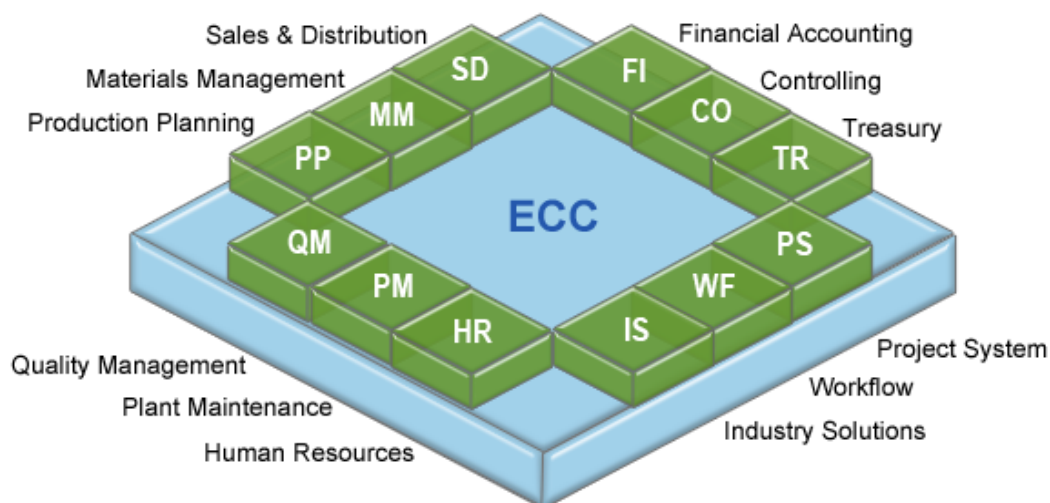
### 7.2 SAP systemen

Het belangrijkste systeem van SAP is Enterprise Resource Planning(ERP). Dit is het hoofd onderdeel van SAP ECC. Op SAP ECC kunnen eventueel ook nog andere systemen gekoppeld worden.

De vijf grootste onderdelen van SAP zijn:

- Enterprise Resource Planning – ERP
- Customer Relationship Management – CRM
- Product Lifecycle Management – PLM
- Supply Chain Management – SCM
- Supplier Relationship Management – SRM

Deze systemen bestaan weer kleinere modules. Hieronder een overzicht van de verschillende modules die er beschikbaar zijn in SAP. Voor dit project zijn vooral Plant Maintenance en Quality Management interessant omdat hier het inspectie management in is ingericht



Figuur 7.2: SAP modules

## 7.3 Back-end systeem keuze

Omdat van te voren niet vast stond welke back-end systeem ik kon gebruiken voor dit project heb ik dit zelf moeten onderzoeken en kiezen. Capgemini implementeert verschillende SAP systemen bij klanten en ik moet een keuze maken met welk systeem ik de koppeling in mijn inspectie management applicatie ga maken.

### Systeemeisen

Om een goede keuze te kunnen maken heb ik eerst een lijstje met systeemeisen opgesteld. Hier moest het systeem aan voldoen voordat ik er een koppeling mee kon maken.

Eisen van het systeem:

- Er moeten inspecties uitgevoerd en geregistreerd kunnen worden in het systeem. Hiervoor moet het systeem over een Quality Management module en Plant Maintenance module beschikken. In deze modules wordt het hele inspectie proces afgehandeld en dit heb ik nodig voor de applicatie.
- Het systeem moet direct beschikbaar zijn. Omdat het project volgens een strakke planning loopt moet het systeem direct beschikbaar zijn. Er is te weinig tijd om hier op te wachten.
- Het systeem moet ingericht zijn. Dit houdt in dat je het gehele inspectie proces in het systeem kunt doorlopen. En dat alle benodigde data en objecten beschikbaar is. Er moeten bijvoorbeeld functieplaatsen en equipments zijn aangemaakt.

### SAP CRM

Na gesprekken met de experts bleek dat het SAP CRM een geschikt systeem is. Dit systeem heeft geschikte Quality Management en Plant Maintenance functionaliteiten. Binnen Capgemini is er bovendien veel kennis van CRM. Het enige probleem was dat ik het systeem nog niet direct kon gaan gebruiken omdat het nog niet volledig is ingericht voor inspectie management.

Omdat er voor zover ik wist weinig alternatieven waren ben ik gaan kijken hoe ik het systeem kon gaan inrichten. Bij het inrichten van het systeem is specifieke inspectie management kennis nodig. Daarom ben ik op zoek gegaan naar mensen binnen Capgemini die mij hiermee kunnen helpen. Ik heb een consultant gevonden die wel kennis heeft op het gebied van CRM maar niet over Inspectie management.

Toen ik wilde beginnen met inrichten kwam ik erachter dat het CRM systeem niet meer beschikbaar was. Nadat ik op onderzoek was gegaan naar de oorzaak bleek het dat de server offline was gegaan. In de zomer van 2012 heeft Capgemini op de planning staan om alle SAP systemen te vervangen voor nieuwe systemen die gehost worden in India. Hierdoor werd er geen onderhoud meer gedaan aan het CRM systeem en was het pas aan het einde van het project weer beschikking over het SAP CRM systeem. Helaas kon ik dus geen gebruik meer maken van het SAP CRM systeem.

### SAP ERP

Omdat dit voor meerdere demosystemen het geval was heb ik besloten om eerst te inventariseren welke demosystemen er wel beschikbaar zijn. En vervolgens te kijken wel systemen geschikt zijn voor de koppeling met de applicatie. Op basis van dit systeem ga ik vervolgens de applicatie ontwerpen. Mogelijk geschikte systemen waren alleen SAP ERP en SAP SRM.

Na enig onderzoek is gebleken dat het Enterprise Resource Planning(ERP) systeem van SAP, een geschikt back-end systeem is waarvan er direct een demo systeem beschikbaar was. In dit systeem is het ook mogelijk om inspecties uit te voeren en te registreren via de Quality Management module en de Plant Maintenance module. Helaas was ook dit systeem nog niet ingericht voor het inspectie proces.

Een ander bijkomend probleem was dat op het demo systeem elke maand een nieuwe image word geladen. Dit houdt in dat alle aanpassingen die aan het systeem worden gedaan, na een maand weer weg zijn. Hierdoor is het onmogelijk om een goed inspectie proces in te richten op het systeem, en deze gedurende het hele project in stand te houden.

Om dit probleem op te lossen ben ik op zoek gegaan naar een ander ERP systeem. Via een collega heb ik in India een aantal mensen gesproken en via hen toegang gevraagd voor een alternatief ERP demosysteem dat in India draait. Dit systeem wordt alleen gebruikt voor demo's bij klanten.

Een voorwaarde voor gebruik van dit demosysteem is dan ook dat er goed gedocumenteerd wordt welke aanpassingen er worden gedaan. Ook moet het project opgeleverd worden met een presentatie over hoe dit verkocht kan worden aan klanten.

## 7.4 Inrichten van het back-end systeem

Nadat er een geschikt back-end systeem was gevonden moest het systeem nog ingericht worden. Deze keer wilde ik een expert vinden die wel verstand had van het Inspectie proces omdat dit essentieel is voor het inrichten van het proces in SAP. Bovendien gaat het inrichten hierdoor waarschijnlijk wat sneller.

Inspectie management is een relatief nieuw proces binnen de ICT en er is hier nog maar weinig kennis over binnen Capgemini. De mensen die wel kennis hebben van inspecties zijn vaak al ingezet bij klanten, en dus niet veel beschikbaar voor dit project. Het project is afhankelijk van de bereidheid van professionals die willen helpen om het proces in te richten op de demo. Daarnaast kan het zijn dat een professional twee weken beschikbaar is en daarna weer ingezet wordt.

Dit betekent dat ik continu zal moeten blijven zoeken naar professionals die kennis hebben van Inspectie management, en vragen of zij mij willen helpen. Dit betekent dat ik een groot netwerk zal moeten creëren met mensen in dit vakgebied. Hier ligt naast een groot risico ook wel een uitdaging voor het project.

Uiteindelijk is het gelukt om een aantal experts te vinden die konden helpen met het inrichten van het proces in de back-end. Zij hadden ook een ingericht systeem nodig voor een demo bij een klant. Op deze manier hadden de experts een goede reden om tijd te steken in het inrichten van dit proces en kon ik mooi gebruik maken van het ingerichte systeem.

Omdat ik aan het afstuderen was in een vakantie periode en de experts druk waren met andere projecten heeft het inrichten van de back-end langer geduurd dan gepland. Hierdoor heeft het vrij lang geduurd voordat ik het hele inspectie proces kon doorlopen in de SAP back-end.

Door gebrek aan kennis heb ik zelf niet heel veel kunnen toevoegen aan de inrichting van het back-end systeem. Daarvoor is veel meer specifieke kennis en ervaring nodig. Wel heb ik hier veel van geleerd omdat ik stap voor stap gezien heb welke objecten en gegevens nodig waren om dit proces te kunnen doorlopen.

Achteraf gezien ben ik blij dat ik al vrij vroeg in het project onderzoek heb gedaan naar een geschikt back-end systeem. Wanneer ik later was begonnen hiermee had ik nooit op tijd een geschikt back-end kunnen vinden, en was er waarschijnlijk niet veel van de opdracht terecht gekomen.

Nadat het systeem ingericht was heb ik de experts meteen betrokken bij het definiëren van de requirements.

## 8. Definitie van requirements

Om de requirements van dit project te achterhalen heb ik een aantal experts binnen Capgemini benaderd. Deze konden mij vertellen wat de behoefte op de markt is, en waar nog kansen liggen. Naast experts op het gebied van inspectie management heb ik ook experts op het gebied van mobility benaderd. Dit omdat ik dan rekening kan worden gehouden met de visie van Capgemini op het gebied van mobility. Bovendien zijn er al vaker mobiele applicaties ontwikkeld door Capgemini. Deze kennis kon ik goed gebruiken voor mijn project. Binnen Capgemini is er nog nooit een applicatie ontwikkeld met het Sybase Unwired Platform en een koppeling met een bestaand SAP systeem. Deze kennis moest ik zelf dus vergaren.

Er zijn diverse gesprekken geweest met verschillende experts bij Capgemini. Uit deze gesprekken is eerst de algemene kennis van inspectie management vergaard, en geprobeerd te begrijpen wat het proces inhoudt. Dit staat beschreven in hoofdstuk zes. Bij het definiëren van de requirements ben ik eerst gaan onderzoeken uit welke stappen het proces precies bestaat en vervolgens welke van deze stappen geschikt zijn om te verwerken in een mobiele applicatie.

### 8.1 Mobiel inspectie management

Toen ik meer wist over het inspectie proces in SAP ben ik gaan nadenken over hoe dit proces kan worden ondersteund door middel van een mobiele applicatie.

In het hele proces zijn drie taken te onderscheiden. De eerste taak is het genereren en accepteren van meldingen, en het creëren van orders voor het uitvoeren van inspecties. De tweede taak is het uitvoeren van de inspecties en het registreren van de resultaten. De derde taak is het bekijken van de inspectie resultaten en besluiten of een equipment is goed of afgekeurd. Elke taak wordt normaliter door een ander persoon uitgevoerd. Bij elk van deze overgangspunten kan er een vertraging ontstaan in het proces omdat die betreffende persoon niet achter de computer zit op dat moment. Hier zie ik kansen om het proces efficiënter te maken.



**Figuur 8.1: Mobiele inspectie applicaties**

Om het Inspectie Management proces te ondersteunen kunnen er het beste drie verschillende applicaties gebouwd worden. Elk geschikt voor een taak. Zodat het altijd en overal mogelijk moet kunnen zijn om een melding te accepteren of een inspectie te kunnen goedkeuren zodat de inspecteur snel zijn verder kan met het uitvoeren van de inspecties.

Omdat er beperkte tijd is in dit project kan er slechts een demo gebouwd worden voor één van de drie applicaties. Ik heb gekozen voor het registreren van inspectie resultaten. Dit lijkt mij namelijk het meest interessante onderdeel van het proces waarbij de voordelen van mobiel het beste naar voren komen. De inspecteur is vaak in het veld bezig en veel dubbel werkt heeft wanneer de resultaten ook nog in het systeem achter een computer ingevuld moeten worden.

Zoals ik al eerder beschreven heb kan het uitvoeren van Inspecties op drie verschillende manieren worden geïmplementeerd.

- Condition Monitoring
- Repair Maintenance
- Calibration Inspection.

Voor elk van deze processen gelden verschillende stappen. Om deze processtappen duidelijk te krijgen heb ik samen met een aantal experts elke stap besproken en beschreven op papier. In deze lijst heb ik beschreven wat de stap precies inhoudt, wie het doet, wat eventueel een mobiele toepassing kan zijn hiervoor, welk SAP object hierbij gebruikt wordt en welke status het object op dat moment heeft. Deze lijst met processtappen heb ik vervolgens met verschillende experts besproken.

Deze lijst kun je ook zien als een lijst met requirements voor het inspectie management in SAP. Het verschil is dat dit de requirements zijn voor het inspectie proces zoals deze op een computer wordt uitgevoerd. Waar ik nog achter moet komen is welke requirements geschikt zijn voor een mobiele applicatie.

Om de requirements vervolgens concreet en duidelijk te maken heb ik ze getekend in een proces flow. Hierdoor is duidelijk en overzichtelijk geworden hoe het proces in elkaar zit en welke rollen er zijn. Dit heb ik voor alle drie de verschillende implementaties gedaan. Ook deze proces flows heb ik weer met een aantal experts besproken.

Door de gesprekken met experts bleek dat de behoefte voor een mobiele applicatie vooral lag bij het condition monitoring proces. Collega's in het veld zien de meeste toegevoegde waarde voor een mobiele applicatie waarbij het invoeren van resultaten mogelijk is. Daarnaast was er een project in zicht waarbij een mobiele applicatie voor condition monitoring interessant kan zijn voor Capgemini. Wanneer ik een demo applicatie zou laten zien aan de klant kwam er misschien een mogelijkheid om de applicatie verder te bouwen voor een echt project.

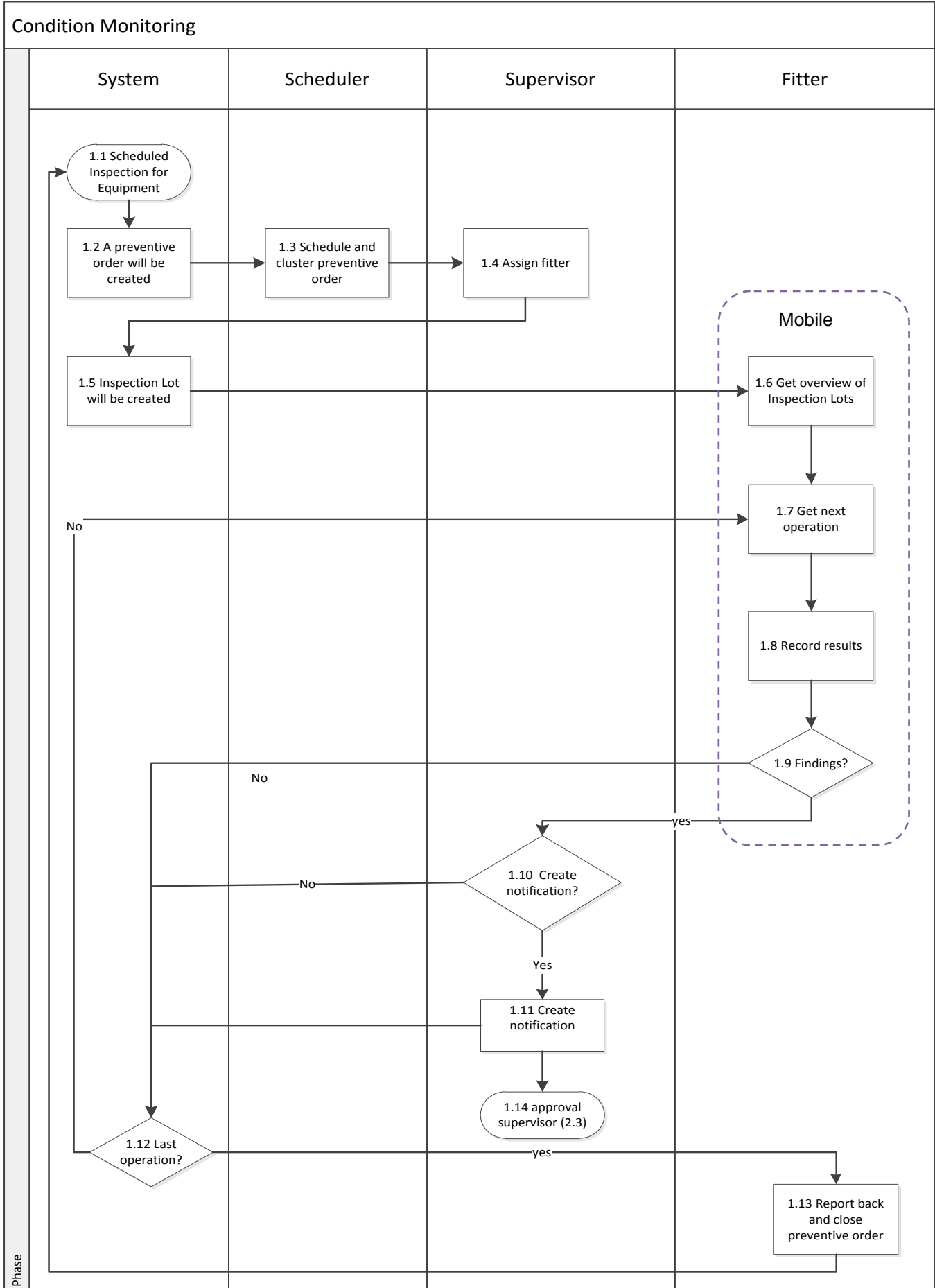
Om een applicatie te ontwikkelen voor het condition monitoring ben ik eerst verder ingegaan op de details en de mobiele mogelijkheden voor dit proces. Dit heb ik gedaan door een proces flow te maken. Hierbij ben ik tegen een aantal moeilijke punten aangelopen. Zo kan het bij sommige implementaties verschillen wie een bepaalde stap uitvoert. Bijvoorbeeld de keuze om een melding van de gemeten resultaten te maken. In sommige gevallen gebeurt dit door de inspecteur zelf en in andere gevallen doet de supervisor dit. Het ligt aan het bedrijf wie deze keuze maakt. Ik heb er in mijn proces voor gekozen om de supervisor deze keuze te laten maken. Dit heb ik gedaan omdat ik denk dat wanneer de inspecteur dit doet, hij bijvoorbeeld snel geneigd is om het resultaat iets te veranderen zodat er geen melding gemaakt hoeft te worden. Wanneer een supervisor dit doet heeft de inspecteur er geen invloed op eventuele vervolg werkzaamheden.

## Procesflow

Bij condition monitoring proces zijn 4 verschillende actoren te onderscheiden

- **Het systeem(System)**  
Het systeem verwerkt alle informatie, en kan ook automatisch objecten genereren. Zo wordt er bijvoorbeeld automatisch een inspectie lot gegenereerd wanneer de order is voorbereid.
- **De planner(Scheduler)**  
De planner bereidt het werk voor, vult in wat er moet gebeuren welke materialen er nodig zijn en hoeveel tijd het gaat kosten. Ook wanneer de order uitgevoerd moet worden.
- **De supervisor(Supervisor)**  
De supervisor koppelt een inspecteur aan een order voordat deze uitgevoerd wordt. Daarnaast neemt de supervisor het besluit om eventueel een melding te maken naar aanleiding van de gemeten resultaten.
- **De inspecteur(Fitter)**  
De inspecteur is de enige persoon die echt in het veld loopt en de inspecties ook daadwerkelijk uitvoert. De inspecteur haalt de Inspection Lots op die aan hem gekoppeld zijn, voert de inspecties uit en registreert de resultaten. Uiteindelijk moet hij de order terugmelden en registreren hoe lang hij erover gedaan heeft en welke materialen hij gebruikt heeft.

Een gedetailleerde beschrijving van het inspectie proces aan de hand van een proces flow:





## 8.2 De applicatie

Nadat het totale proces duidelijk was heb ik gekeken welke stappen efficiënter kunnen worden door deze uit te voeren op een mobiel apparaat. Een proces stap is geschikt op het moment dat er tijd winst mee behaald kan worden, of dat er fouten mee voorkomen kunnen worden.

Er kan tijd winst gehaald worden en er kunnen fouten voorkomen worden door de resultaten van de inspecties direct in te voeren. Ook het direct ophalen van de uit te voeren inspecties en het afmelden hiervan moet efficiënter kunnen. Bijna alle stappen waarvoor de inspecteur nu nog vanuit het veld terug naar z'n werkplek moet voor het invoeren en ophalen van informatie kunnen efficiënter worden door een mobiele applicatie. Daarom heb ik besloten dat de proces stappen van de inspecteur het meest geschikt zijn voor mijn mobiele applicatie.

In eerste instantie had ik gekozen om alle mogelijke functionaliteiten voor de inspecteur beschikbaar te maken in de applicatie, zoals bijvoorbeeld het terugmelden van een order, het invullen van gewerkte uren en het invullen van gebruikt materiaal. Maar tijdens het onderzoek naar Sybase Unwired Platform en het gereed maken van de back-end liep ik tegen veel problemen op, dit heeft dan ook langer geduurd als gepland. Tijdens het onderzoek ben ik er ook erachter gekomen dat het ontwikkelen van een applicatie ingewikkelder is dan ik van te voren dacht. Om deze redenen heb ik besloten om het aantal functionaliteiten in te perken.

Ik zal alleen de functionaliteiten bouwen die de inspecteur nodig heeft tijdens het registreren van de meet resultaten. Proces stappen die de inspecteur moet doen in het proces zijn: het opvragen van Inspection Lots die aan hem gekoppeld zijn, het uitvoeren van de metingen en het registreren van de resultaten. Aan de hand van het proces flow en de proces beschrijving heb ik daarom een aantal stories beschreven in het product backlog.

Uiteindelijk de volgende functionele stories voor de te ontwikkelen applicatie ontstaan:

- Inloggen
- Ophalen van inspecties
- Ophalen van uit te voeren taken
- Resultaten van verrichte metingen invoeren
- Resultaten van voorgaande metingen bekijken

Met deze functionaliteiten is het voor de inspecteur mogelijk om de applicatie te gebruiken voor het inspecteren van het equipment. Deze applicatie verschaft klanten en Capgemini een goed beeld van de mogelijkheden van Inspectie management met Sybase Unwired Platform en een koppeling naar SAP back-end systeem

De functionaliteiten van de applicatie heb ik verder uitgeschreven in use cases, zodat zowel voor mij als de opdrachtgever duidelijk is wat deze functionaliteiten precies inhouden. Deze functionaliteiten heb ik ook gevalideerd met de verschillende experts binnen Capgemini.

Naast de functionele requirements zijn er ook een aantal technische requirements opgesteld (techstories) dit zijn de requirements waar het systeem aan moet voldoen voordat de applicatie kan worden gebruikt.

- SAP back-end systeem inrichten
- Sybase Workspace installeren
- Sybase Unwired Platform Server Instellen
- Verbinding maken tussen Sybase Unwired Platform en Workspace
- Verbinding maken tussen workspace en SAP back-end systeem
- Applicatie Ontwikkel omgeving installeren
- Verbinding maken tussen mobiele applicatie en het Sybase Unwired Platform
- Gegevens synchroniseren met Back-end

Al deze functionele en technische stories heb ik beschreven in het product backlog, te vinden in de bijlage. Per story heb ik een aantal storypoints toegekend, dit zijn punten die in verhouding staan met elkaar. Door middel van deze punten krijg je een goed beeld van hoe groot de story is en hoeveel tijd ik denk er mee bezig te zijn. Het aantal punten dat ik per story heb toegekend heb ik in eerste instantie zelf ingeschat. Daarna heb ik het aantal punten gevraagd aan de mede studenten bij mij op de afdeling. Ik heb elke keer het gemiddelde gepakt, maar mijn aantal punten zwaarder mee laten weten. Dit omdat het nogal lastig is voor de andere studenten om de punten toe te wijzen over een onderwerp waar ze niet veel van afweten.

## 9. Het ontwerp

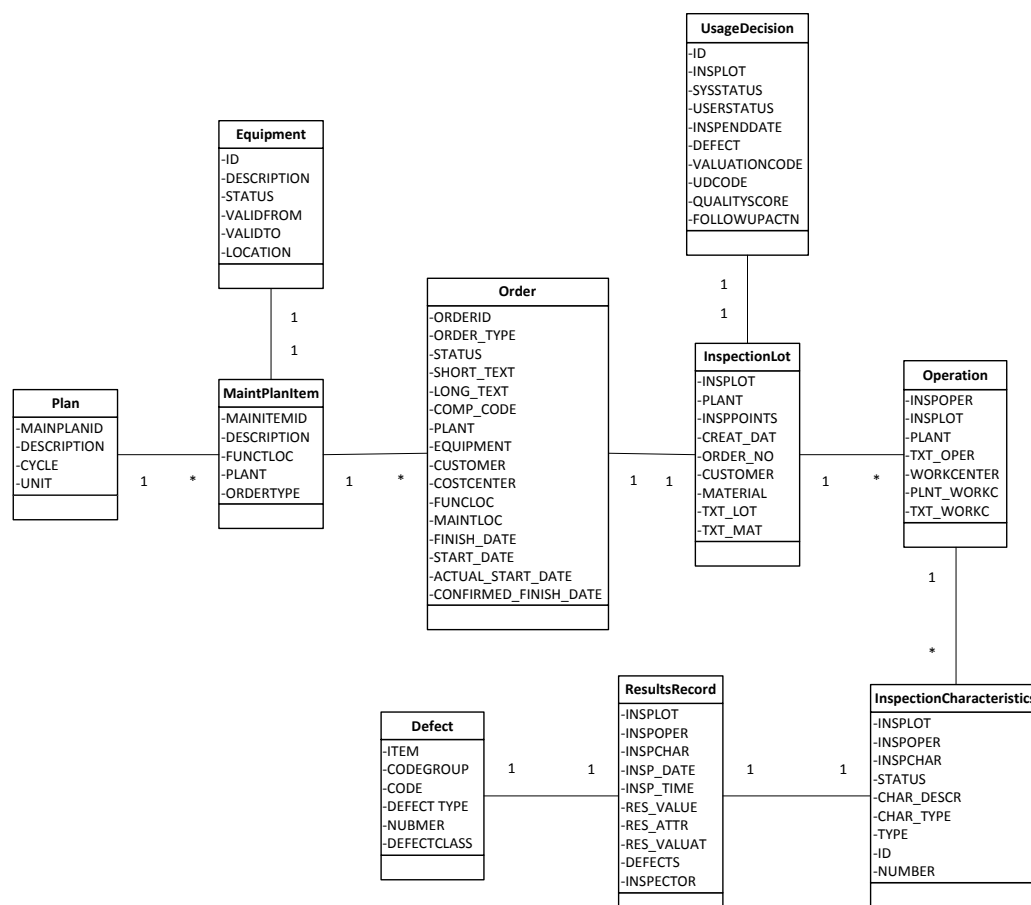
Dit project pak ik aan volgens de SCRUM methode. Volgens deze methode hoef ik geen specifieke ontwerpen te maken omdat je hierbij al teveel van de applicatie vastlegt. Bijvoorbeeld het technisch en functioneel ontwerp zijn meer documenten die worden gebruikt bij waterval methodes. Toch wil ik bij het ontwikkelen van deze applicatie een datamodel en een functioneel ontwerp maken omdat deze mij goed kunnen helpen bij het begrijpen van het proces het maken van aanpassingen en het analyseren van problemen. Daarnaast zal het voor de opdrachtgever en een buitenstaander eenvoudiger worden om de applicatie en de structuur hiervan beter te begrijpen.

### 9.1.1 Datamodel

Om het Inspectie proces duidelijk te maken heb ik drie datamodellen gemaakt. Één model voor de masterdata, een model voor de inspecties en een datamodel voor het uitvoeren van inspecties. Hierin is dat achterliggende database structuur van het Inspectie Management te zien. Ook kun je hieruit afleiden welke objecten uit SAP ik gebruik in de mobiele applicatie.

In het datamodel van SAP staan meer attributen dan ik hierin heb weergegeven. De attributen die zijn weergegeven zijn de attributen die ik heb gebruikt voor de applicatie. Deze selectie heb ik gemaakt op basis van informatie die ik van experts heb gekregen. Het is belangrijk om alleen de informatie in je applicatie op te slaan die ook echt gebruikt worden. Alle overige informatie is onnodig dataverkeer tussen applicatie en server.

Onderstaand datamodel is voor het uitvoeren van inspecties. Hierin is te zien welke objecten voor dit proces van belang zijn. De objecten die ik in dit datamodel gebruik zijn voortgekomen uit gesprekken met de verschillende experts tijdens het achterhalen van de requirements en door het SAP proces in het systeem te doorlopen.



Figuur 6: Inspectie management data model

## 9.1.2 Functioneel ontwerp

Als tweede ontwerp heb ik een functioneel ontwerp gemaakt. Hierin laat ik abstract zien hoe de applicatie er uit komt te zien, welke gegevens ik wil weergeven en welke informatie er moet worden ingevoerd. Ik heb dit functioneel ontwerp gemaakt in microsoft office Visio met behulp van wireframe. Dit is een speciaal template van Visio voor het visueel weergeven van applicaties. Ik heb het functioneel ontwerp gemaakt op basis van de requirements die besproken zijn met de experts. Ik zal in dit hoofdstuk twee belangrijke schermen toelichten.

### Inspectie overzicht

In het inspectie overzicht scherm is een overzicht te zien van alle inspecties met een beschrijving en een uit te voeren datum. Hierin kan de inspecteur alle aan hem toegewezen inspecties bekijken en door te selecteren ook het uitvoeren.

Wat een belangrijk verschil is tussen de SAP applicatie op een computer en een mobiele SAP applicatie is dat je geen ruimte hebt om heel veel informatie weer te geven. Wanneer er in SAP een lijstje opgevraagd wordt krijg je eerst 5 verschillende schermen te zien waarin je de filter aan kunt passen. Wanneer je de juiste filter hebt geselecteerd krijg je vervolgens een groot overzicht met alle informatie over het item. Bij een mobiele applicatie heb je die ruimte simpelweg niet omdat het scherm veel kleiner is.

Om dit op te lossen moet er een selectie worden gedaan op de gegevens die je weer wil geven in de mobiele applicatie. Ook moet de filter functie flink worden beperkt. Gezien de gegevens van inspecties heb ik voorgesteld om te kunnen filteren/zoeken op beschrijving, datum, klant of type inspectie. En vervolgens van elke inspectie te zien krijgt welk inspectie het is, de omschrijving het type en de start datum. Eventueel kun je hier ook nog de klant in vermelden maar dan wordt het onoverzichtelijker.

### Resultaten invoeren

In het volgende scherm krijg je een overzicht te zien welke resultaten er gemeten moeten worden. Daarnaast krijg je een aantal details van de taak te zien.

In dit scherm kom ik weer dezelfde problemen tegen als in alle schermen. Er is in SAP veel te veel informatie te zien voor een mobiele applicatie. Het is dus heel belangrijk om goed te overleggen met de experts welke details er wel zichtbaar moeten zijn en welke gegevens minder belangrijk zijn.

Wanneer tijdens het ontwikkelen blijkt dat dit niet genoeg gegevens zijn voor het uitvoeren van inspecties is eventueel een oplossing om door middel van een longclick functie extra details te laten zien wanneer iemand een lijst item lang ingedrukt houdt.

#### Inspection lots

Enter Text	Search
------------	--------

Inspection lots	
Inspection Lot: 14000000000084 Description: Plant 1 for Capgemini France Type: 14 Start date: 30.08.2012	>
Inspection Lot: 14000000000085 Description: Plant 2 for Capgemini France Inspection Type: 14 Start date: 10.09.2012	>
Inspection Lot: 14000000000086 Description: Plant 1 for Capgemini Netherlands Inspection Type: 14 Start date: 12.09.2012	>
Inspection Lot: 14000000000087 Description: Plant 2 for Capgemini Netherlands Inspection Type: 14 Start date: 12.09.2012	>
Inspection Lot: 14000000000084 Description: Plant 1 for Capgemini Netherlands Inspection Type: 14 Start date: 12.09.2012	>

#### Result Recording

Operations for inspectionlot: 14000000089	
Operation: 0010 Description: Messure oil temp Specifications: 50 ... 70 C Original Value: 60	>
Result: <input type="text" value="value"/>	
Operation: 0020 Description: Messure pipe width Specifications: 50 ... 60 mm Original Value: 55	>
Result: <input type="text" value="value"/>	
<input type="button" value="Submit"/>	

## 10. Architectuur

In dit hoofdstuk zal beschreven worden welke architectuur keuzes ik heb gemaakt op het gebied van de Android applicatie. Een deel architectuur van de applicatie is al vastgelegd omdat ik gebruik maak van Sybase. Hoe deze architectuur er precies uitziet heb ik beschreven in het onderzoek naar Sybase Unwired Platform.

### Native of workflow

Een van de belangrijkste keuzes voor het ontwikkelen van een applicatie met Sybase Unwired Platform is de keuze of het een Native of Workflow applicatie wordt. Deze keuze hangt grotendeels af van de functionaliteiten die de applicatie gaat krijgen.

Ik heb gekozen om een **native** applicatie te ontwikkelen. Een wens van de opdrachtgever is dat de applicatie offline functionaliteiten moet hebben. Dit wil hij omdat de inspecteurs niet afhankelijk moeten zijn van een internetverbinding willen ze de inspecties uitvoeren. Daarnaast heeft de applicatie ook een zekere intelligentie nodig voor het vergelijken van de inspectie resultaten met de toegestane afwijkingen. Bovendien is het voor in de toekomst een goede toevoeging om een koppeling te maken met een agenda, voor het plannen van inspecties. Gezien deze functionaliteiten is de keuze gevallen voor een native applicatie.

Het is voor Capgemini ook veel interessanter om te weten hoe er een native applicatie ontwikkeld moet worden. Een Workflow applicatie is al eens ontwikkeld door Capgemini en hier hebben ze dus al ervaring mee. Hierdoor is het juist interessanter om native applicatie te ontwikkelen en hiermee ervaring op te doen. Juist bij native applicaties is het interessant om te weten wat de extra mogelijkheden met zich meebrengen en wat de voordelen zijn. Bovendien is komen er veel extra handelingen bij kijken om de applicatie te laten verbinden met het Sybase platform.

### Platform

Nadat ik gekozen heb om een Native applicatie te gaan ontwikkelen moet er ook gekozen worden voor een platform waarop de applicatie zal gaan draaien. In de aanleiding van dit verslag heb ik geschreven over het aantal gebruikers van de verschillende mobiele platformen. Hieruit blijkt dat Android en IOS de meest gebruikte platformen zijn. Het is dan ook de bedoeling dat de applicatie gaat draaien op beide platformen.

Gezien de tijd waarin het project afgerond moet zijn is het niet mogelijk om twee applicaties te ontwikkelen, ik zal daarom maar voor één platform een demo applicatie ontwikkelen. Voor dit project moet ik dus een keuze maken tussen Android en IOS. Het is bekend dat er bij het ontwikkelen van IOS applicaties veel licenties en dergelijke nodig zijn. Ook het uitbrengen van de applicatie in de App store gaat gepaard met veel controles en beperkingen vanuit Apple. Het kan soms weken duren voordat een applicatie in de App store verschijnt. Bij Android zijn er veel minder controles, bovendien is Android open source en kan alle ontwikkelsoftware gratis van het internet gedownload worden. Daarnaast kun je met Android veel eenvoudiger de applicatie op een telefoon krijgen.

Omdat dit om een applicatie gaat waarbij het belangrijk dat veel mensen het snel kunnen zien en proberen is er gekozen voor **Android**. Dit is ideaal voor een demo applicatie omdat je op deze manier de applicatie eenvoudig aan veel verschillende klanten demonstreren.

## 11. Wijziging van de doelstelling

Uit het onderzoek wat ik gedaan heb en is gebleken dat het ontwikkelen van een native applicatie met Sybase niet zo eenvoudig is als aan het begin van dit project gedacht werd. Na het achterhalen van de requirements bij de verschillende experts ben ik erachter gekomen dat de onderliggende gedachte van de opdracht eigenlijk ergens anders over gaat. Het gaat over het onderzoeken van de mogelijkheden om een mobiele applicatie te ontwikkelen op het Sybase Unwired Platform die bedrijfsprocessen in SAP kan ondersteunen. Ik heb dan ook besloten om de doelstelling van dit project te wijzigen.

### 11.1 Probleem

Het probleem van de opdracht is dat het Sybase Unwired Platform een nieuwe oplossing is en er binnen Capgemini nog weinig ervaring is opgedaan met het ontwikkelen van een applicatie op dit platform. Er is veel informatie te vinden over SUP, het klinkt allemaal heel positief en de mogelijkheden zijn eindeloos, maar omdat er nog niemand is geweest die een applicatie heeft ontwikkeld die echt een koppeling heeft met een SAP back-end lijkt iedereen een beetje huiverig om dit platform te gebruiken. Er is dan ook behoefte aan ervaring met het ontwikkelen van dergelijke applicaties en een advies over hoe de ontwikkeling in de praktijk gaat.

### 11.2 Doelstelling

De nieuwe doelstelling is het onderzoeken hoe er een applicatie met Sybase Unwired Platform gebouwd kan worden en wanneer SUP het geschikte platform is om dit te doen. Hierover zal dan uiteindelijk een advies over gegeven worden aan de opdrachtgever.

In dit advies zal ik Capgemini adviseren over het gebruik van het Sybase Unwired Platform. Bij wat voor soort oplossingen zij dit het beste kunnen gebruiken, wanneer klanten geschikt zijn om een Sybase oplossing te implementeren en waar op gelet moet worden bij het ontwikkelen van een applicatie op het Sybase Unwired Platform.

Daarnaast zal er in de praktijk een applicatie ontwikkeld worden als voorbeeld. Deze laat zien hoe de koppeling eruit ziet en hoe de applicatie en synchronisatie werkt. De bevindingen van de ontwikkeling kunnen dan ook gebruikt worden als hulpmiddel of advies tijdens volgende ontwikkelingen van applicaties op dit platform.

### 11.3 Resultaat

Het resultaat van de gewijzigde doelstelling zal een advies zijn waarin beschreven wordt wat het Sybase Unwired Platform is, voor welke klanten deze oplossing geschikt is en bij welk soort applicaties het SUP platform geschikt is.

Ook beschrijf ik mijn bevindingen, tegen welke problemen ik ben aangelopen tijdens de ontwikkeling van een applicatie en hoe deze opgelost zijn. Hiermee kunnen ontwikkelaars in de toekomst sneller problemen analyseren en de oplossingen vinden. Ook kunnen ze hiermee rekening houden tijdens de ontwikkeling van nieuwe applicaties.

## 12. Ontwikkeling van de applicatie

In hoofdstuk 12 beschrijf ik de ontwikkeling van de mobiele applicatie. Ik heb in dit project twee sprints kunnen doen. Ik had er drie willen doen maar hier was niet genoeg tijd meer voor omdat het proces ervoor meer tijd heeft gekost dan gepland. In de eerste sprint heb ik het systeem klaar gezet en de BAPI's bepaald. In de tweede sprint heb ik een deel van applicatie ontwikkeld.

### 12.1 Gereed maken van het systeem

Dit is de eerste sprint van het project. Ik heb hierin vooral de techstories behandeld. Hierin zal ik het systeem klaar zetten en alle onderdelen installeren zodat ik na deze sprint klaar ben om er een Android applicatie mee te ontwikkelen.

#### 12.1.1 Planning

Voordat ik kon beginnen met sprinten is het belangrijk om eerst een sprintplanning te maken. Hierin wordt beschreven welke stories allemaal gemaakt moeten worden in de sprint.

Omdat het installeren en configureren van alle onderdelen in het hele systeem veel werk is heb ik besloten om dit te doen in een aparte sprint. In deze sprint zullen dus vooral de techstories uit het backlog worden gemaakt. Deze stories moeten gemaakt worden voordat ik kan beginnen met het ontwikkelen van de applicatie.

De stories die ik volgens de planning gemaakt heb tijdens de eerste sprint zijn:

- SAP back-end systeem inrichten
- Sybase Workspace installeren
- Sybase Unwired Platform Server Instellen
- Verbinding maken tussen Sybase Unwired Platform en Workspace
- Verbinding maken tussen Workspace en SAP back-end systeem
- Applicatie ontwikkel omgeving installeren

Behalve het installeren en configureren van het systeem is het ook belangrijk te weten welke BAPI's ik nodig heb om de functionele stories van de applicatie te kunnen maken. Ik zal in deze sprint dan ook onderzoeken welke BAPI's ik ga gebruiken voor de tweede sprint.

Verder zal ik in deze sprint rekening houden met een grote leercurve voor het werken met Sybase. Daarnaast zal er tijd worden gestoken in het leren programmeren met Java in Android door het volgen van tutorials op internet.

Aan het einde van de sprint worden de gemaakte stories getest. In de eerste sprint heb ik dit gedaan door de MBO's te testen. Je kunt dit in de Workspace testen door een preview op te vragen van de gegevens. Hiermee test je of de verbinding tussen de workspace en het back-end goed gaat en ook of het goed geïnstalleerd is.

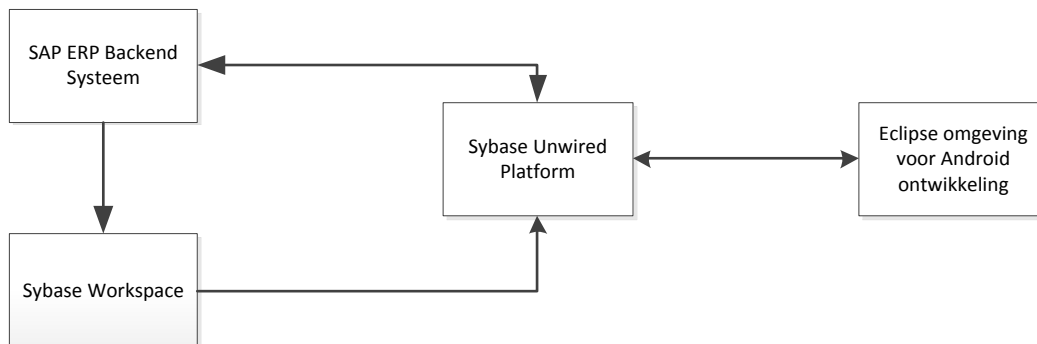
Naast de verbinding tussen de workspace en het back-end systeem moet ook de verbinding tussen de workspace en de Unwired Server getest worden. Dit heb ik gedaan door te testen of de server bereikbaar is voor de Workspace door deze te pingen.

De verbinding tussen de ontwikkelomgeving en de Unwired Server kan pas getest worden tijdens het ontwikkelen van de applicatie. Dit heb ik dan ook tijdens het testen van de tweede sprint gedaan.



### 12.1.2 Ontwikkeling

Tijdens de sprint heb ik alle systeem onderdelen geïnstalleerd en geconfigureerd. Hieronder een overzicht van alle verschillende programma's en servers om overzichtelijk te maken hoe de infrastructuur eruit ziet.



**Figuur 12.1: Applicatie omgeving**

#### SAP ERP back-end

De eerste story is het inrichten van het SAP back-end systeem. Omdat ik er al vrij snel in het project achter kwam dat dit veel tijd ging kosten en ik ook nog een back-end systeem uit moest kiezen heb ik deze story al eerder in het project gedaan. Voordat ik aan deze sprints begon was er dus al een ingericht back-end systeem. Dit systeem is grotendeels ingericht door experts. Hoe ik de keuze van het back-end systeem heb genomen en hoe het systeem is ingericht heb ik beschreven in hoofdstuk 7. In dit systeem is het mogelijk om het gehele inspectie proces te doorlopen.

#### Sybase Workspace

De volgende stap is het installeren van de Workspace. Via de mobility lead binnen Capgemini heb ik toegang gekregen tot een download center van Sybase. Hierin kun je de Workspace downloaden en installeren. Bij de installatie kwam ik erachter dat je een licentie nodig hebt om het programma langer dan 30 dagen te gebruiken. Van verschillende collega's kreeg ik te horen dat Capgemini een overeenkomst heeft met SAP dat ze deze licenties zoveel als nodig kunnen gebruiken. Alleen niemand wist precies bij wie ik moest zijn voor deze licenties.

Na lang zoeken en vragen kwam ik uit bij iemand uit Amerika die ook werkt voor Capgemini. Door met hem in contact te komen kreeg ik een developer licentie waarmee het mogelijk is om de Workspace te installeren en te koppelen met een Sybase Server voor het ontwikkelen van applicaties.

#### Sybase Unwired Platform

De volgende story is het installeren van een Sybase Server. Dit heeft heel wat problemen met zich meegebracht en heeft achteraf veel meer tijd gekost dan ik gepland had. Om te kunnen starten heb ik eerst een eigen Sybase server geïnstalleerd. Deze draait lokaal op je pc en is een trial versie waar je 30 dagen gratis gebruik van mag maken. Dit is een goede optie om in het begin te leren werken met het platform maar gaat voor de hele demo applicatie niet werken omdat de opdracht langer duurt dan 30 dagen. Tijdens de 30 dagen trial ben ik dan ook op zoek gegaan naar een andere oplossing.

Een goed alternatief was om een gratis trial omgeving aan te vragen van SAP. Dit betekent dat je een cloud server krijgt van SAP en hier staat het hele systeem voor je ingericht inclusief een SAP back-end. Deze is in principe ook 30 dagen beschikbaar, maar je krijgt de mogelijkheid om de trial licentie drie keer te verlengen. In totaal zou ik dus 90 dagen gebruik kunnen maken van dit systeem. Gezien de tijd van het project zou dit een goed alternatief zijn geweest. Maar helaas kon er na de trial periode geen gebruik meer gemaakt kunnen worden van de applicatie en dus kan Capgemini na 90 dagen de applicatie niet meer demonstreren aan klanten, dit is wel het doel van het project. Daarnaast is de SAP omgeving niet ingericht om het inspectie proces te kunnen doorlopen, ik zal dus het back-end systeem opnieuw moeten inrichten. Dit zou teveel tijd gaan kosten en bovendien heb ik de rechten daar niet voor gekregen van SAP. Helaas was dit dus ook geen juiste optie.

Nadat deze optie ook was afgefallen ben ik opzoek gegaan naar weer een ander alternatief. De SAP afdeling binnen Capgemini is bezig met het installeren van een aantal nieuwe systemen waaronder ook een eigen Sybase server, helaas zal deze te laat in het project beschikbaar zijn en kan ik hier dus niet op wachten. Wel is mij gevraagd om te helpen met het installeren en configureren van deze server. Omdat ik bij dit proces betrokken ben geraakt heb ik veel kunnen leren over het opzetten van een Sybase Server. Zo moeten er tijdens het installeren een aantal keuzes worden gemaakt over hoe de server opgezet moet worden.

Sybase options	Unwired platform	Online Data Proxy
Deployment options	Personal system	Enterprise system
High level design options	Non-Clustered systems	Clustered systems

Bij de Sybase opties kun je kiezen uit een Unwired platform en een Online Data Proxy. Een Unwired platform is gewoon een fysieke server en bij een Online Data Proxy host je de server bij SAP. Het is dan alleen mogelijk om workflow applicaties te ontwikkelen. Bij de deployment optie kun je kiezen voor personal system, dit is voor wat kleinere bedrijven en voor enterprise system voor de wat grotere bedrijven. Het verschil zit in het aantal mobiele apparaten wat je kunt aansluiten en de hoeveelheid Workspaces die je erop kunt instellen. non clustered of clustered system bepaalt of je een aantal servers in een cluster wilt hebben of maar één server gebruikt.

Doordat ik al begonnen was met het ontwikkelen van Sybase en geholpen heb met het opzetten van de nieuwe server ben ik in contact gekomen met een ontwikkelaar uit India. Ik kon hem aanspreken voor eventuele vragen over Sybase. Via die collega uit India kwam ik erachter dat zij werken met een Sybase Unwired Platform server. Ik heb dan ook aan hem gevraagd of ik toegang kon krijgen tot die server. Na overleg met de manager in India heb ik die toegang gekregen. Deze server wordt door collega's uit India gebruikt als demo omgeving. Het heeft uiteindelijk veel langer geduurd dan verwacht maar er was een server beschikbaar gekomen waar ik de applicatie mee kon ontwikkelen.

## Verbinding maken tussen Sybase Unwired Platform en Workspace

Nu de Server, de Workspace en de back-end geïnstalleerd en geconfigureerd zijn kan ik deze onderdelen gaan verbinden met elkaar. De eerste verbinding is tussen de Workspace en het platform. Omdat dit allebei onderdelen zijn van hetzelfde systeem is dit vrij eenvoudig te realiseren.

Wanneer je deze verbinding wilt opzetten moet je een aantal connectie gegevens invullen en je moet inloggegevens hebben van de Unwired server. Deze gegevens heb ik gekregen vanuit India. Omdat de server goed beveiligd is, is het alleen mogelijk om een verbinding met de server te maken op het moment dat je binnen het Capgemini netwerk bent. Daarbuiten wordt de verbinding geblokkeerd door de firewall.

Wanneer de verbinding gelegd is kun je het project wat je aan het ontwikkelen bent uitrollen naar de server zodat de applicatie gebruikt kan worden.

**Unwired Server Connection Properties**

Host: in-blr-solman  
IP domain: corp.capgemini.com  
Port: 2000  
User name: supAdmin  
Password: \*\*\*\*\*  
☒ Save password  
☐ Secure

Cluster information:

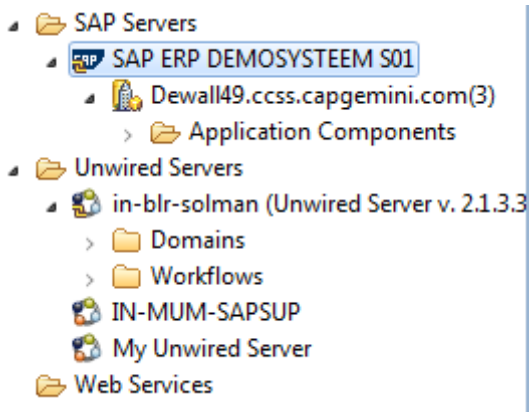
Property	Value
Server host	IN-BLR-SOLMAN
Protocol	http,https
Replication-based Synchronizatio	
Port	2480
Messaging-based Synchronizatio	
Port	5001,80

Figuur 12.2: Unwired Server connectie

## Verbinding maken tussen Workspace en SAP back-end

De volgende stap is een verbinding leggen tussen de Workspace en de SAP Back-end. Dit gaat op dezelfde manier als het maken van een connectie met de Unwired server. Deze gegevens kun je ook van de SAP server halen.

Wanneer er een verbinding is gelegd met de SAP back-end is het mogelijk om de verschillende BAPI's op te halen en te gebruiken. Nu de Workspace verbinding heeft gemaakt met een back-end systeem en de Unwired Server is deze omgeving klaar. In de enterprise explorer van de Workspace is te zien welke connecties er gelegd zijn. In dit geval gebruik ik de connectie naar SAP ERP DEMOSYSTEEM S01 en de Unwired Server in-blr-solman.



Figuur 72.3: Project connecties

Figuur 12.4: Back-end verbinding

## Applicatie ontwikkel omgeving installeren

De laatste omgeving die nu nog geïnstalleerd moet worden is de Eclipse omgeving met een Android plug-in om Android applicaties te ontwikkelen. Het installeren van deze omgeving heeft niet teveel problemen opgeleverd omdat dit heel goed beschreven staat in de online documentatie. Eerst moet er Eclipse geïnstalleerd worden en vervolgens kun je de Android plug-in installeren. Naast het installeren van de Android plug-in heb ik ook een simulator geïnstalleerd. Met behulp van deze simulator is het mogelijk om de applicatie te draaien en testen op een computer.

Het was mogelijk om de Android plug-in te installeren op dezelfde Eclipse omgeving als waar de Sybase Workspace op geïnstalleerd is. Dit heb ik geprobeerd maar ik kreeg toch allemaal vreemde fouten. Dit kwam vermoedelijk omdat de omgevingen niet compleet gescheiden waren. Ik heb er daarom voor gekozen om de Applicatie en de Workspace opnieuw gescheiden van elkaar te installeren. Hierdoor waren de problemen verdwenen, en het verder geen problemen tijdens de ontwikkeling.

## Te gebruiken BAPI's

Nadat de omgeving geïnstalleerd was en ook tijdens het installeren heb ik gekeken welke BAPI's ik kon gebruiken voor het bouwen van de applicatie. Er zijn in het SAP systeem standaard een aantal BAPI's gebouwd. Hieruit kun je de benodigde BAPI's kiezen en deze gebruiken om de applicatie mee te bouwen. Het komt ook voor dat de benodigde BAPI niet beschikbaar is, dan is er de mogelijkheid om zelf BAPI's te maken. De meeste BAPI's die nodig zijn voor dit project zijn beschikbaar in SAP.

Het bouwen van BAPI's is iets waar veel kennis van SAP en ABAP nodig is. Er is onderzocht of het mogelijk is om zelf een BAPI te ontwikkelen maar dit vergt zoveel tijd en kennis dat ik heb besloten om dit buiten de scope van het project te houden. De functionaliteiten die niet te realiseren zijn met de bestaande BAPI's zullen dus niet in de demo applicatie geïmplementeerd worden. Een voorbeeld van een dergelijke functionaliteit is het opvragen van de handleiding voor een inspectie uit het Document Management Systeem(DMS).

De vervolg stap is het zoeken naar de benodigde BAPI's. Met behulp van een SAP consultant heb ik het hele proces doorlopen in SAP. Hierbij zijn de Business objecten bepaald die nodig zijn bij het proces. Bij deze business objecten zijn de juiste BAPI's gezocht. De BAPI's die ik kan gebruiken tijdens het ontwikkelen zijn:

BAPI	Description
BAPI_INSPLOT_GETLIST	Select inspection lots
BAPI_INSPLOT_GETDETAIL	Load detail data for inspection lot
BAPI_INSPLOT_GETOPERATIONS	Select inspection operations for inspection lots
BAPI_INSPOPER_GETDETAIL	Read detail data for inspection operation
BAPI_INSPOPER_GETCHAR	Select inspection lot characteristics
BAPI_INSPCHAR_SETRESULT	Write inspection results

Met deze BAPI's is het mogelijk om Mobile Business Objecten(MBO) te maken. Omdat er volgens Scrum ontwikkeld wordt kunnen de functionaliteiten van de applicatie nog veranderen. Deze lijst van BAPI's kan tijdens de ontwikkeling dus nog wijzigen. Met het Unwired Platform is het mogelijk om achteraf nog MBO's toe te voegen of te verwijderen en daarna toe te voegen aan de applicatie.

Wanneer er een MBO aangemaakt wordt moeten er een aantal input en output parameters worden gedefinieerd. Deze velden komen overeen met de database velden uit SAP. Wanneer je bijvoorbeeld een order wil hebben die bij een inspectie lot hoort moet je als input parameter het inspectionlotID meegeven en als output alle benodigde gegevens van de order. Een lastig punt hierbij was de initiële lijst met inspecties. Je hebt hierbij dan niet een specifieke input parameter en dit moet je dan ook aangeven met een speciaal veld waarin je een maximaal aantal records aangeeft en in welke volgorde.

Nadat alle MBO's aangemaakt zijn in Sybase worden de relaties tussen deze MBO's gelegd. Deze relaties komen overeen met het datamodel dat ik eerder gemaakt heb. Wanneer alle MBO's gegenereerd zijn en de relaties gemaakt is de basis van de applicatie klaar. Vervolgens moet de applicatie op de server uitgerold worden en kan de applicatie bij deze gegevens via de server.

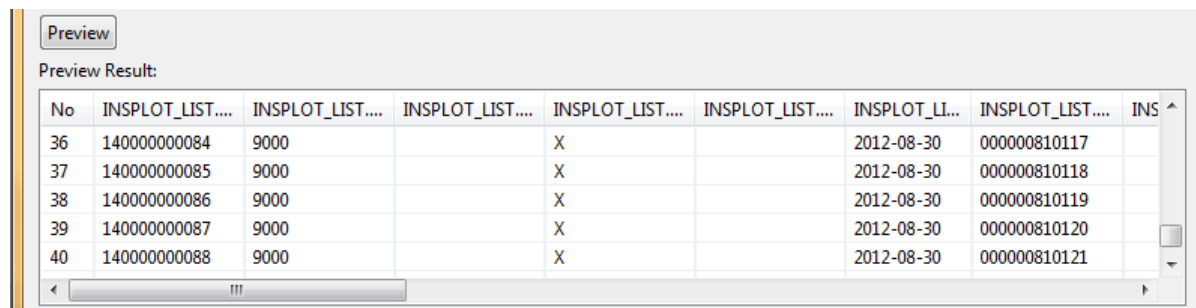


Figuur 12.5: Van MBO naar applicatie

### 12.1.3 Testen

In de testfase van deze sprint heb ik vooral de MBO's getest op de juiste werking. Hiermee controleer je of de BAPI's juist zijn ingesteld. Je kunt de MBO's testen door een preview op te vragen van de gegevens die je opvraagt. Wanneer je dan de gegevens terugkrijgt die je verwacht is de MBO goed ingesteld.

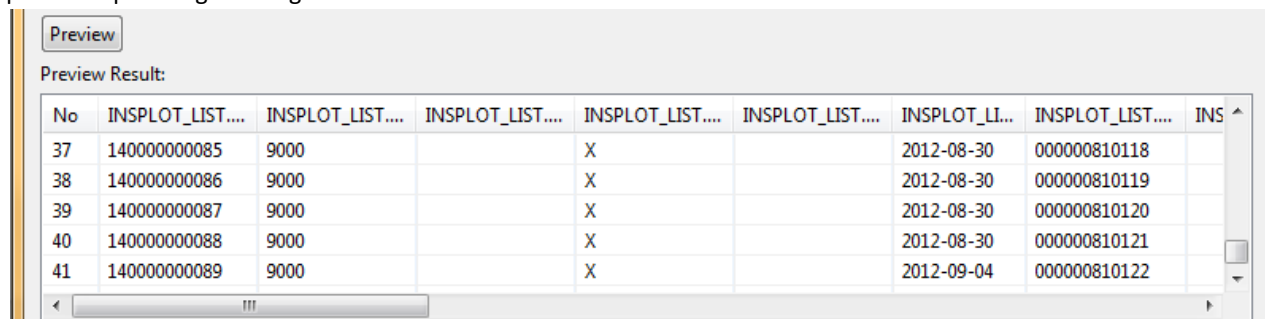
De eerste MBO die ik ga testen zal de lijst met inspecties zijn. Hier hoeft je dus geen specifieke input parameters aan mee te geven. Wanneer je op preview klikt, verwacht je een lijst met inspection lots te zien. Zoals hieronder is te zien gaat dit goed.



No	INSPLOT_LIST....	INSPLOT_LIST....	INSPLOT_LIST....	INSPLOT_LIST....	INSPLOT_LIST....	INSPLOT_LI...	INSPLOT_LIST....	INS
36	140000000084	9000		X		2012-08-30	000000810117	
37	140000000085	9000		X		2012-08-30	000000810118	
38	140000000086	9000		X		2012-08-30	000000810119	
39	140000000087	9000		X		2012-08-30	000000810120	
40	140000000088	9000		X		2012-08-30	000000810121	

Figuur 12.6: Test Resultaat 1

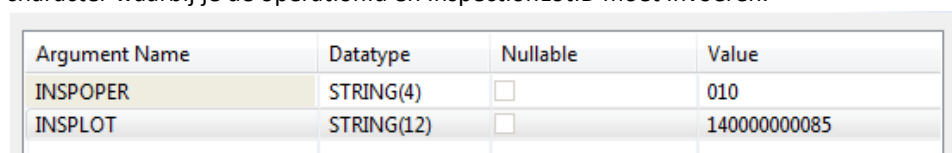
Wat ik ook heb getest is of het systeem de nieuwste gegevens rechtstreeks uit de back-end haalt of ophaalt uit het eigen geheugen. Om dit te testen heb ik een nieuw record gemaakt en gekeken of deze ook opgehaald wordt. Na opnieuw de preview op te vragen kreeg ik inderdaad het nieuwe record te zien.



No	INSPLOT_LIST....	INSPLOT_LIST....	INSPLOT_LIST....	INSPLOT_LIST....	INSPLOT_LIST....	INSPLOT_LI...	INSPLOT_LIST....	INS
37	140000000085	9000		X		2012-08-30	000000810118	
38	140000000086	9000		X		2012-08-30	000000810119	
39	140000000087	9000		X		2012-08-30	000000810120	
40	140000000088	9000		X		2012-08-30	000000810121	
41	140000000089	9000		X		2012-09-04	000000810122	

Figuur 8.7: Test Resultaat 2

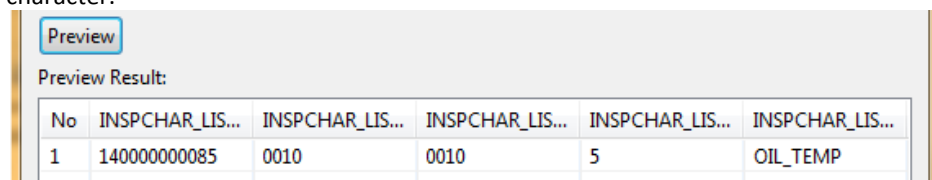
Vervolgens heb ik ook een MBO getest waarbij je input parameters moet opgeven. Als voorbeeld neem ik een inspection character waarbij je de operationId en InspectionLotID moet invoeren.



Argument Name	Datatype	Nullable	Value
INSPOPER	STRING(4)	<input type="checkbox"/>	010
INSPLOT	STRING(12)	<input type="checkbox"/>	140000000085

Figuur 12.8: Test input parameters

Als resultaat krijg je hierbij een lijst van de characters behorende bij de operatie. Dit is zoals ik verwachtte maar één character.



No	INSPCHAR_LIS...	INSPCHAR_LIS...	INSPCHAR_LIS...	INSPCHAR_LIS...	INSPCHAR_LIS...
1	140000000085	0010	0010	5	OIL_TEMP

Figuur 12.9 Test Resultaat 3

Op deze manier heb ik getest of de werking van de MBO's juist was. En hier kun je dus ook uit concluderen dat de verbinding met het back-end systeem goed werkt. Wat ik ook heb getest is of de verbinding met de Unwired Server goed gaat. Dit heb ik gedaan door middel van een ping. Het resultaat hiervan is of ping succeeded of ping failed. Tijdens het testen ging de ping goed en had ik dus verbinding. Wanneer de ping niet goed was gegaan lag het waarschijnlijk aan de connectie instellingen van de Workspace.

## 12.2 Ontwikkelen applicatie

In de tweede sprint ga ik een gedeelte van de applicatie ontwikkelen. Hierin zullen de functionaliteiten geprogrammeerd worden, zodat er uiteindelijk een demo omgeving kan komen. Deze sprint is later begonnen dan ik gepland had en ook tijdens de sprint is niet alles gegaan zoals verwacht. Hierdoor is de applicatie helaas niet helemaal af gekomen. Wel is er een werkende applicatie uitgekomen die een goed beeld geeft van de mogelijkheden van Sybase Unwired Platform

### 12.2.1 Planning

Na het klaarzetten van het systeem en het onderzoeken welke BAPI's ik kan gebruiken ben ik begonnen met het programmeren van de applicatie. De applicatie is ontwikkeld in Eclipse met de Android plug-in. Eerst wordt er code gegenereerd van de MBO's die ik gedefinieerd heb in de Workspace. Vervolgens heb ik deze code in het project gezet en kon ik de package gebruiken voor het programmeren van de applicatie.

In deze sprint was gepland om de volgende functionele stories te bouwen:

- Ophalen van inspecties
- Ophalen van uit te voeren taken
- Inloggen
- Resultaten van verrichte metingen invoeren
- Resultaten van voorgaande metingen bekijken

En de volgende technical stories:

- Verbinding maken tussen mobiele applicatie en het Sybase Unwired Platform
- Gegevens synchroniseren met Back-end

Bij het plannen van deze sprint heb ik rekening gehouden met het feit dat ik nog geen ervaring heb met het ontwikkelen van een Android applicatie. Ik zal dus vooral in het begin nog veel moeite hebben met het programmeren maar daarnaast ook veel leren.

Aan het einde van deze sprint is het mogelijk om een lijst met inspecties te krijgen en bij elk van deze inspecties de resultaten in te kunnen voeren. Echter door de weinige ervaring die er is over Sybase en de koppeling met SAP is het moeilijk in te schatten of het een realistische planning is.

Uiteindelijk is het niet gelukt om alle stories in de applicatie te krijgen binnen deze sprint. Ik had eigenlijk nog een sprint nodig om alle functionaliteiten in te bouwen. Maar door de beperkte tijd kon ik deze sprint niet meer inplannen.

Om de kwaliteit van de applicatie aan te tonen moet deze getest worden. Het testen van deze mobiele applicatie wilde ik doen met behulp van:

- JUnit testen
- Activity testen
- Layout testen
- Gebruikers testen

Met JUnit testen test je de code van de applicatie en de methodes die geschreven zijn. Vooral de methodes die gebruikt zijn om de verbinding met de Unwired server tot stand te brengen zijn interessant om te testen.

Activity Testen houdt in dat je de verschillende statussen van een activity nabootst, bijvoorbeeld sluiten, starten en pauze. Deze statussen krijgt de applicatie wanneer die gebruikt wordt. Het is belangrijk om dit te testen om te voorkomen dat een applicatie crasht wanneer de gebruiker bijvoorbeeld tussendoor een andere applicatie opent.

Met Layout testen test je hoe de applicatie zich gedraagt met verschillende scherm resoluties. Het is belangrijk om te weten dat een applicatie niet crasht of dat de controls op rare plekken komen te zitten wanneer een gebruiker een tablet gebruikt met een groter scherm.

Met de gebruikerstest wil ik testen of de user interface van de applicatie duidelijk is en of er nog bugs of dergelijke in de applicatie zitten. Dit kun je het beste testen door gebruikers de applicatie te laten gebruiken.

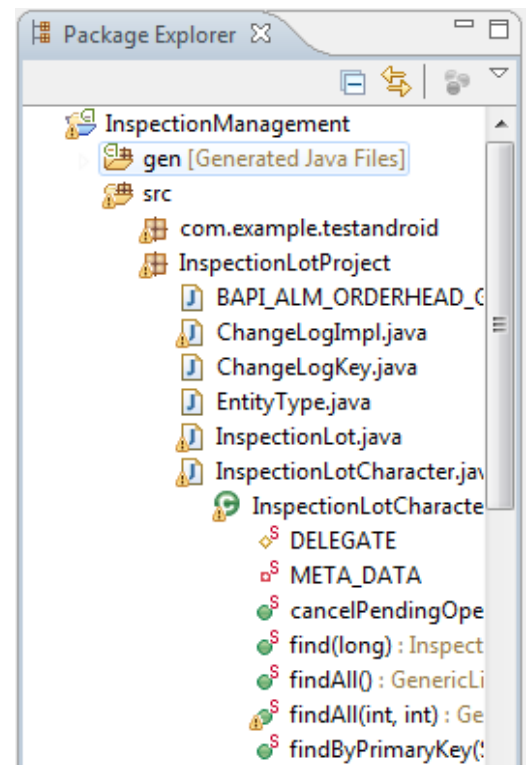


## 12.2.2 Ontwikkeling

Nadat er de benodigde MBO's gemaakt zijn is het mogelijk om code hiervoor te laten genereren. In deze code worden de MBO's omgezet naar objecten in bijvoorbeeld JAVA of Objective C. Ook worden de connectie gegevens in de code gegenereerd. Deze code kan dan vervolgens in een Android project worden geïmporteerd als een package. In deze classes zitten een aantal gegenereerde methode, bijvoorbeeld een getList methode of een FindByPrimaryKey methode.

Hier worden voor de verschillende objecten een classe met methodes gegeneerd, er wordt ook een class aangemaakt waarmee je verbindingen kan maken. Deze klasse heet InspectionLotProjectDB. Hierin kun je verschillende connectie properties aanmaken die zorgen voor de verbinding met de Unwired Server. In deze class staan bijvoorbeeld methodes als this.register() en this.connect(). Dit kan heel erg handig zijn omdat je deze methodes niet zelf hoeft te schrijven. Wat ik wel gemerkt heb tijdens het ontwikkelen is dat je niet kunt zien wat er precies gebeurt in de methode. Wat er vaak gebeurde is bijvoorbeeld dat je bij het starten van de applicatie niet door de register() methode heen kwam. Het is dan heel lastig te achterhalen waar het precies fout gaat in deze methode.

In de figuur hiernaast is goed de structuur te zien van het project. Je hebt de com.example.testandroid package. Deze package heb ik zelf aangemaakt en hierin staan de classes van de applicatie. De inspectionLotProject package is de gegenereerde package. Hierin staan alle gedefinieerde MBO's met daarin de verschillende methodes die je kunt gebruiken. Voordat je deze methodes echt kunt gebruiken moet er wel eerst verbinding worden gelegd met de Unwired Server.

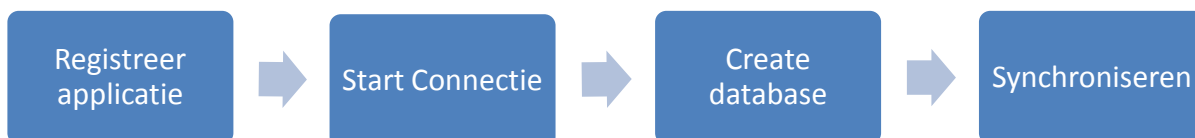


Figuur 9.10: package classes

### Verbinding maken tussen mobiele applicatie en het Sybase Unwired Platform

Voordat er gegevens opgehaald kunnen worden door de applicatie moet er eerst een verbinding komen tussen de applicatie en de Unwired Server. Deze verbinding moet in de code tot stand komen. Het verbinden van de applicatie gebeurt in een aantal stappen. De eerste keer dat de gebruiker de applicatie opstart zal de applicatie zich eerst moeten registreren vervolgens de connectie starten, database aanmaken en dan synchroniseren. Bij het registreren wordt er een user aangemaakt in het control center. Wanneer de applicatie voor de tweede keer op hetzelfde apparaat wordt gestart herkend de server de user en hoeft er niet meer geregistreerd te worden en hoeft er ook geen database meer aangemaakt te worden.

Hieronder laat ik met behulp van een proces flow zien welke stappen er genomen moeten worden voordat de applicatie verbinding heeft met de Unwired Server.



Figuur 12.11: Verbinden van de applicatie met SUP

Het registreren van de applicatie kun je doen met een methode uit de gegenereerde package. Wel moeten er een aantal parameters meegegeven worden. Hierin staan onder andere de inloggegevens op de server, de locatie van de server en via welke poort de verbinding tot stand moet komen.

Bij het verbinden van de applicatie zal de applicatie zich aanmelden met de geregistreerde user. De user komt dan ook online te staan in het control center en er wordt een log bijgehouden zodat eventuele fouten of meldingen geregistreerd worden. In onderstaande code is te zien hoe er geregistreerd en verbonden wordt.

```
ConnectionProperties connProps = app.getConnectionProperties();
LoginCredentials loginCredentials = new LoginCredentials(USERNAME, PASSWORD);
connProps.setLoginCredentials(loginCredentials);
connProps.setServerName(HOST);
connProps.setPortNumber(CONNPORT);

if (app.getRegistrationStatus() != RegistrationStatus.REGISTERED)
{
    app.registerApplication(TIMEOUT);
    Log.d(TAG, "timeout");
    // isReg = true;
}
else
{
    app.startConnection(TIMEOUT);
    // isReg = true;
    Log.d(TAG, "server connected");
}
```

Bij het verbinden van de applicatie liep ik vaak tegen problemen op. De problemen waren vaak ook onverklaarbaar en steeds verschillend. Het lijkt erop dat het platform dan ook nog niet zo stabiel genoeg is om native applicaties te ontwikkelen. Een voorbeeld van een dergelijk probleem is dat de applicatie bij het verbinden geen toestemming meer had op de server. Nadat ik dan vanaf het begin opnieuw het hele project uitgerold had op de server en opnieuw de code gegenereerd en geïmporteerd had in het project deed die het dan wel weer. Het lijkt er dus op dat er tijdens het uitrollen kleine foutjes optreden waardoor de hele applicatie niet werkt.

Nadat er verbinding is gelegd moet er een database gecreëerd worden op de smartphone. Hier slaat de applicatie zijn gegevens op zodat er niet continue gegevens over en weer worden gestuurd. De database bestaat alleen uit de informatie die de applicatie gebruikt. Deze database hoeft ook alleen de eerste keer dat de applicatie start aangemaakt te worden. Ook dit gebeurt automatisch met de gegenereerde code.

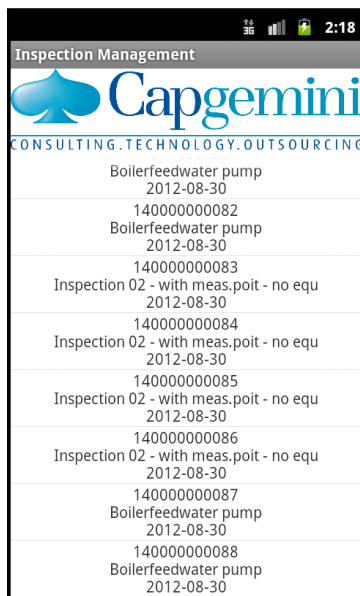
Als er een connectie en een database is kan de applicatie zijn gegevens synchroniseren met de server. Dit gebeurt via het HTTP protocol en moet ook in de code aangegeven worden. De gegevens worden dan vanaf de server overgezet naar de database op de smartphone. Dit gebeurt via Replication Based Synchronisation. Hoe dit precies gaat heb ik onderzocht in het onderzoek. Zodra de gegevens op de smartphone staan kan ik de gegevens in de applicatie gebruiken

### Ophalen van inspecties

Ik bouw de applicatie in de programmeertaal JAVA. De eerste stap is het ophalen van gegevens en deze tonen op het scherm. In dit geval een lijst met inspecties waar de gebruiker uit kan kiezen. Het bouwen van een Android applicaties met JAVA is iets anders dan gewoon een programma schrijven voor een pc.

Android werkt met activities, elke activity is een apart scherm. Een activity bestaat uit twee lagen. Een deel voor de presentatie(XML) en een deel voor de applicatie(JAVA). In het presentatiedeel geef je aan welke controls er in het scherm moeten zitten. Dit kunnen bijvoorbeeld een knop of een listview zijn. In het applicatie deel kun je alle methodes uitvoeren. Hierin geef je dus aan wat de knoppen moeten doen of wat er in de lijst moet komen te staan.

Om een lijst met inspecties op het scherm te krijgen moet je de inspecties ophalen en aan de lijst koppelen. Deze lijst laat je dan vervolgens op het scherm zien. Om een lijst met inspecties aan een listview in JAVA te koppelen heb ik gebruik gemaakt van het adapter pattern. Dit houdt in dat er een adapter class aan de listview gekoppeld wordt. In deze class wordt bepaald welke gegevens er in de lijst komen en hoe deze weergegeven worden. De adapter weet dus welke gegevens er zijn en de listview eet alleen maar dat er een adapter gekoppeld is.



Op deze manier heb ik een scherm kunnen bouwen waarin alle inspecties te zien zijn. In de adapter wordt bepaald welke gegevens er getoond worden. Dus in de toekomst is het mogelijk om hier een filter aan toe te voegen waardoor bijvoorbeeld alleen inspecties voor een bepaalde user getoond kunnen worden.

Hiernaast is te zien hoe het scherm eruit ziet in de applicatie. Hierin kun je dus overzichtelijk zien welke inspecties er zijn.

Wanneer je een inspectie selecteert door er op te klikken wordt er een nieuwe activity gestart. In dit scherm zie je de verdere details van de inspectie en kun je de resultaten invoeren. Aan dit scherm kun je het inspection lot doorgeven waardoor de applicatie gelijk weet welk lot het heeft. Dit object kun je doorgeven met behulp van een Intent. Met een intent is het mogelijk om allerlei informatie tussen activiteiten uit te wisselen. Dit kunnen bijvoorbeeld strings zijn maar ook objecten.

In het volgende scherm zijn alle details van het inspection lot zichtbaar. Er is te zien om welk lot het gaat en de beschrijving. In de specificatie staat tussen welke waarde het gemeten resultaat moet liggen en bij de originele waarde staat het verwachte resultaat. Het uiteindelijke resultaat kun je invullen onder het kopje resultaat. Wanneer je het resultaat hebt ingevuld druk je op save en ga je terug naar het inspectieoverzicht.

Dit scherm heb ik nog niet volledig functioneel gekregen. Zo kun je nu nog maar één inspectieresultaat invoeren. Wanneer er meer inspectie resultaten ingevoerd moeten worden per inspectie werkt dit nog niet.

Dit heb ik niet volledig afgekregen omdat ik niet genoeg tijd had om tijdens deze sprint genoeg van Android programmeren te leren om dit soort schermen te ontwikkelen. Uiteindelijk heb ik maar twee weken echt kunnen programmeren en binnen deze korte tijd is het niet mogelijk om een complete applicatie te bouwen zonder dat je veel verstand van Android applicaties bouwen hebt.

Wel denk ik dat de applicatie na wat kleine aanpassingen een goede indruk kan geven over wat er mogelijk is met het Sybase Unwired Platform

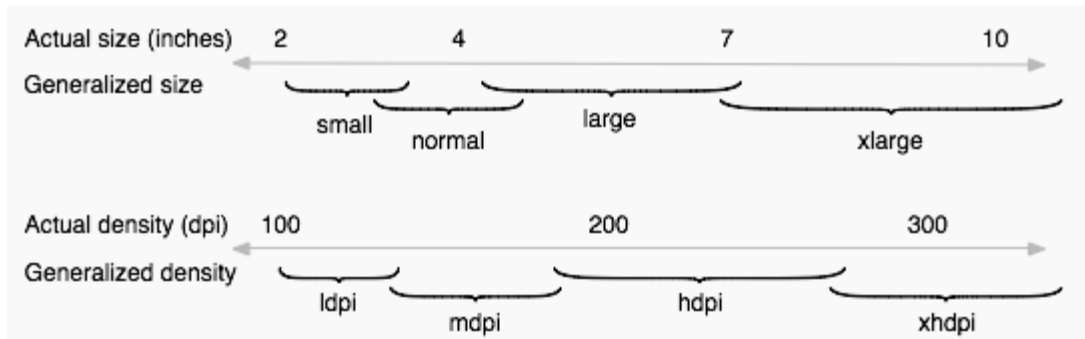
Als laatste heb ik nog een log in scherm ontwikkeld. Dit scherm werkt nog niet met de echte SAP user gegevens. Het idee van dit scherm is dat potentiële klanten de applicatie als demo kunnen gebruiken en hierdoor met gegevens uit een testsysteem van Capgemini kunnen testen. Hiermee kan de applicatie de klanten overtuigen om dit systeem voor hun bedrijf te gebruiken.

Uiteindelijk is het wel de bedoeling dat gebruiker via dit scherm kan inloggen op de applicatie en hiermee kan worden gecontroleerd wie de applicatie gebruikt en of de gebruiker toegang heeft tot deze gegevens.

### 12.2.3 Testen

Na deze korte sprint heb ik de geschreven code getest om de kwaliteit hiervan aan te tonen. Helaas heb ik niet genoeg tijd gehad om de applicatie af te maken en volledig te kunnen testen. Bijvoorbeeld de gebruikerstest is nog niet gedaan omdat de applicatie nog niet alle functionaliteiten heeft. Ik heb ervoor gekozen om alleen de layout test te doen. Ik heb hiervoor gekozen omdat ik voor de JUnit test nog niet genoeg geschikte code had. Bij de meeste code die ik tot nu toe heb maakt de applicatie verbinding met de server en het is niet wenselijk om bij elke test de Unwired server aan te spreken. Ik heb de Activity test niet gedaan omdat ik hier niet genoeg tijd voor had, het is zeker wel belangrijk om deze test te doen voordat de applicatie als demo uitgebracht wordt.

Met de layout test kun je testen of de applicatie er goed uit ziet en nog bruikbaar is op het moment dat de scherm grote verandert. Android ondersteund vanaf versie 1.6 vier verschillende scherm grote en pixel dichtheid te onderscheiden.



Elke schermgrote die hierboven aangegeven wordt heeft een minimum resolutie.

- Xlarge scherm: 960dp x 720dp
- Large scherm: 640dp x 480dp
- Normal scherm: 470dp x 320dp
- Small scherm: 426dp x 320dp

Bij het zoeken naar een goede methode om dit te testen kwam ik erachter dat Android dit voornamelijk zelf regelt. Alle controls zoals knoppen, lijsten en teksten worden door Android zelf geschaald naar de juiste resolutie.

(<http://www.tested.com/news/news/1381-how-android-scales-apps-for-different-screen-sizes/>)

Waar ik aan moet denken is dat de afbeeldingen die worden gebruikt niet een te lage resolutie hebben. Waar ook aan gedacht moet worden is dat bij plaatsen van controls niet een vaste positie wordt meegegeven. De positie die wordt meegegeven moet afhankelijk zijn van de parent van dit control.

Om de controls neer te zetten zonder een positie heb ik gebruik gemaakt van `fill_parent`, dit zorgt ervoor dat de hoogte en de breedte altijd overeen komen met de parent. In dit geval is de parent het frame layout. En hiervan de parent is het scherm zelf. Wanneer een scherm dus van hoogte of breedte verandert, veranderen deze controls mee.

Mat gravity geef je aan ten opzicht van wat het control zich moet bevinden. Bij `gravity = bottom` zal het control altijd dezelfde positie houden ten opzichte van de onderkant van het scherm.

```
android:layout_width="fill_parent"
android:layout_height="fill_parent"
android:orientation="vertical" >
```

```
<FrameLayout
    android:layout_width="fill_parent"
    android:layout_height="fill_parent" >

<ImageView
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    android:scaleType="fitStart"
    android:src="@drawable/logo_capgemini" />
```

```
<ListView
    android:id="@+id/listView1"
    android:layout_width="fill_parent"
    android:layout_height="402dp"
    android:layout_gravity="bottom" >
</ListView>
```

## 13. Advies

Tijdens het uitvoeren van het project is de doelstelling gewijzigd. Door het onderzoek en het achterhalen van de requirements ben ik erachter gekomen dat de vraag eigenlijk is hoe er een applicatie op het Sybase Unwired Platform ontwikkeld kan worden. Ik zal in dit hoofdstuk dan ook een advies geven aan de opdrachtgever over wanneer Sybase Unwired Platform een geschikt platform is voor een bedrijf om applicaties te ontwikkelen en welke applicaties er dan mee ontwikkeld kunnen worden.

### Klanten

De eerste vraag is voor welke bedrijven het Sybase Unwired Platform een geschikt platform is. Ofwel waarom zouden bedrijven MEAP willen gebruiken. MEAP kan een hele goede oplossing zijn voor grote bedrijven die hun bedrijfsprocessen mobiel willen gaan ondersteunen. Sommige bedrijven denk dat ze er al zijn wanneer ze iedere medewerker een mobiele telefoon geven. Maar vergeten dat het belangrijkste is om ervoor te zorgen dat de medewerkers deze telefoons ook echt gaan gebruiken om de processen efficiënter te maken. Dit gaan medewerkers alleen doen wanneer er de juiste applicaties voor ontwikkeld worden en wanneer de gebruikers de juiste instructies krijgen voor het gebruik. Dit laatste wordt door veel bedrijven ook vaak vergeten. Het is belangrijk dat bedrijven zicht goed blijven realiseren wat het doel is van de applicatie die ze in SUP willen gaan ontwikkelen. Wanneer een applicatie te snel of zonder genoeg functionaliteiten gebouwd wordt, of de applicatie ziet er niet aantrekkelijk genoeg uit weerhoudt dit gebruikers ervan om deze te gebruiken.

Wanneer een bedrijf een goed beeld heeft over welk proces ze mobiel willen gaan ondersteunen kunnen ze gebruik gaan maken van een MEAP als Sybase Unwired Platform. Ieder bedrijf dat gebruik maakt van één of meerdere back-end systemen is geschikt om SUP te gebruiken. De voorkeur van het Sybase Unwired Platform gaat uit naar een SAP back-end omdat hierbij de ondersteuning het best is. Wel moeten bedrijven rekening houden met de grote investering die ze gaan doen en de uiteindelijk baten die ze eruit kunnen halen. Omdat SUP nog in de kinderschoenen staat kunnen de ontwikkelingen hard gaan en is het lastig om te zeggen waar SUP over een jaar staat. Gezien de positie die SUP nu heeft zou ik adviseren dat een minimum aantal medewerkers dat een bedrijf moet hebben of verwachten te gaan krijgen rond de 150 liggen willen de voordelen groter zijn dan de nadelen. Bij een aantal gebruikers van minimaal 150 zullen de "Total Cost of Ownership"(TCO) kleiner zijn dan bij andere oplossingen zoals een Mobile Point Solution.

### Applicaties

De tweede vraag is wat voor soort applicaties er het beste ontwikkeld kunnen worden met het Sybase Unwired Platform. Een antwoord op deze vraag is dat alle soorten applicaties gebouwd kunnen worden, zolang de data maar in een back-end systeem staat. Natuurlijk moet er bij het ontwikkelen van een mobiele applicaties goed gekeken worden naar het nut van de applicatie. Wegen de kosten van het ontwikkelen wel op tegen het efficiënter maken van het proces. Daarnaast moet de applicatie zo ontwikkeld zijn dat een medewerker het ook daadwerkelijk gaat gebruiken, ze moeten het nut ervan inzien. Naar aanleiding van de verschillende SUP experts die ik gesproken heb ligt de grootste behoefte op dit moment vooral bij applicaties die administratieve processen kunnen ondersteunen. Denk hierbij bijvoorbeeld aan het invullen van uren.

Ik ben erachter gekomen tijdens dit project dat wanneer er een native applicatie ontwikkeld gaat worden er specifieke kennis nodig is van ABAP en de programmeertaal van het platform. Daarnaast zijn er in de huidige versie van SUP een aantal onderdelen, zoals het verbinden en synchroniseren van een native applicatie, naar mijn mening nog niet stabiel genoeg.

Wanneer een bedrijf SUP wil gebruiken zou ik adviseren om gebruik te maken van de eenvoudigere workflow applicaties. Hier heb je geen mensen voor nodig die specifieke programmeer kennis hebben, en het ontwikkelen hiervan gaat een stuk sneller. Bovendien hoeft er maar één applicatie ontwikkeld te worden voor alle mobiele apparaten in het bedrijf.

Met een workflow applicatie kunnen er misschien iets minder functionaliteiten ingebouwd worden, maar hierdoor blijft de applicatie wel heel basaal waardoor het puur doet waarvoor het gebouwd is. Hierdoor weten gebruikers ook precies wat ze kunnen verwachten van de applicatie en zal de drempel om het te gebruiken lager zijn.

## 14. Evaluatie

In dit hoofdstuk evalueer ik het doorlopen proces, de opgeleverde producten en hoe deze tot stand zijn gekomen.

### 14.1 Proces evaluatie

Voordat ik officieel begonnen ben met het afstuderen had ik er al een proces opzitten. Door omstandigheden ben ik zelf op zoek gegaan naar een nieuwe opdracht binnen Capgemini en heb ik zelf een bedrijfsbegeleider gevonden. Achteraf ben ik blij dat ik dit heb gedaan omdat je hiermee veel mensen in het vakgebied leert kennen en de verschillende ideeën hoort. Ook leer je het bedrijf beter kennen en leer je hoe je een goed idee uiteindelijk uit kunt werken binnen een groot bedrijf.

Nadat de opdracht was goedgekeurd door school ben ik begonnen met uitvoeren van de opdracht. Eerst heb ik me georiënteerd op de opdracht. Hierbij heb ik vooral gezocht naar de verschillende experts op het gebied van inspectie management en heb ik de opdracht besproken met mijn begeleider. Op deze manier kom je meer te weten over de opdracht en hoor je de verschillende ideeën. Met behulp van deze ideeën heb ik het plan van aanpak opgesteld.

Eén van de persoonlijke doelen die ik vooraf gesteld heb is het leren samenwerken in een internationale setting. Vooral het samenwerken met mensen uit India. Van Nederlands collega's had ik wel gehoord dat dit vaak heel verwarrend kan zijn door het cultuurverschil. Daarnaast was ik niet gewend om in het Engels te werken. Tijdens het project kom je er dan ook achter dat het cultuur verschil nog het grootste probleem is. Ik heb gemerkt dat het heel belangrijk is om te zorgen dat alles wat je zegt of vraagt maar op één manier geïnterpreteerd kan worden, anders krijg je het antwoord op heel andere vragen dan je bedoeld hebt. Ook is het niet gebruikelijk om in die cultuur nee te verkopen, ze zullen dus altijd ja zeggen terwijl ze soms niet zeker weten of het zo is.

Ik heb geleerd dat binnen een groot bedrijf als Capgemini alles mogelijk is, zolang je het maar zelf regelt. Er wordt tijdens het afstuderen veel zelfstandigheid en eigen verantwoordelijkheid van je verwacht. Er is niemand geweest die mij tijdens het project stap voor stap heeft verteld wat ik moet doen en met wie ik moet gaan praten. Ik heb alles zelf uitgezocht en zelf de juiste mensen gevonden om de kennis te krijgen die ik nodig had.

Wat ik ook geleerd heb tijdens dit project is dat je niet zomaar er vanuit kan gaan dat alles wat mensen zeggen waar is, ook niet binnen een bedrijf als Capgemini. Je moet het zelf zeker weten en gezien hebben. Zo is mij bij het definiëren van de opdracht gezegd dat er al een back-end systeem beschikbaar is en dat een Sybase Unwired Server ook geen probleem is. Uiteindelijk kom je er dan achter dat het back-end systeem niet meer draait en dat een alternatief systeem nog ingericht moet worden. Dit systeem zal eind van de zomer pas beschikbaar zijn. Ook krijg je te horen dat er eigenlijk nog geen Sybase Unwired Server is maar dat die wel op de planning staat om er te komen. Hierdoor is de opdracht uiteindelijk anders verlopen dan gepland. In het vervolg is het dus belangrijk om dit soort toezeggingen vanuit het bedrijf eerst goed zelf te onderzoeken voordat je er vanuit gaat dat het waar is.

Aan het begin van de afstudeerperiode heb ik een onderzoek gedaan naar Sybase Unwired Platform. Ik ben blij dat ik dit heb gedaan omdat dit een totaal nieuw product voor mij was. Hierdoor ben ik er achter gekomen dat het ontwikkelen van een applicatie iets meer met zich meebrengt dan ik in het begin dacht.

Tijdens het definiëren van de requirements vond ik het lastig om de verschillende experts te spreken. Ik wist van te voren dat je dit soort gesprekken goed moest inplannen omdat de experts vaak bij klanten zitten, maar ik had ook nog het probleem dat ik precies tijdens de vakantie periode aan het afstuderen was. Veel collega's waren daardoor tussentijds een aantal weken weg.

De planning die ik aan het begin gemaakt had is uiteindelijk flink veranderd. De activiteiten en producten zijn grotendeels wel gelijk gebleven maar de tijd die ik aan een aantal activiteiten besteedt heb is veel meer geworden dan verwacht. Vooral het ontwerpen van de applicatie, het kiezen en inrichten van het back-end systeem en beschikbaar krijgen van de Unwired Server heeft meer tijd in beslag genomen dan gepland. Dit komt omdat de benodigde systemen uiteindelijk niet direct beschikbaar waren.



Omdat vooral het proces voor de ontwikkeling lang heeft geduurd heb ik voortdurend de scope van het project aangepast. In het begin van het project wilde ik drie sprints doen en daarbij alle functionaliteiten in bouwen. Het installeren en onderzoeken van de verschillende omgevingen meer tijd in beslag heeft genomen dan ik dacht, heeft de applicatie minder functionaliteiten gekregen.

Door het definiëren van de requirements en het onderzoek naar Sybase Unwired Platform ben ik erachter gekomen dat de originele doelstelling niet juist is. De achterliggende doelstelling van deze opdracht is het onderzoeken hoe er applicaties ontwikkeld kunnen worden op het Sybase Unwired Platform.

Het gebruik van Scrum is goed bevallen tijdens het project. Ik heb door uitloop van het project niet de sprints kunnen doen die ik van te voren had gepland. Het grote voordeel van Scrum is dat de requirements tijdens de ontwikkeling nog kunnen veranderen. Dit is handig tijdens het ontwikkelen van een applicatie met een nieuw product als SUP. Hierbij weet je niet zeker of de requirements die je in het begin opstelt wel haalbaar zijn. Tijdens dit project heb ik dan ook het aantal requirements van de applicatie moeten beperken omdat opzetten van de infrastructuur meer tijd in beslag nam dan gepland.

Terugkijkend op de afstudeerperiode heb ik veel geleerd over het werken in een groot bedrijf. Ik heb geleerd om zelfstandig te werken en goed te communiceren met collega's uit binnen en buitenland. De gesprekken met experts zijn goed verlopen en uiteindelijk is iedereen tevreden over het resultaat, ondanks dat er nog geen compleet functionele applicatie is opgeleverd.

Tijdens het afstuderen heb ik ook veel geleerd over Sybase Unwired Platform en de techniek erachter. Ik ben nu een van de weinige mensen binnen Capgemini Nederland die ervaring heeft met het ontwikkelen van een applicatie op dit platform. Door de koppeling met het SAP systeem heb ik veel van SAP en inspectie management geleerd. Ik vind inspectie management een zeer interessant onderwerp en de kans is groot dat ik hier in de toekomst meer mee ga doen. Naast SUP en SAP heb ik ook ervaring op gedaan met het ontwikkelen van mobiele applicaties. Omdat mobiele applicaties in opkomst zijn denk ik ook hier nog veel gebruik van te gaan maken in de toekomst.

## 14.2 Product evaluatie

In de product evaluatie zullen de opgeleverde producten worden besproken en hoe deze tot stand zijn gekomen.

### 14.2.1 Plan van aanpak

In het begin van het project heb ik eerst een plan van aanpak gemaakt. Hierin staat beschreven hoe ik het project aangepakt heb. In dit plan van aanpak staat ook een eerste planning, deze is in de loop van het project gewijzigd. De meest recente versie staat in dit verslag. In dit plan van aanpak is extra aandacht besteed aan het opstellen de Dit heb ik gedaan omdat mijn vorige afstudeer opdracht mislukt is omdat het niet voldeed aan de randvoorwaarden. Het was belangrijk voor mij om zeker te weten dat het project aan deze randvoorwaarden kon voldoen en dat iedereen wist wat de randvoorwaarden waren.

Aan het begin van het project kon er nog niet worden voldaan worden aan de voorwaarden. Dit kwam omdat de benodigde systemen niet beschikbaar waren. Tijdens het project heb ik hier zelf aan gewerkt. Aan het eind van het project is wel aan voorwaarden voldaan. Het plan van aanpak heb ik opgesteld voor mijzelf zodat ik kon vastleggen wat er moest gebeuren en welke producten er opgeleverd moesten worden. Daarnaast was het voor het bedrijf goed om te weten welke producten en mensen ik nodig had voor het voltooien van het project.

### 14.2.2 Onderzoek rapport

Het onderzoeksrapport is uiteindelijk een zeer belangrijk document geworden. Ik heb hierin precies onderzocht hoe er een applicatie ontwikkeld kan worden met dit platform en waar rekening mee gehouden moet worden. Na dit onderzoek wist ik welke architectuurkeuzes er nog gemaakt moeten worden en welke het platform al voor mij maakt. Ik denk dat het ook goed is dat ik deze kennis heb vastgelegd in een onderzoeksrapport. Mensen die na mij een applicatie willen gaan ontwikkelen kunnen het rapport lezen en hiermee al een duidelijk beeld krijgen hoe dit in de praktijk werkt. Tijdens het onderzoek kwam ik erachter dat het ontwikkelen van een native applicatie niet zo eenvoudig is als dat ik van te voren verwacht had. Er is wel degelijk veel kennis nodig van programmeren en ontwikkelen om een goede applicatie te kunnen ontwikkelen.



### 14.2.3 Product Backlog

In het product backlog staan alle requirements die ik voor dit project gedefinieerd heb. Het definiëren van deze requirements vond ik een lastig onderdeel omdat het moeilijk is om één inspectie proces mobiel te maken. Iedere expert heeft een andere implementatie van het proces in zijn hoofd heeft hierdoor werd ik vaak verschillende richtingen opgestuurd. Ik heb in Excel een generiek proces beschreven dat ik mobiel kon maken. Al deze proces stappen heb beschreven en besproken met de experts. Ook heb ik de mobiele processtappen uitgewerkt in Use Case beschrijvingen.

Aan de hand van deze lijst is vervolgens het product backlog gemaakt. In dit product backlog heb ik ook een groot aantal techstories neergezet. Dit zijn stories die niet zozeer functionele meerwaarde hebben maar die wel op technisch gebied van groot belang zijn. Uiteindelijk denk ik dat het product backlog een goed beeld geeft van de functionaliteiten waaraan de applicatie moet voldoen.

### 14.2.4 Ontwerp

Met het ontwerp van de applicatie heb ik moeite gehad. Dit is om dezelfde reden als bij het product backlog, namelijk dat het lastig is om één generiek proces te definiëren. Daarnaast is bijvoorbeeld het maken van een datamodel zeer complex omdat SAP een groot en complex systeem is. Je kunt niet even eenvoudig opzoeken welke tabellen er in SAP zitten.

Door het doorlopen van het proces in SAP en door beschrijvingen van experts ben ik uiteindelijk tot de datamodellen gekomen. In eerste instantie lijkt je maar een klein aantal objecten nodig te hebben. Maar wanneer je dan verder gaat zoeken en vragen naar de achterliggende structuur kom je erachter dat er veel meer objecten en tabellen aan hangen dan je van tevoren verwacht.

Naast het datamodel heb ik ook een functioneel ontwerp gemaakt. Het was hierbij lastig om te bepalen welke gegevens wel in het mobiele scherm komen en welke niet. Omdat er in SAP veel gegevens in één scherm te zien zijn wordt het onoverzichtelijk. In een mobiele applicatie heb je die ruimte niet en moet je gaan selecteren wat je wel wilt zien en wat niet. Helemaal omdat ik nog niet heel veel ervaring heb met SAP is het lastig om de gegevens te selecteren.

### 14.2.5 Applicatie

Tijdens dit project heb ik een mobiele applicatie ontwikkeld. Deze applicatie is uiteindelijk niet helemaal afgekomen omdat ik te weinig tijd had. Dit komt omdat ik veel tijd heb besteed aan onverwachte activiteiten zoals het uitkiezen van het back-end systeem, beschikbaar krijgen van de Sybase Unwired Server en het duidelijk krijgen van het inspectie proces. Met de applicatie is het wel mogelijk om verbinding te maken met het SAP back-end en om een lijst met inspecties op te vragen. Ook kunnen er bijbehorende orders gevonden worden. Verder ben ik een heel eind gekomen met het invoeren van resultaten. Het afmaken van de applicatie is dan ook een kwestie van tijd.

Ik vind het jammer dat het niet is gelukt om de hele applicatie af te maken. Wel is er een mogelijke klant voor de applicatie en krijg ik misschien nog de gelegenheid om de applicatie voor een echte klant verder te bouwen.

### 14.2.6 Advies

Omdat de doelstelling van de opdracht gewijzigd is heb ik een advies uitgebracht voor Capgemini over het gebruik van Sybase Unwired Platform en het ontwikkelen van applicaties met dit platform. Met dit advies deel ik mijn ervaringen over het ontwikkelen van de applicatie. Dit advies kunnen collega's in de toekomst gebruiken om te bepalen of ze dit platform willen gebruiken voor het ontwikkelen van mobiele applicaties. Ook kunnen ze met dit advies bepalen of dit platform geschikt is voor een specifieke klant. Ik denk dat het advies heel waardevol is voor Capgemini omdat het een van de weinige ervaringen is die Capgemini heeft met het Sybase Unwired Platform.

## 15. Aantonen beroepstaken

Aan het begin van deze afstudeeropdracht heb ik een aantal beroepstaken geselecteerd waarvan ik wilde aantonen het gewenste niveau bereikt te hebben. Een van deze beroepstaken was het ontwerpen systeemdeel (3.2) . Door het wijzigen van de doelstelling is deze beroepstaak niet op het gewenste niveau gehaald. In plaats hiervan is meer aandacht besteed aan het voorbereiden en opstarten van het softwareontwikkeltraject (1.2).

### 15.1 Voorbereiden en opstarten softwareontwikkeltraject

Beroepstaak 1.2 is het voorbereiden en opstarten van het softwareontwikkeltraject. Ik vind dat ik hieraan heb voldaan omdat er duidelijk is onderzocht en beschreven wat het probleem is, wat de oplossing is en hoe deze oplossing gebouwd kan worden. Ik heb met meerdere experts de requirements besproken en een oplossing beschreven die rekening houdt met de verschillende eisen en wensen. Ook heb ik onderzocht en beschreven met welke software de applicatie gebouwd kan worden en volgens welke methode. Tijdens het onderzoek heb ik beschreven waarmee rekening moet worden gehouden bij het bouwen van de applicatie en hoe er een koppeling gerealiseerd kan worden met SAP. In dit project heb ik er ook voor gezorgd dat er een geschikt back-end systeem beschikbaar is om de koppeling te maken, dat er een Unwired Server draait en dat alle ontwikkelomgevingen geïnstalleerd en geconfigureerd zijn.

### 15.2 Uitvoeren analyse door definitie van requirements

Beroepstaak 1.4 is het uitvoeren van analyse door definitie van requirements. Ik vind dat ik hieraan voldaan heb omdat ik verschillende eisen en wensen van de verschillende experts heb weten samen te brengen tot één lijst met requirements voor een generiek inspectie management proces. Deze requirements heb ik besproken met de verschillende experts en vastgelegd in Use Cases. Dit is complex geweest omdat de requirements van het inspectieproces heel erg verschillend kunnen zijn. Het inspectie management proces in SAP is erg ingewikkeld en kan op meerdere manieren geïmplementeerd worden. Het is dan ook erg lastig om er één geschikt proces uit te definiëren.

### 15.3 Ontwerpen software architectuur

Beroepstaak 3.1 is het ontwerpen van een software architectuur. Ik vind dat ik hieraan voldaan heb omdat ik een onderzoek gedaan heb naar Sybase Unwired Platform. Hierin heb ik beschreven hoe dit platform in elkaar zit en hoe er applicaties mee ontwikkeld kunnen worden. Ik ben gedetailleerd ingegaan op de architectuur van Sybase en welke keuzes er gemaakt moeten worden. Daarnaast onderzoek gedaan naar verschillende subsystemen en componenten van Sybase. Ook heb ik de omgeving waarin de applicatie draait en de verschillende benodigde subsystemen beschreven en geïnstalleerd. Verder heb ik met behulp van de requirements een aantal datamodellen en een functioneel ontwerp van de architectuur beschreven.

### 15.4 Bouwen applicatie

Beroepstaak 3.3 is het bouwen van de applicatie. Ik vind dat ik hieraan voldaan heb omdat ik een nieuwe mobiele applicatie ontwikkeld heb. Dit betekent dat ik alle functionaliteiten zelf moest realiseren. Ik heb eerst een ontwerp gemaakt en requirements opgesteld. Vanuit deze gegevens heb ik met behulp van JAVA een Android applicatie ontwikkeld in Eclipse. Ik heb hierbij gebruik gemaakt van design patterns en een koppeling gerealiseerd met een SAP back-end. Het was de eerste keer voor mij dat ik een mobiele applicatie heb ontwikkeld en heb hier veel van geleerd. Nadat ik de laatste tijd vooral gewend was om met C# te programmeren vond ik het ook interessant om in JAVA iets te bouwen. Helaas is het niet gelukt om binnen de tijd de applicatie helemaal af te maken. Wel is het gelukt om een goede indruk te geven wat SUP allemaal kan. Hopelijk kan ik hiermee een klant overtuigen om in de toekomst de applicatie verder te ontwikkelen.

## 16. Literatuurlijst

Sybase (2010). Sybase Unwired Platform Mobile Enterprise Application Platform.  
*Sybase Report door Sybase*

Sybase (2011-2). Sybase Unwired Platform Version 2.0.  
*Sybase Report door Sybase*

Sybase (2011-3). Sybase Unwired Platform Development Paradigm.  
*Sybase Report door Sybase*

Sybase (2011-4). Sybase Unwired Platform Security.  
*Sybase Report door Sybase*

Sybase(2012). Android Object API Application Development  
*Sybase Report door Sybase*

EMF (2010). Considerations for Mobile Enterprise Application Development, Deployment and Management.  
*The Enterprise MobilityFoundation*

Taplidou, S. (2012), Mobile Solutions with Sybase Unwired Platform and SAP NetWeaver Gateway  
*Consultant CoE MobileSolution SAP Switzerland*

Erk, R. (2012), Mobile Enterprise Application Platform – From Initiation to the start of integration  
*University of Amsterdam & Capgemini*

Infuze Mobile (2011). Advantages and Disadvantages of MEAPs.

AT&T (2010). Mobilizing Enterprise Applications.  
*AT&T*

Malik, A. (2011). Dissertation SAP Enterprise Mobility, A Market Audit.  
*University of Applied Science & Capgemini*

## More Information



### About Capgemini

With more than 115,000 people in 40 countries, Capgemini is one of the world's foremost providers of consulting, technology and outsourcing services. The Group reported 2010 global revenues of EUR 8.7 billion.

Together with its clients, Capgemini creates and delivers business and technology solutions that fit their needs and drive the results they want.

A deeply multicultural organization, Capgemini has developed its own way of working, the Collaborative Business Experience™, and draws on Rightshore®, its worldwide delivery model.

Learn more about us at:

[www.capgemini.com](http://www.capgemini.com).

*Rightshore® is a trademark belonging to Capgemini.*



**People matter, results count.**

**[www.capgemini.com](http://www.capgemini.com)**

The information contained in this presentation is proprietary and confidential. ©2011 Capgemini. All rights reserved

# Mobiel inspectie management

## *Bijlagen*

Naam: Jerry van Holst

Studentnummer: 08024146

Datum: 24 september 2012



# Bijlagen

---

<b>Bijlage A:</b>	<b>Afstudeerplan</b>
<b>Bijlage B:</b>	<b>Plan van aanpak</b>
<b>Bijlage C:</b>	<b>Inspectie proces beschrijving</b>
<b>Bijlage D:</b>	<b>Product Backlog</b>
<b>Bijlage E:</b>	<b>Ontwerp</b>



# Bijlage A: Afstudeerplan

---

*Mobiel Inspectie Management*

Naam: Jerry van Holst  
Studentnr: 08024146  
Datum: 7-5-2012

## Afstudeerplan

Informatie afstudeerder en gastbedrijf

Afstudeerblok: 2012-1.2 (start uiterlijk 7 mei 2012)

Startdatum uitvoering afstudeeropdracht: 7 mei 2012

Inleverdatum afstudeerdossier volgens jaarrooster: 24 september 2012

Studentnummer: 08024146

Achternaam: dhr van Holst

Voorletters: J.A.

Roepnaam: Jerry

Adres: Molenstraat 73

Postcode: 2681BN

Woonplaats: Monster

Telefoonnummer: 0174-298396

Mobiel nummer: 0643760005

Privé emailadres: jerryvanholst@hotmail.com

Opleiding: Informatica

Locatie: Den Haag

Variant: voltijd

Naam studieloopbaanbegeleider: O. Zor

Naam begeleidend examiner: A. Lousberg

Naam tweede examiner: O. Zor

Naam bedrijf: Capgemini

Afdeling bedrijf: ICT

Bezoekadres bedrijf: Papendorpseweg 100

Postcode bezoekadres: 3528 BJ

Postbusnummer:

Postcode postbusnummer:

Plaats: Utrecht

Telefoon bedrijf: 030689 0000

Telefax bedrijf: 030689 9999

Internetsite bedrijf: <http://www.nl.capgemini.com/>

Achternaam opdrachtgever: dhr BarkMeijer

Voorletters opdrachtgever: M

Titulatuur opdrachtgever:

Functie opdrachtgever: Business Analyst

Doorkiesnummer opdrachtgever: 06-51117003

Email opdrachtgever: menno.barkmeijer@capgemini.com

Achternaam bedrijfsmentor: mw Hoekstra

Voorletters bedrijfsmentor: F.

Titulatuur bedrijfsmentor:

Functie bedrijfsmentor: SAP Solutions

Doorkiesnummer bedrijfsmentor: 030 6895342

Email bedrijfsmentor: femke.hoekstra@capgemini.com

Doorkiesnummer afstudeerder:

Functie afstudeerder (deeltijd/duaal):

Ontwikkelen van een mobiele inspectie applicatie

# Opdrachtschrijving

## Bedrijf

Capgemini ontwerpt en integreert technologische oplossingen, ontwikkelt innovaties en transformeert de technische omgevingen van klanten. Deze diensten concentreren zich met name op systeemarchitectuur, -integratie en -infrastructuur. Capgemini werkt daarbij samen met strategische partners om de issues en uitdagingen van klanten slagvaardig en met alle benodigde expertises te kunnen aanpakken.

De afstudeeropdracht wordt uitgevoerd binnen de divisie SAP Solutions. Capgemini investeert proactief in de ontwikkeling van nieuwe dienstverlening en samen met SAP is Capgemini ontwikkelaar van nieuwe producten en methoden. Capgemini is al vijftien jaar SAP-partner en heeft meer dan 3000 SAP- projecten voor 1500 klanten uitgevoerd.

## Probleemstelling

In de huidige SAP systemen kan een Quality Management module ingebouwd worden. Met behulp van deze module is het mogelijk om de uitvoer van inspecties te administreren. Bij inspecties moet gedacht worden aan het controleren van grote machines op eventuele gebreken. Een inspecteur bekijkt het inspectieplan, voert de inspectie uit en noteert de resultaten op een papier. Later moeten deze resultaten ingevoerd worden in het systeem. Bij een product zijn er van te voren bepaalde kwaliteitseisen vastgesteld waar het product aan moet voldoen. Met behulp van deze inspecties wordt bepaald of het product aan die eisen voldoet of niet.

Het probleem wat men op dit moment heeft is dat de inspecteur de machine controleert, en deze gegevens noteert op een papier. Nadat de inspectie voltooid is gaat de inspecteur naar een computer met toegang tot het systeem en moet vervolgens deze informatie opnieuw invoeren. Hierbij gaat veel tijd verloren.

## Doelstelling van de afstudeeropdracht

Het ontwikkelen van een mobiele applicatie waarmee inspecteurs de resultaten van hun inspectie direct kunnen invoeren in het huidige SAP systeem.

Er zullen requirements achterhaald moeten worden bij klanten en interne experts om er achter te komen wat de behoeftes zijn. Dit zal gebeuren door middel van een aantal interviews. Uiteindelijk moet er een ideale mix gevonden worden om al deze behoeftes te realiseren in de applicatie. Door de verschillende stakeholders binnen deze opdracht ligt hierin een stukje complexiteit.

Bij het opstellen van deze requirements hoort ook de keuze of het een native of web-based applicatie gaat worden. Dit hangt af van de wensen van de klant. Wanneer er gekozen wordt voor een native applicatie, komt er ook nog een keuze voor een besturingssysteem. Dit kunnen bijvoorbeeld Android, Blackberry of iOS zijn. Om het systeem duidelijk te beschrijven en ontwerpen zal er een architectuur document gemaakt worden.

Om de applicatie te laten samenwerken met SAP moet er een koppeling komen tussen de Quality Management module en het Sybase Unwired Platform (SUP). SUP is een onderdeel van SAP om de zakelijke toepassing van mobiele oplossingen te managen. Het functioneert als een tussenlaag voor mobiele apparaten en het SAP systeem. Binnen Capgemini is er een demo-omgeving van SUP beschikbaar.

Bij deze opdracht zal de integratie met SUP en de SAP QM module beschreven worden. Omdat het een vrij nieuwe omgeving is zal deze integratie eerst onderzocht moeten worden en daarna gerealiseerd. Het is op dit moment nog niet te zeggen in hoeverre het realiseren van de integratie mogelijk is binnen de gestelde tijd. Er zal een Proof of Concept van de koppeling komen om aan te tonen dat de koppeling zoals die is beschreven werkt. Bij deze koppeling ligt dan ook een stukje complexiteit van deze opdracht.

## Resultaat

Het resultaat is een mobiele applicatie met grote meerwaarde voor de gebruikers van de applicatie:

- Wettelijke bepalingen: Bij juridische geschillen moet er aangetoond kunnen worden dat bepaalde producten na de service- en onderhoudsbeurt correct zijn overgedragen;
- Veiligheid: Meerdere en gestandaardiseerde inspecties kunnen de veiligheid verhogen;
- Inspecteur efficiency: Metingen hoeven maar één keer ingevoerd te worden. Metingen noteren en later overtypen is verleden tijd. Fouten worden op deze manier voorkomen;
- Onderhoud efficiency: Door inspecties kan het preventieve onderhoud beter worden geanalyseerd en geoptimaliseerd.

Daarnaast zal het ook een meerwaarde voor Capgemini zijn omdat er onderzocht wordt hoe er een koppeling realiseerbaar is tussen de applicatie, Sybase Unwired Platform en SAP QM.

## Uit te voeren werkzaamheden, inclusief een globale fasering, mijlpalen en bijbehorende activiteiten

Activiteiten	Op te leveren producten	Geschatte Tijd
Maken Plan van aanpak	Plan van Aanpak	5 dagen
Inlezen inspectie management en Sybase		5 dagen
Onderzoeken technische mogelijkheden		5 dagen
Onderzoeken beschrijven integratie	Beschrijving integratie	10 dagen
Opstellen requirements	Product backlog	15 dagen
Ontwerpen Architectuur	Architectuur document	10 dagen
3 Sprints:		
Maken ontwerp	Ontwerp	15 dagen
Maken systeemdeel	Applicatie	15 dagen
Bouwen PoC integratie	PoC integratie	10 dagen
Opstellen testplan	Testplan	3 dagen
Testen	Test rapport	7 dagen

## Op te leveren (tussen)producten

- **Plan van aanpak**  
In dit plan wordt beschreven hoe het project aangepakt wordt en waar rekening mee gehouden moet worden.
- **Beschrijving integratie**  
In dit rapport zal onderzocht en beschreven worden hoe er een integratie gerealiseerd kan worden tussen de mobiele applicatie en het huidige SAP systeem.
- **Product backlog**  
Hierin staan de requirements van het systeem beschreven, ook wel user stories genoemd.  
  
Deze staan op volgorde van belangrijkheid. En elke requirement krijgt een geschatte waarde van moeilijkheid.
- **Architectuur document**  
In dit document zal de architectuur van mijn applicatie beschreven worden.
- **Ontwerp**  
Het ontwerp van het systeem, denk hierbij aan proces flow, klassendiagram, sequence diagrammen en eventueel use cases.
- **Systeem**  
De mobiele applicatie die uiteindelijk gebouwd gaat worden aan de hand van de requirements en het ontwerp.
- **PoC integratie**  
Met deze Proof of Concept van de integratie wordt aangetoond dat de integratie die is onderzocht/beschreven ook echt werkt. Dit zal gebeuren door data te laten communiceren tussen front-end en back-end.
- **Testplan**  
In het test plan wordt beschreven wat er getest moet worden en hoe dat gedaan wordt. Dit wordt gedaan om te bepalen of het systeem aan de eisen voldoet.
- **Testrapport**  
Het systeem zal uiteindelijk getest worden volgens het testplan.

## Te demonstreren competenties en wijze waarop

Beroepstaak	Product	Niveau
1.4 Uitvoeren analyse door definitie van requirements	- Product backlog	4
3.1 Ontwerpen Softwarearchitectuur	- Architectuur doc.	3
3.2 Ontwerpen Systeemdeel	- Ontwerp - PoC	3
3.3 Bouwen applicatie	- Systeem - PoC	3

# Bijlage B: Plan van aanpak

---

*Mobiel Inspectie Management*

Naam: Jerry van Holst Studentnr: 08024146 Datum: 14-5-2012
--

# Inhoud

<b>1.</b>	<b>Inleiding.....</b>	<b>9</b>
<b>2.</b>	<b>Bedrijf.....</b>	<b>9</b>
<b>3.</b>	<b>Opdracht .....</b>	<b>9</b>
	Probleemstelling .....	9
	Doelstelling.....	10
	Resultaat.....	10
<b>4.</b>	<b>Aanpak .....</b>	<b>11</b>
	Onderzoek aanpak .....	11
	Ontwikkelmethode.....	12
<b>5.</b>	<b>Randvoorwaarden .....</b>	<b>13</b>
<b>6.</b>	<b>Risicofactoren .....</b>	<b>14</b>
<b>7.</b>	<b>Benodigde mensen en middelen.....</b>	<b>14</b>
<b>8.</b>	<b>Planning.....</b>	<b>15</b>
<b>9.</b>	<b>Op te leveren (tussen)producten .....</b>	<b>16</b>



# Inleiding

In het kader van mijn afstuderen bij de opleiding informatica op de Haagse Hogeschool loop ik een afstudeerstage bij Capgemini. Het doel van deze afstudeeropdracht is het ontwikkelen van een mobiele applicatie op het gebied van inspectie management. Deze applicatie zal gekoppeld worden met sybase en een back-end systeem van SAP.

In het plan van aanpak beschrijf ik de afstudeer opdracht en geef ik duidelijkheid in hoe ik het project aan ga pakken.

# Bedrijf

Capgemini ontwerpt en integreert technologische oplossingen, ontwikkelt innovaties en transformeert de technische omgevingen van klanten. Deze diensten concentreren zich met name op systeemarchitectuur, - integratie en -infrastructuur. Capgemini werkt daarbij samen met strategische partners om de issues en uitdagingen van klanten slagvaardig en met alle benodigde expertises te kunnen aanpakken.

De afstudeeropdracht wordt uitgevoerd binnen de divisie SAP Solutions. Capgemini investeert proactief in de ontwikkeling van nieuwe dienstverlening en samen met SAP is Capgemini ontwikkelaar van nieuwe producten en methoden. Capgemini is al vijftien jaar SAP-partner en heeft meer dan 3000 SAP- projecten voor 1500 klanten uitgevoerd.

# Opdracht

## Probleemstelling

In de huidige SAP systemen kan een Quality Management module ingericht worden. Met behulp van deze module is het mogelijk om de uitvoer van inspecties te administreren. Bij inspecties moet gedacht worden aan het controleren van grote machines op eventuele gebreken. Een inspecteur bekijkt het inspectieplan, voert de inspectie uit en noteert de resultaten op papier. Later moeten deze resultaten ingevoerd worden in het systeem. Voor elk product zijn er van te voren kwaliteitseisen vastgesteld waar het product aan moet voldoen. Met behulp van deze inspecties wordt bepaald of het product aan die eisen voldoet.

Het probleem wat men op dit moment heeft is dat de inspecteur de machine controleert, en deze gegevens noteert op papier. Nadat de inspectie voltooid is gaat de inspecteur naar een computer met toegang tot het systeem en moet vervolgens deze informatie invoeren. Hij moet de resultaten dus dubbel registreren. Hierbij gaat veel tijd verloren en is een verhoogde kans op fouten door verkeerd overnemen.

Een ander probleem is dat wanneer de inspectieresultaten afwijken van de norm er een reparatie melding gedaan moet worden. Op dit moment gebeurt dit nog handmatig, en moet alles overgetypt worden in het SAP systeem.

## Doelstelling

Het doel van de opdracht is het ontwikkelen van een mobiele applicatie die het inspectie proces in SAP ondersteund. In de applicatie moet het mogelijk zijn voor inspecteurs om snel en eenvoudig de geplande inspecties in te zien en de resultaten hiervan invoeren in het huidige SAP back-end systeem. Door deze mobiele applicatie moet het inspectie proces efficiënter worden, sneller en met minder fouten.

Om de applicatie te laten samenwerken met SAP moet er een koppeling komen tussen het inspectie proces in SAP en het Sybase Unwired Platform (SUP). SUP is sinds kort onderdeel van SAP en dient om de zakelijke toepassing van mobiele oplossingen te managen. Het functioneert als een tussenlaag voor mobiele apparaten en het SAP systeem. Dit soort middleware wordt ook wel een Mobile Enterprise Application Platform (MEAP) genoemd.

Sybase Unwired platform is een nieuw product, er is dan ook nog weinig praktische kennis beschikbaar binnen Capgemini. Bij het bouwen en ontwerpen van de applicatie zullen er dan ook veel vragen optreden. Omdat de kennis er niet is, zal er zelf naar oplossingen gezocht moeten worden. Er zal dan ook eerst een onderzoek komen naar Sybase Unwired Platform om erachter te komen waar rekening mee gehouden moet worden en wat de valkuilen zijn. Het ontwikkelen van een applicatie met Sybase Unwired Platform zal een uitdaging worden en hier ligt ook een stukje complexiteit van de opdracht.

## Resultaat

Het resultaat is een mobiele applicatie waarmee de inspecteur sneller en efficiënter zijn inspecties kan uitvoeren. De resultaten worden automatisch in het SAP systeem verwerkt en hoeft dus niet meer apart door een inspecteur gedaan te worden. Deze inspectie applicatie heeft een grote meerwaarde voor de gebruikers:

- Wettelijke bepalingen: Bij juridische geschillen moet er aangetoond kunnen worden dat bepaalde producten na de service- en onderhoudsbeurt correct zijn overgedragen;
- Veiligheid: Meerdere en gestandaardiseerde inspecties kunnen de veiligheid verhogen;
- Inspecteur efficiency: Metingen hoeven maar één keer ingevoerd te worden. Metingen noteren en later overtypen is verleden tijd. Fouten worden op deze manier voorkomen.
- Onderhoud efficiency: Door inspecties kan het preventieve onderhoud beter worden geanalyseerd en geoptimaliseerd.

Deze opdracht zal ook een meerwaarde zijn voor Capgemini. Er wordt onderzocht hoe er een applicatie ontwikkeld kan worden met behulp van het Sybase Unwired Platform. Het wordt ook duidelijk hoe er een koppeling realiseerbaar is tussen de mobiele applicatie, Sybase Unwired Platform en een SAP back-end. Bovendien wordt er praktische ervaring opgedaan op het gebied van Sybase Unwired platform. Het is dan ook belangrijk om dit goed te kunnen overdragen aan het einde van het project.

# Aanpak

Om het project tot een goed einde te brengen is het van belang om van te voren goed na te denken over de aanpak. In dit hoofdstuk beschrijf ik hoe ik het onderzoek naar Inspectie management en het Sybase Unwired Platform gedaan heb en welke ontwikkelmethode ik heb gebruikt voor het ontwikkelen van de mobiele applicatie. Daarnaast zal ik ook de planning en de op te leveren producten beschrijven.

## Onderzoek aanpak

Voordat ik de requirements kan gaan opstellen en de applicatie kan gaan ontwikkelen moet ik meer kennis hebben van inspectie management in SAP en Sybase Unwired Platform. Ik ga daarom eerst een onderzoek doen naar inspectie management in SAP en daarna onderzoeken wat het Sybase Unwired platform is en hoe er applicaties mee ontwikkeld kunnen worden.

Het onderzoek naar inspectie management ga ik doen behulp van interne experts. Hierdoor ga ik dit proces leren begrijpen en weet ik wat het doel van is inspectie management.

Het onderzoek naar Sybase Unwired Platform wordt een literatuuronderzoek, dit houdt in dat ik alle informatie vooral uit bestaande literatuur ga halen. Dit kunnen bijvoorbeeld wetenschappelijke artikelen op internet, boeken of bestaande onderzoek verslagen zijn. Aan de hand van deze informatie kan ik vervolgens eigen conclusies trekken.

Voor het zoeken van informatie op internet zal ik gebruik maken van de zoekmachine "Google", de wetenschappelijke artikelen via zoekmachine "Google Scholar" en de technische documenten via de website van Sybase. Als zoekmethode zal ik gebruik maken van de sneeuwbal methode. Dit houdt in dat je bij de inleidende tekst over het onderwerp kijkt in de literatuurlijst. Die artikelen zoek je op, en gebruik je waar nodig. Ook hier kijk je weer in de literatuur lijst en ook dit gebruik je waar nodig enzovoort. Hierdoor wordt de lijst van literatuur steeds groter (net als een sneeuwbal). Het is belangrijk om hier in de gaten te houden wat de hoofdvraag is omdat je snel geneigd bent om alles uit te zoeken.

De eerste stap is het opstellen van de onderzoeksvraag. Het doel van dit onderzoek is erachter komen wat het Sybase Unwired platform is en hoe er applicaties mee ontwikkeld kunnen worden. Ook hoe er een koppeling met een back-end systeem gemaakt kan worden en of het uiteindelijk een goede keuze is om het Sybase Unwired Platform te gebruiken voor het bouwen van deze applicatie. Vandaar dat de hoofdvraag luidt:

"Wat is Sybase Unwired Platform en hoe kun je er aan SAP gekoppelde mobiele applicaties mee ontwikkelen?"

Aan de hand hiervan heb ik een aantal deelvragen opgesteld. Via deze deelvragen kan ik antwoord geven op de hoofdvraag. De deelvragen zijn opgesteld door de hoofdvraag op te splitsen in drie delen. Het eerste deel gaat over SUP en wat het kan, het tweede deel over hoe ik de koppeling kan maken met SAP en in het derde deel hoe er een mobiele applicatie ontwikkeld kan worden.

De volgende deelvragen zijn opgesteld:

1. Wat is Sybase Unwired Platform?
2. Wat is een Mobile Enterprise Application Platform (MEAP)?
3. Wat zijn de voordelen van MEAP?
4. Wat zijn de nadelen van MEAP?
5. Hoe kun je Sybase Unwired Platform koppelen met een SAP back-end?
6. Hoe kunnen er applicaties ontwikkeld worden met Sybase Unwired Platform?
7. Welke architectuur keuzes moeten er gemaakt worden?

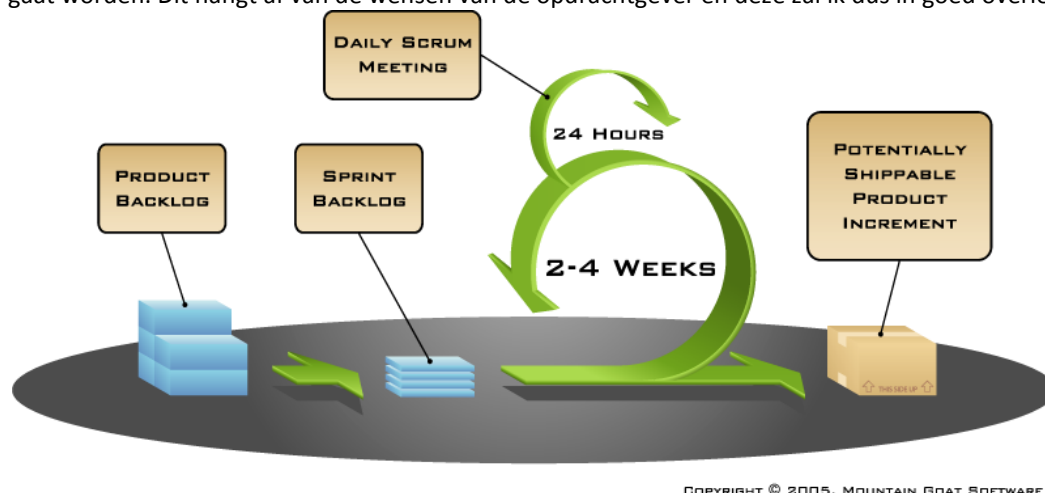
## Ontwikkelmethode

Voor het ontwikkelen van de applicatie ga ik gebruik maken van de SCRUM methode. Dit is een Agile methode waarbij in korte sprints werkende software wordt opgeleverd. Scrum is zeer goed geschikt voor projecten die niet van te voren helemaal uit te schrijven zijn. De wensen en eisen zijn namelijk gedurende het hele project nog aan te passen. Scrum is gekozen voor dit project omdat ik gewerkt heb met een vrij nieuw product en daarom wist ik van te voren nog niet precies wat ik allemaal tegen zou komen en wat de mogelijkheden zijn.

Bij scrum wordt er van te voren een product backlog gemaakt. Hierin staan alle functionele eisen van het systeem gesorteerd op prioriteit. De eis met de hoogste prioriteit staat bovenaan en zal als eerste ontwikkeld worden. Zodra alle eisen duidelijk zijn wordt er een sprint backlog gemaakt. Hierin staan alle eisen die in de betreffende sprint gerealiseerd zullen worden. Bij elke sprint zal er een werkend stukje software worden opgeleverd en gepresenteerd.

Om er achter te komen wat de functionele eisen zijn van het systeem zal ik eerst kennis inwinnen van interne experts op het gebied van inspectie management. Zij weten waar de behoefte op de markt ligt en wat de bottlenecks zijn in het huidige inspectie proces. Deze experts zal ik zelf zoeken en benaderen. Omdat Capgemini een detacheer bedrijf is zitten de experts vaak bij een klant voor een project. Het is daarom van belang om ruim van te voren afspraken te plannen met de experts.

Daarnaast moet ik een aantal architectuur keuzes maken. Bijvoorbeeld de keuze of het een native of web-based applicatie gaat worden. Dit hangt af van de wensen van de opdrachtgever en deze zal ik dus in goed overleg moeten maken.



Figuur 1.1: Scrum

Bij scrum ligt de focus vooral op het samenwerken maar omdat dit een individuele afstudeer opdracht is, kan er niet gewerkt worden in een team. Wel zijn er andere studenten op de afdeling aan een zelfstandig project bezig. Om toch een moment te creëren waarin de problemen besproken kunnen worden zal er een wekelijkse scrum meeting zijn. Hierin word er kort verteld wat er gedaan is, wat je gaat doen en of er nog problemen zijn. Er is een project manager aangesteld vanuit Capgemini. Zij zal de scrum meetings begeleiden. Ook kan ik bij de project manager terecht voor afstudeer gerelateerde vragen.

Daarnaast zal er wekelijks contact zijn met mijn inhoudelijke begeleider om de vorderingen van het project te bespreken. Ook is dat het moment om problemen te bespreken en vragen te stellen. Tijdens deze afspraken worden ook de producten opgeleverd.

# Randvoorwaarden

Randvoorwaarden voor het succesvol afronden van dit project.

- **Geschikte Hardware**  
Er moet geschikte hardware zoals computers, printers etc aanwezig zijn zodat ik alle noodzakelijk programma's kan draaien en mijn producten kan uitprinten.
- **Werkplek**  
Om tot goede producten te komen is het noodzakelijk om een goede werkplek te hebben waar geconcentreerd gewerkt kan worden.
- **Toegang tot een ontwikkelomgeving**  
Afhankelijk van de soort applicatie(native of web based) zal er in eclipse of visual studio worden ontwikkeld.
- **Toegang tot de Sybase Demo omgeving**  
Om een integratie met Sybase te kunnen realiseren zal er een demo omgeving van sybase beschikbaar moeten zijn.
- **Toegang tot een back-end omgeving**  
Om een integratie met een back-end systeem (SAP) te kunnen realiseren zal ook hier een demo systeem van beschikbaar moeten zijn.
- **Technische ondersteuning voor sybase**  
Om het systeem te begrijpen en te kunnen programmeren is er technische ondersteuning nodig, iemand die mij kan helpen met problemen en tips kan geven. En eventueel een Cursus
- **Inhoudelijke ondersteuning inspectie management**  
Voor de vragen over inspectie management en dit proces zal ook hier ondersteuning voor nodig zijn.

# Risicofactoren

De risicofactoren worden behandeld in het personeelsreglement van het bedrijf. Hierin staan regels voor bijvoorbeeld ziekte of bedrijfsongevallen. Hieronder staan een aantal andere risicofactoren die tijdens dit project kunnen voorkomen.

- **Opdrachtgever onbereikbaar**  
Wanneer de opdrachtgever onbereikbaar is kan het voorkomen dat ik langer op een antwoord zal moeten wachten. Dit kan resulteren in het uitlopen van de planning.
- **Geen werkruimte beschikbaar**  
Vanwege de drukte in het gebouw kan het voorkomen dat er geen geschikte werkplek beschikbaar is.
- **Interne experts geen tijd op korte termijn**  
Voor het opstellen van de requirements heb ik input nodig van interne experts. Het kan voorkomen dat deze experts op korte termijn geen tijd hebben om deze input te geven.
- **Verlaat toegang tot Sybase**  
Om uiteindelijk een integratie te kunnen realiseren tussen de front-end en back-end is sybase nodig. Het kan voorkomen dat ik, door omstandigheden, verlaat toegang krijg tot dit systeem.
- **Verlaat toegang tot het Back-end systeem**  
Er is een back-end systeem nodig om een proof of concept te maken van de integratie. Zodat er daadwerkelijk wat data gecommuniceerd kan worden.
- **Te weinig technische kennis van Sybase**  
Er moet een onderzoek en PoC komen gebaseerd op Sybase. Het kan voorkomen dat Sybase te complex is om zonder voldoende technische kennis een integratie te ontwikkelen.
- **Te weinig technische kennis van het ontwikkelen van de front-end**  
Er moet een front-end gebouwd worden. Dit gebeurt web-based of native. Er kan te weinig kennis zijn van het ontwikkelen van dit front-end.

# Benodigde mensen en middelen

De benodigde mensen en middelen voor het succesvol afronden van dit project

- Begeleider voor inhoudelijke vragen over inspectie management
- Begeleider voor algemene vragen over documenten en afstudeer zaken.
- Begeleider met technische kennis van Sybase
- Verschillende interne experts/opdracht gevers voor het opstellen van requirements
- Een aantal externe klanten voor het valideren van de requirements
  
- Toegang tot Sybase demo systeem
- Toegang tot een back end demo systeem
- Toegang tot ontwikkelomgeving
- Cursus Sybase
- Geschikte Hardware
- Werkplek

# Planning

Week→	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
Activiteit ↓																	
Oriëntatie																	
- Oriëntatiefase																	
- Plan van Aanpak																	
Back-end Systeem Kiezen																	
- Back-end systeem																	
Inlezen inspectie management																	
- Onderzoeken inspectie management																	
Onderzoek Sybase Unwired Platform																	
- Onderzoeksrapport																	
Opstellen Requirements																	
- Proces flow																	
- Requirements																	
Opstellen Architectuur																	
- Architectuur																	
Sprint 1: gereed maken systeemdeel																	
- Klaarzetten onderdelen																	
- testen																	
Sprint 2: Bouwen basis applicatie																	
- Ontwerp																	
- Applicatie																	
- testen																	
Sprint 3: Bouwen applicatie																	
- ontwerp																	
- applicatie																	
- testen																	
Afronden Scriptie																	
- Eindverslag																	
Schrijven Scriptie																	



# Op te leveren (tussen)producten

- **Plan van aanpak**  
In dit plan wordt beschreven hoe het project aangepakt wordt en waar rekening mee gehouden moet worden.
- **Beschrijving integratie**  
In dit rapport zal onderzocht en beschreven worden hoe er een integratie gerealiseerd kan worden tussen de mobiele applicatie en het huidige SAP systeem.
- **Product backlog**  
Hierin staan de requirements van het systeem beschreven, ook wel user stories genoemd. Deze staan op volgorde van belangrijkheid. En elke requirement krijgt een geschatte waarde van moeilijkheid.
- **Architectuur document**  
In dit document zal de architectuur van mijn applicatie beschreven worden.
- **Ontwerp**  
Het ontwerp van het systeem, denk hierbij aan proces flow, klassendiagram, sequence diagrammen en eventueel use cases.  
**Systeem**  
De mobiele applicatie die uiteindelijk gebouwd gaat worden aan de hand van de requirements en het ontwerp.
- **PoC integratie**  
Met deze Proof of Concept van de integratie wordt aangetoond dat de integratie die is onderzocht/beschreven ook echt werkt. Dit zal gebeuren door data te laten communiceren tussen front-end en back-end.
- **Testplan**  
In het test plan wordt beschreven wat er getest moet worden en hoe dat gedaan wordt. Dit wordt gedaan om te bepalen of het systeem aan de eisen voldoet.
- **Testrapport**  
Het systeem zal uiteindelijk getest worden volgens het testplan.

# Bijlage C: Inspectie proces beschrijving

---

*Mobiel Inspectie Management*

Naam: Jerry van Holst  
Studentnr: 08024146  
Datum: 10-7-2012

# Inhoud

Proces stappen beschrijving .....	19
Stappen geschikt voor mobiel .....	20
Proces flows .....	21

## Proces stappen beschrijving

Condition Monitoring					
Proces	Proces Step	Actor	Mobile Use	SAP Object	Status
1.1	Scheduled Inspection for equipment	System	Overview of assigned Inspection Lots and operations	Preventive Order	REL
1.2	A preventive order will be created	System		Preventive Order	schd
1.3	Schedule and cluster preventive orders	Scheduler		Preventive Order, Employee	assd
1.4	Assign fitter	Supervisor		Inspection Lot	
1.5	A Inspection Lot will be created	System		Inspection Lot Tasks	
1.6	Get overview of operations	Fitter		Inspection Lot Tasks	
1.7	Get next first operation	Fitter		Inspection Lot	
1.8	record results per inspection characteristic	Fitter		Inspection Lot Tasks	
1.9	Registrate finding, usage decision	Fitter		Maintenance Notification	CRTD
1.10	Fitter decision	Fitter			
1.11	Create Notification	Fitter	Create Notification for finding during condition monitoring	Inspection Lot	TAFS
1.12	If last order task then system shows next Inspection Lot else system shows next order	System	Register used Material and performed hours for Preventive	Maintenance Notification	
1.13	Report back and closes Preventive order	Fitter			
1.14	Created notification has to be approved by supervisor. Maintenance repair step 2.3	System			

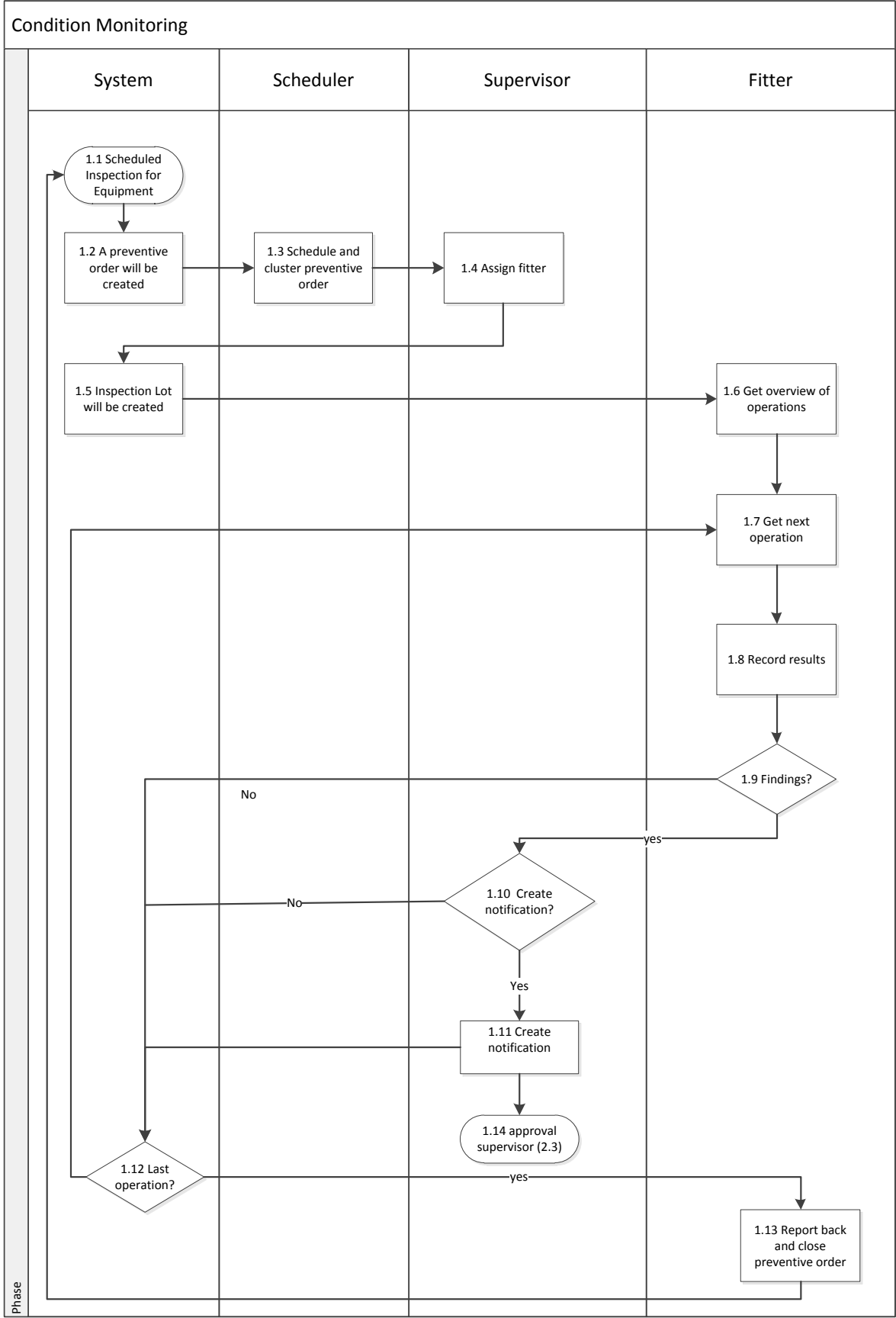
Maintenance repair					
Proces	Proces Step	Actor	Mobile Use	SAP Object	Status
2.1	A defect appear		Create new Notification for defect	Maintenance Notification	OSNO
2.2	Operator creates Notification	Operator		Maintenance Notification	NOPR
2.3	Shift supervisor approves Notification	Shift supervisor		Maintenance Work order	CRTD
2.4	Planner creates Work order	Planner		Maintenance Work order	prpd
2.5	Planner prepares Work order	Planner		Maintenance Work order	REL
2.6	Scheduler plans and releases Work order	Scheduler		Maintenance Work order, Material	
2.7	Planner receives required material	Planner		Maintenance Work order, Employee	assd
2.8	Supervisor assigns fitter	Supervisor		Maintenance Work order Tasks	
2.9	Fitter executes tasks	Fitter		Maintenance Work order Task	CNF
2.10	Fitter reports tasks ready	Fitter		Maintenance Work order Task	PCNF
2.11	Supervisor checks and corrects	Supervisor	Add new task to the work order to perform unforeseen work	Maintenance Work order Task	CNF
2.12	Fitter repairs and reports task ready	Fitter	Registrates used Materials and worked hours for Work order	Maintenance Work order Task	TAFS
2.13	Supervisor checks and approves Work order	Supervisor		Maintenance Notification	appr
2.14	Operator tests and approves Notification	Operator		Maintenance Work order	
2.15	Fitter reports Work order back	Fitter		Maintenance Notification	NOCO
2.16	Shift supervisor closes Notification	Shift supervisor			

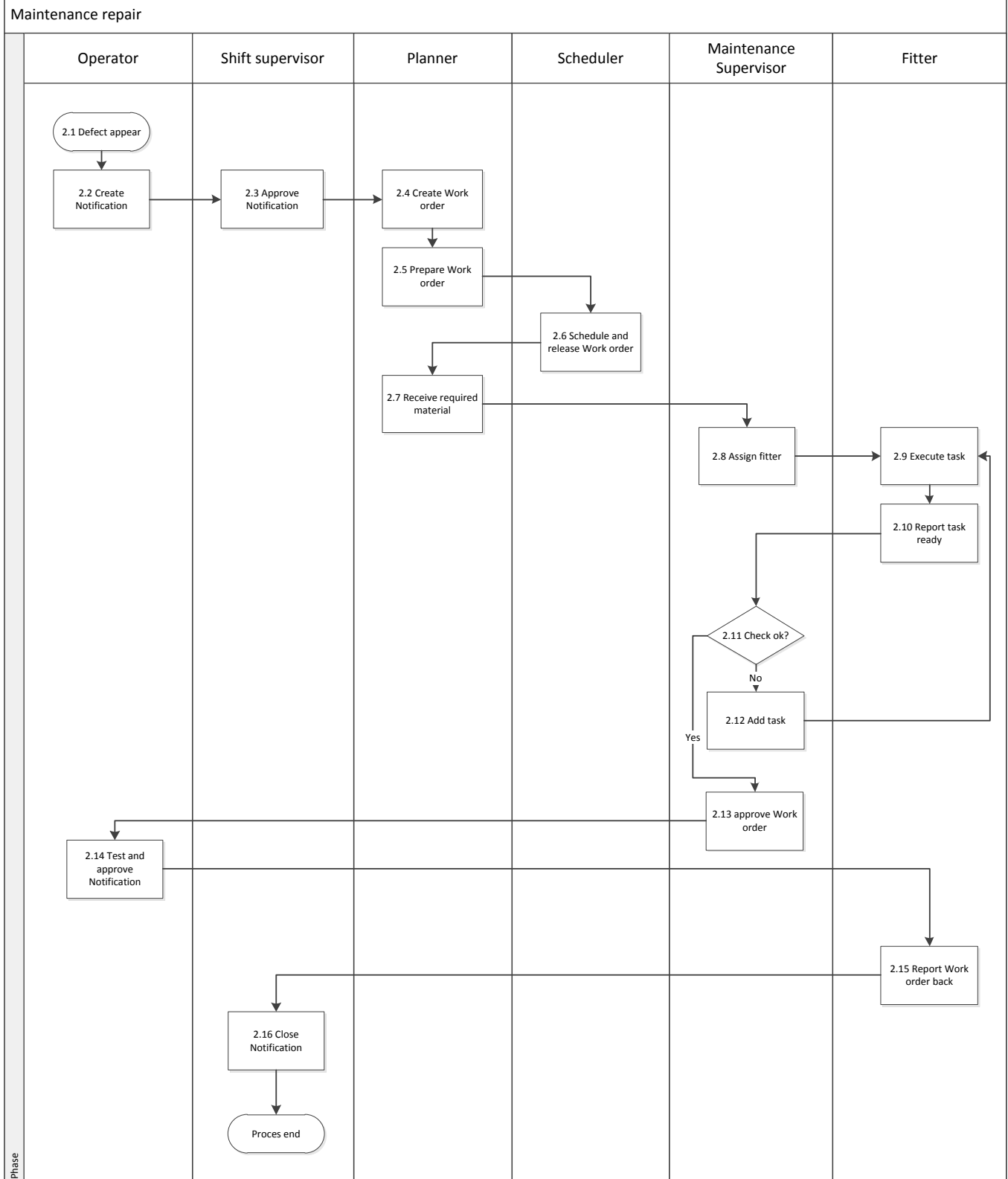
Equipment Calibration					
Proces	Proces Step	Actor	Mobile Use	SAP Object	Status
3.1	Scheduled Inspection for equipment	System	Overview of tasks for Calibration order	calibration order	REL
3.2	A preventive order will be created	System		Calibration order	schd
3.3	Schedule and cluster preventive orders	Scheduler		Calibration order, Employee	assd
3.4	Assign Engineer	Supervisor		Calibration order tasks	
3.5	Get overview of tasks	Engineer		Inspection lot	
3.6	Get first operation	Engineer		Inspection lot	
3.7	record results per inspection characteristic	Engineer		Inspection lot	
3.8	Check if Measured value exceeds threshold	System		Inspection lot	
3.9	Check if Calibration is possible	Engineer		Inspection lot	
3.10	if possible calibrate Equipment	Engineer		Inspection lot	
3.11	If not possible create notifications	Engineer	Insert Used materials and worked hours into Calibration order	Inspection lot	
3.12	Determine overall inspection results	Engineer		Calibration order	TAFS
3.13	If last order task then system shows next order, else system shows next order	System			
3.14	Report back and closes Preventive order	System			
3.15	Created notification has to be approved by supervisor. Maintenance repair step 2.3	Engineer			

## Stappen geschikt voor mobiel

proces stap	Mobile requirements Condition monitoring	SAP Object	Status
1.5	Overview of assigned Inspection lots and tasks	Preventive Order Tasks	CRTD TAFS
1.7	register measured data during monitoring condition of equipment	Preventive Order Tasks	
1.8	Create Notification for finding during condition monitoring	Maintenance Notification	
1.9	Register used Materials and performed hours for Preventive order	Preventive Order	
	Mobile requirements maintenance repair	SAP Object	Status
2.2	Create new Notification for defect	Maintenance Notification	OSNO
2.9	Overview of assigned Work orders and tasks	Maintenance Work order Tasks	CNF PCNF
2.10	Report executed task ready	Maintenance Work order Tasks	
2.11	Add new task to the work order to perform unforeseen work	Maintenance Work order Tasks	
2.15	Registers used Materials and worked hours for Work order	Maintenance Work order	
	Mobile requirements equipment calibration	SAP Object	Status
3.5	Overview of tasks for Calibration order	Calibration order tasks	TAFS
3.6	Register measured data during calibration equipment	Inspection lot	
3.9	Insert Used materials and worked hours into Calibration order	Calibration order	

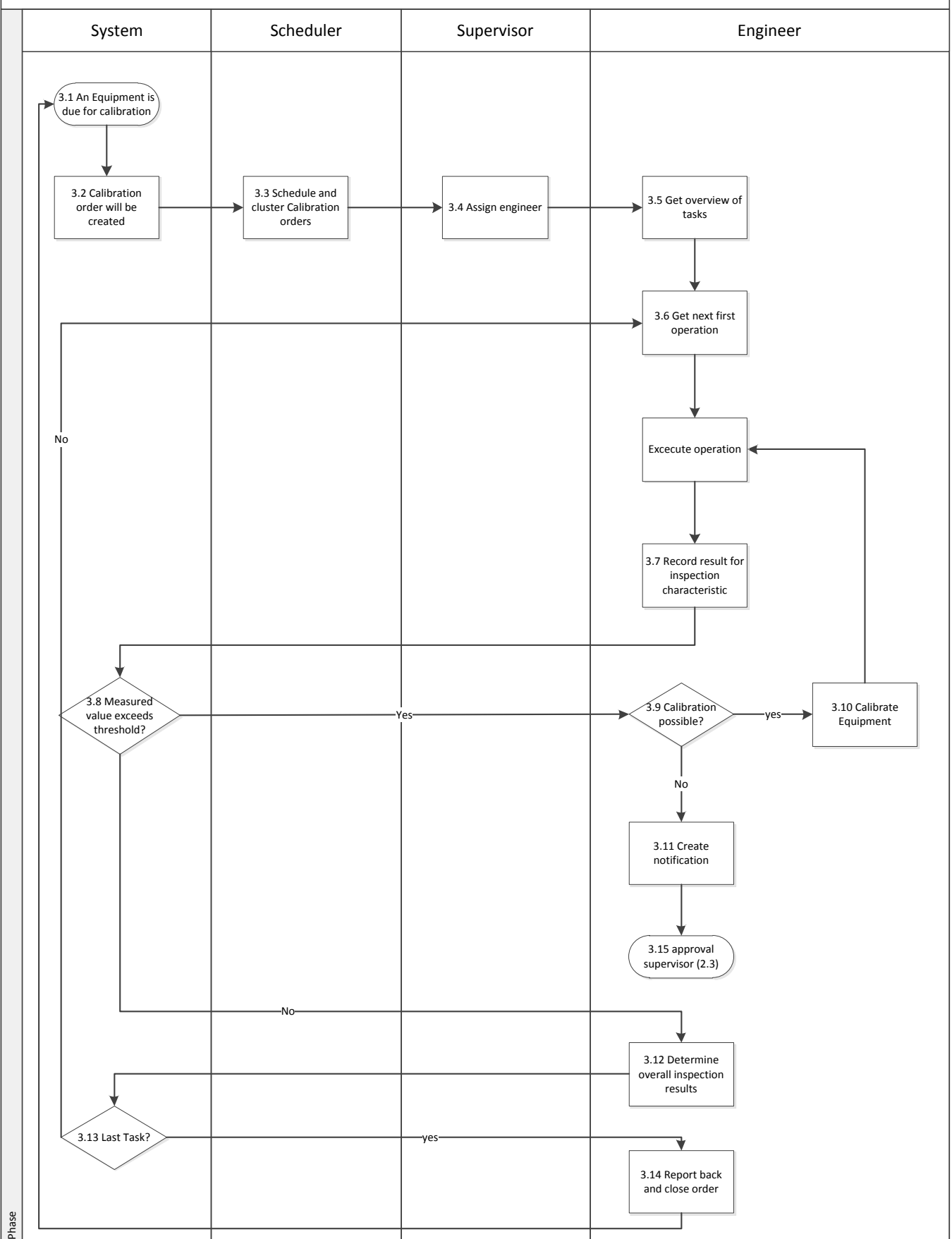
Proces flows







## Equipment Calibration



# Bijlage D: Product Backlog

---

*Mobiel Inspectie Management*

Naam: Jerry van Holst  
Studentnr: 08024146  
Datum: 3-8-2012

## Inhoud

<b>1. Stories .....</b>	<b>26</b>
1) SAP backend systeem inrichten.....	26
2) Sybase Workspace installeren .....	26
3) Sybase Unwired Platform Server Instellen .....	26
4) Verbinding maken tussen Sybase Unwired Platform en Workspace .....	26
5) Verbinding maken tussen workspace en SAP backend systeem.....	26
6) Applicatie Ontwikkel omgeving installeren .....	26
7) Verbinding maken tussen mobiele applicatie en het Sybase Unwired Platform .....	27
8) Inloggen .....	27
9) Ophalen van Inspecties .....	27
10) Ophalen van uit te voeren taken .....	27
11) Resultaten van verrichte metingen invoeren .....	27
12) Resultaten van voorgaande metingen bekijken .....	27
13) Gegevens synchroniseren met Backend.....	27

# Stories

*De Productowner in dit project is Menno Barkmeijer*

## 1) SAP backend systeem inrichten

{techstory} Het SAP backend systeem moet ingericht worden zodat het inspectie proces te doorlopen is. Hierdoor weet je zeker dat alle gegevens die nodig zijn voor de applicatie te vinden is en dat het proces werkt.

**Storypoints: 13**

## 2) Sybase Workspace installeren

{techstory} Er moet een Sybase workspace geïnstalleerd worden waarin de applicatie gedefinieerd kan worden. Hiermee kun je de benodigde gegevens uit het backend ophalen en modeleren in de mobiele applicatie.

**Storypoints: 8**

## 3) Sybase Unwired Platform Server Instellen

{techstory} De Sybase Unwired Platform Server moet ingesteld worden. Dit kan of door zelf een server te installeren of een reeds bestaande server te gebruiken. Deze server zorgt ervoor dat alle applicatie gebruikt kunnen worden door de verbonden apparaten

**Storypoints: 13**

## 4) Verbinding maken tussen Sybase Unwired Platform en Workspace

{techstory} Wanneer er een Server en een workspace is moeten deze twee nog met elkaar kunnen communiceren. Hiermee kun je dan de applicatie uitrollen op de server en kan er gebruik van worden gemaakt door verbonden apparaten.

**Storypoints: 5**

## 5) Verbinding maken tussen workspace en SAP backend systeem

{techstory} Er moet een verbinding worden gelegd tussen de workspace en het SAP backend systeem. Door deze verbinding kunnen de workspace en de backend met elkaar communiceren en kan de workspace gegevens uit het systeem halen.

**Storypoints: 5**

## 6) Applicatie Ontwikkel omgeving installeren

{techstory} De omgeving waarin de mobiele applicatie uiteindelijk ontwikkeld gaat worden moet geïnstalleerd worden. In deze SDK kan later de mobiele applicatie ontwikkeld worden. Deze omgeving staat los van de Workspace

**Storypoints: 5**

## 7) Verbinding maken tussen mobiele applicatie en het Sybase Unwired Platform

{techstory} Er is een verbinding nodig tussen de mobiele applicatie en het Sybase Unwired Platform en de mobiele applicatie zodat de applicatie verbinding kan leggen, gegevens kan opvragen en gegevens kan terugsturen. Hierdoor komt het platform ook te weten wie de gebruiker is.

**Storypoints: 20**

## 8) Inloggen

Als inspecteur wil ik kunnen inloggen op de applicatie met mijn SAP inlog gegevens zodat de applicatie beveiligd is en niet iedereen zomaar mijn gegevens kan zien in de applicatie.

**Storypoints: 20**

## 9) Ophalen van Inspecties

Als inspecteur wil ik alle inspecties in een lijst kunnen zien die ik aan mij toegewezen zijn zodat ik snel en overzichtelijk kan zien welke inspecties voor wanneer gepland staan en niet elke keer terug naar een computer moet voor dit overzicht.

**Storypoints: 13**

## 10) Ophalen van uit te voeren taken

Als inspecteur wil ik bij elke inspectie kunnen zien welke taken of operaties hierbij horen en welke metingen ik moet verrichten zodat ik voor elke inspectie duidelijk zie wat er gedaan moet worden en dit niet elke keer op een computer hoeft op te zoeken.

**Storypoints: 8**

## 11) Resultaten van verrichte metingen invoeren

Als inspecteur wil ik bij elke operatie de gemeten resultaten direct in kunnen voeren. Dit zorgt ervoor dat ik niet eerst alles op een papier moet schrijven en vervolgens moet overtypen op een computer. Dit voorkomt ook fouten.

**Storypoints: 20**

## 12) Resultaten van voorgaande metingen bekijken

Als inspecteur wil ik bij het invoeren van de resultaten zien wat de voorgaande resultaten zijn geweest. Hierdoor kan ik sneller opmerken dat de uitkomst juist wel of niet in lijn ligt met de voorgaande inspecties. En kan ik sneller constateren dat er iets defect is.

**Storypoints: 13**

## 13) Gegevens synchroniseren met Backend

{techstory} Wanneer er gegevens zijn ingevoerd door een inspecteur moeten deze gegevens gesynchroniseerd worden met het backend systeem. Hierdoor zullen de gegevens ook ingevoerd worden in het backend systeem zodat de supervisor het equipment kan goed of afkeuren.

**Storypoints: 20**

# Bijlage E: Ontwerp

---

*Mobiel Inspectie Management*

<p>Naam: Jerry van Holst Studentnr: 08024146 Datum: 20-8-2012</p>
---

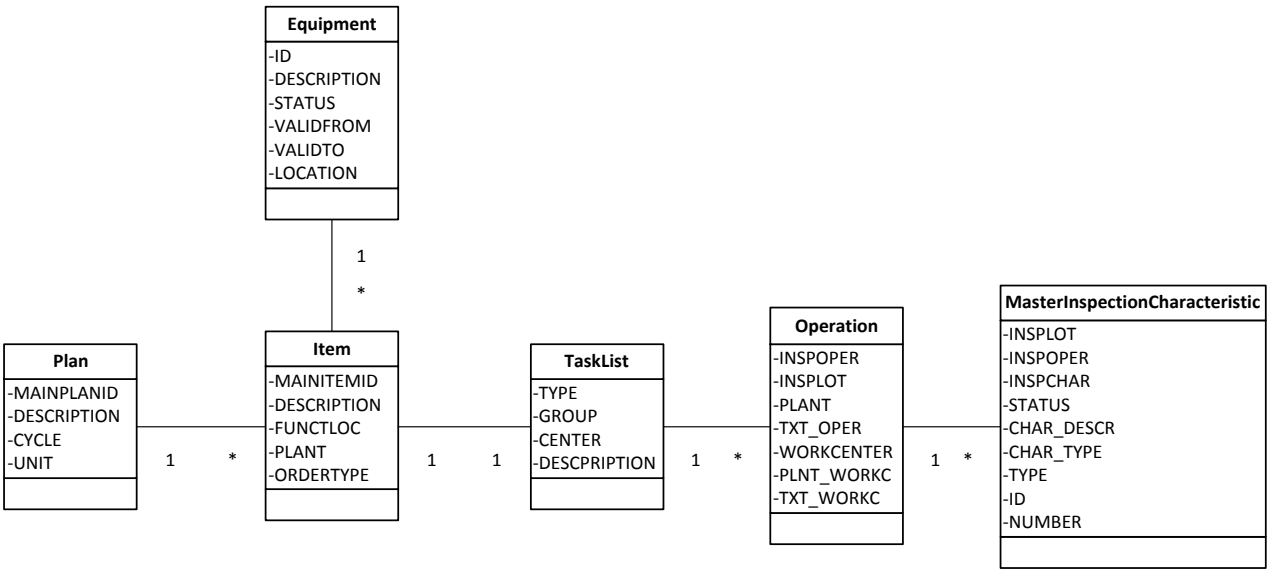
# Inhoud

1.	Datamodellen .....	30
2.	Functioneel ontwerp .....	32

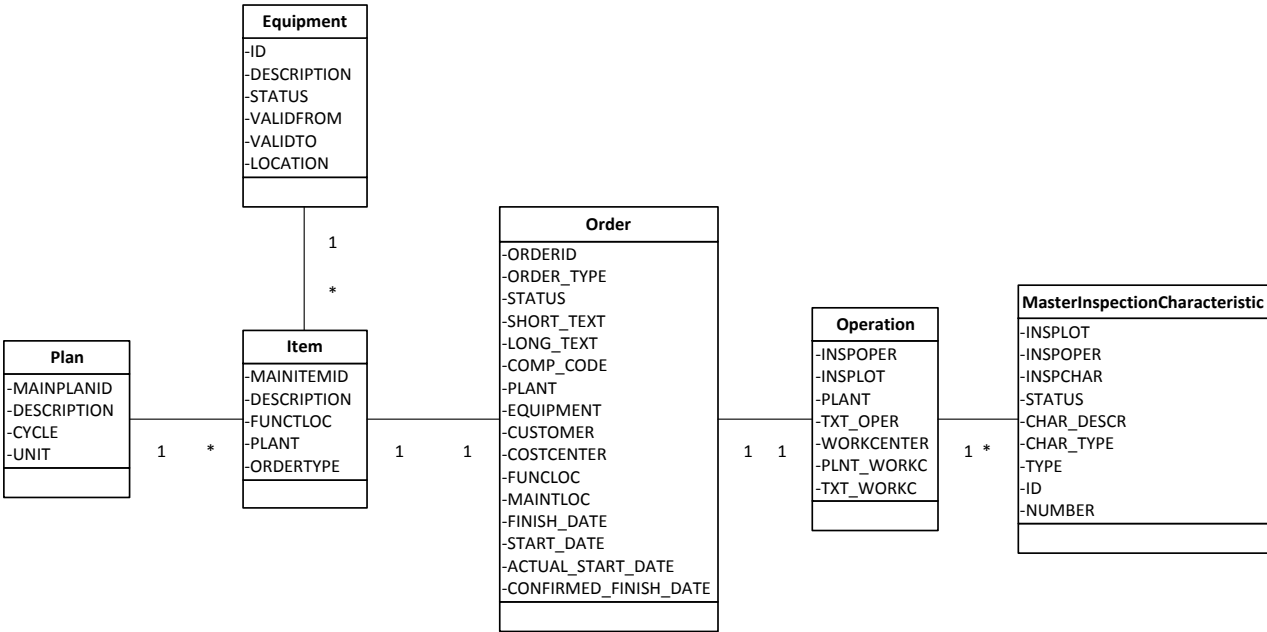
# Datamodellen

In dit hoofdstuk heb ik de datamodellen opgezet van SAP van inspectie management. Het eerste model zijn alle objecten te zien van masterdata. Het tweede gedeelte alle objecten voor een inspectie, en wanneer je de inspectie daadwerkelijk wil uitvoeren heb je een aantal extra objecten nodig.

## Master data

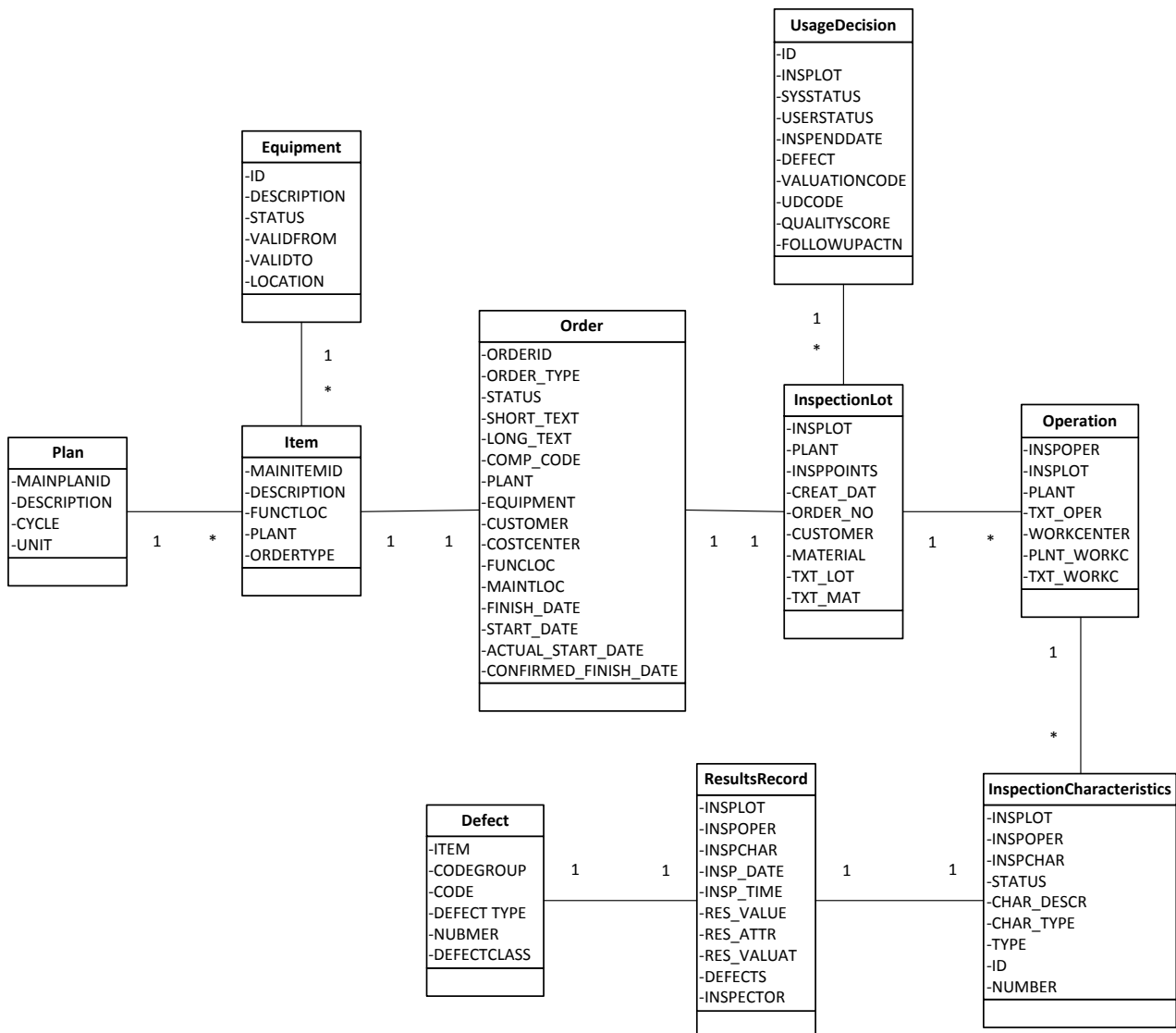


## Inspectie plan





# Uitvoeren van Inspecties



# Functioneel ontwerp

Ik heb ook een functioneel ontwerp gemaakt. Hierin laat ik zien hoe de applicatie er abstract uit komt te zien. Welke gegevens ik wil weergeven en welke informatie er moet worden ingevoerd. Ik heb dit functioneel ontwerp gemaakt in microsoft office Visio met behulp van wireframe. Dit is een speciaal template van Visio voor het visueel weergeven van applicaties. Ik heb het functioneel ontwerp gemaakt op basis van de requirements die besproken zijn met de experts.

## 1. Inlogschermb

In het eerste begin scherm is het inlogschermb te zien. Hierin moet het mogelijk zijn voor de gebruiker om in te loggen met de SAP credentials die hij of zij al heeft in het SAP systeem.

Deze gegevens zijn bekend bij de server en hoef ik dus niet mee te nemen in het datamodel. Bij het genereren van de code in de workspace wordt automatisch al een methode aangemaakt waarbij de applicatie kan controleren of de inloggegevens juist zijn.

Inlog scherm

Username

Enter Text

Password

\*\*\*\*\*

Login

## 2. Inspecties overzicht

In het volgende scherm is een overzicht te zien van alle inspecties met een beschrijving en een uit te voeren datum. Hierin kan de inspecteur alle inspecties bekijken en door te selecteren ook het uitvoeren.

Wat een belangrijk verschil is tussen de SAP applicatie op een computer en een SAP applicatie op een scherm is dat je geen ruimte hebt om heel veel filter mogelijkheden te geven. Wanneer er in SAP een lijstje opgevraagd wordt krijg je eerst 5 verschillende schermen te zien waarin je de filter aan kunt passen. Bij een mobiele applicatie heb je die ruimte simpelweg niet.

Ik heb dit opgelost door in de lijst alleen de belangrijkste gegevens te verwerken. En door middel van een zoekfunctie de mogelijkheid te geven wel in alle achterliggende gegevens te zoeken zoals bijvoorbeeld locatie of klant gegevens .

Inlog scherm

Enter Text

Search

Inspection lots

Inspection Lot: 14000000000084 Description: Plant 1 for Capgemini France Type: 14 Start date: 30.08.2012	>
Inspection Lot: 14000000000085 Description: Plant 2 for Capgemini France Inspection Type: 14 Start date: 10.09.2012	>
Inspection Lot: 14000000000086 Description: Plant 1 for Capgemini Netherlands Inspection Type: 14 Start date: 12.09.2012	>
Inspection Lot: 14000000000087 Description: Plant 2 for Capgemini Netherlands Inspection Type: 14 Start date: 12.09.2012	>
Inspection Lot: 14000000000084 Description: Plant 1 for Capgemini Netherlands Inspection Type: 14 Start date: 12.09.2012	>

### 3. Resultaten invoeren

Wanneer er een inspection lot is geselecteerd krijg je een overzicht te zien welke resultaten er gemeten moeten worden. Daarnaast krijg je een aantal details van de operation te zien.

In dit scherm kom ik eigenlijk weer dezelfde problemen tegen als in alle schermen. Er is in SAP veel te veel informatie te zien voor een mobiele applicatie. Het is dus heel belangrijk om goed te overleggen met de experts welke gegevens er wel zichtbaar moeten zijn in een lijst en welke gegevens niet.

Wanneer tijdens het ontwikkelen blijkt dat dit niet genoeg gegevens zijn voor het uitvoeren van inspecties is eventueel een oplossing om door middel van een longclick functie extra gegevens te laten zien wanneer iemand een lijst item lang ingedrukt houdt.

Inlog scherm

Operations for inspectionlot: 14000000089

Operation: 0010  
Description: Messure oil temp  
Specifications: 50 ... 70 C  
Original Value: 60

Result:

Operation: 0020  
Description: Messure pipe width  
Specifications: 50 ... 60 mm  
Original Value: 55

Result:

Submit

## More Information



### About Capgemini

With more than 115,000 people in 40 countries, Capgemini is one of the world's foremost providers of consulting, technology and outsourcing services. The Group reported 2010 global revenues of EUR 8.7 billion.

Together with its clients, Capgemini creates and delivers business and technology solutions that fit their needs

A deeply multicultural organization, Capgemini has developed its own way of working, the Collaborative Business Experience™, and draws on Rightshore®, its worldwide delivery model.

Learn more about us at:

[www.capgemini.com](http://www.capgemini.com).

*Rightshore® is a trademark belonging to Capgemini.*



**People matter, results count.**

[www.capgemini.com](http://www.capgemini.com)

The information contained in this presentation is proprietary and confidential. ©2011 Capgemini. All rights reserved