

Serviceprogramma	
Titel: UML – Use-cases en templates	
Studentversie	
Afdeling	ICT
Opleiding	Applicatieontwikkelaar
Crebonummer	95311
Niveau + BOL/BBL	4 BOL
Startdatum in curriculum	
Leerjaar	2
Periode	5
Week	1
Niveau serviceprogramma	Verkenning

ALGEMENE INFORMATIE

Soort serviceprogramma	Vaktechnische competentie	
Competenties	H (Overtuigen en beïnvloeden), I (Presenteren), J (Formuleren en rapporteren), K (Vakdeskundigheid toepassen), M (Analyseren), N (Onderzoeken), O (Creëren en innoveren) en R (Op de behoeften en verwachtingen van de “klant” richten)	
Werkprocessen	1.1 (Stelt de vraag en/of informatiebehoefte vast)	
Tijd	160 minuten	
Veronderstelde voorkennis	Geen	
Opbouw serviceprogramma	20 minuten	Opstart (welkom, onderwerp bekend maken, doelen doornemen, VOORDOEN)
	120 minuten	Oefenen, Oefenen, Oefenen (bijsturen waar nodig, rondlopen)
	20 minuten	Afsluiting (evalueren doelen, aantekeningen laten maken en in portfoliomap laten stoppen)
Didactische werkvormen	Instructie Oefening –UML-Use-case diagram en Use-case templates maken Presenteren	
Rol trainer	<ul style="list-style-type: none"> Voordoen (expert) Vragensteller Feedbackgever 	
Doelen	Aan het einde van het serviceprogramma beheerst / weet de student het volgende: VAARDIGHEIDSDOELEN, de student KAN: <ol style="list-style-type: none"> Een Use-case diagram ontwerpen van een aangeleverde context. Een Use-case template maken vanuit het use-case diagram en de context. 	

	KENNISDOELEN, de student WEET: <ol style="list-style-type: none"> 1. De betekenis van de verschillende symbolen die gebruikt worden in use-case diagrammen. 2. De manier waarop deze symbolen gebruikt worden om een use-case diagram logisch te ontwerpen, zodat dit de basis voor het functioneel ontwerp is.
Benodigde ruimte	Ruimte waar met laptops gewerkt kan worden
Leermiddelen en materialen	<ul style="list-style-type: none"> • Laptop • Visio
Bijlagen	bijlagen <ol style="list-style-type: none"> 1. Persoonlijke leerdoelen 2. Serviceprogramma
Literatuur en bronnen	<ul style="list-style-type: none"> - Informatieanalyse – Gerlof Donga en Bert Pinkster - Praktisch UML – Jos Warmer en Anneke Kleppe - Handleiding Visio
Opmerkingen	

Definitie

Unified Modeling Language

De Unified Modeling Language, afgekort UML, is een modelleer taal om objectgeoriënteerde analyses en ontwerpen voor een informatiesysteem te kunnen maken. UML is ontworpen door Grady Booch, James Rumbaugh en Ivar Jacobson in de 1990er jaren en het is sinds 1997 een standaard. Kenmerkend is dat de UML-modellen een grafische weergave zijn van bepaalde aspecten van het informatiesysteem.

UML is een modelleertaal die breed gedragen wordt, wat te zien is aan het feit dat in de jaren negentig van de twintigste eeuw er een UML-consortium werd opgericht met als deelnemers o.a. de volgende bekende organisaties: DEC, Microsoft, Oracle, HP en Texas Instruments. Dit consortium is de Object Management Group (OMG) en heeft de standaard voor UML vastgelegd.

Met UML kunnen niet alleen beschrijvingen worden gemaakt van statische verschijnselen, maar ook van dynamische processen. Het dient vooral als een veelzijdig te gebruiken instrument dat in verschillende fasen van de systeembouw kan worden toegepast. Een van de krachtige aspecten van UML is dat er op relatief eenvoudige wijze meta-beschrijvingen kunnen worden gemaakt.

In tegenstelling tot wat vaak wordt gedacht, is UML zelf geen methode, maar een notatiewijze die bij verschillende methodes kan worden gebruikt. Een methode die gebruik maakt van UML is RUP (Rational Unified Proces).

Serviceprogramma Object Oriented Design en UML

Dit is het eerste Serviceprogramma in de serie Object Oriented Design en UML. De totale Serviceprogramma serie bestaat uit de volgende onderdelen:

1. Object Oriented Design en UML, Use-cases en Use-case template
2. Object Oriented Design en UML, Activiteitendiagrammen
3. Object Oriented Design en UML, Klassendiagrammen en Specificaties voor een database
4. Object Oriented Design en UML, Sequentiediagrammen

Use-case diagrammen

Functionele requirements in use-cases

Met behulp van deskresearch en interview kunnen we de functionele eisen (requirements) van een te bouwen systeem vastleggen. Deze requirements worden verwerkt tot use-cases.

De scope

Allereerst moet de scope (=de grenzen aan het te ontwikkelen systeem) vastgesteld worden:

- a. Wie is er betrokken bij het te ontwikkelen systeem?
- b. Hoe willen deze betrokkenen werken met het systeem?
- c. Wat gaat het kosten?

Activiteiten:

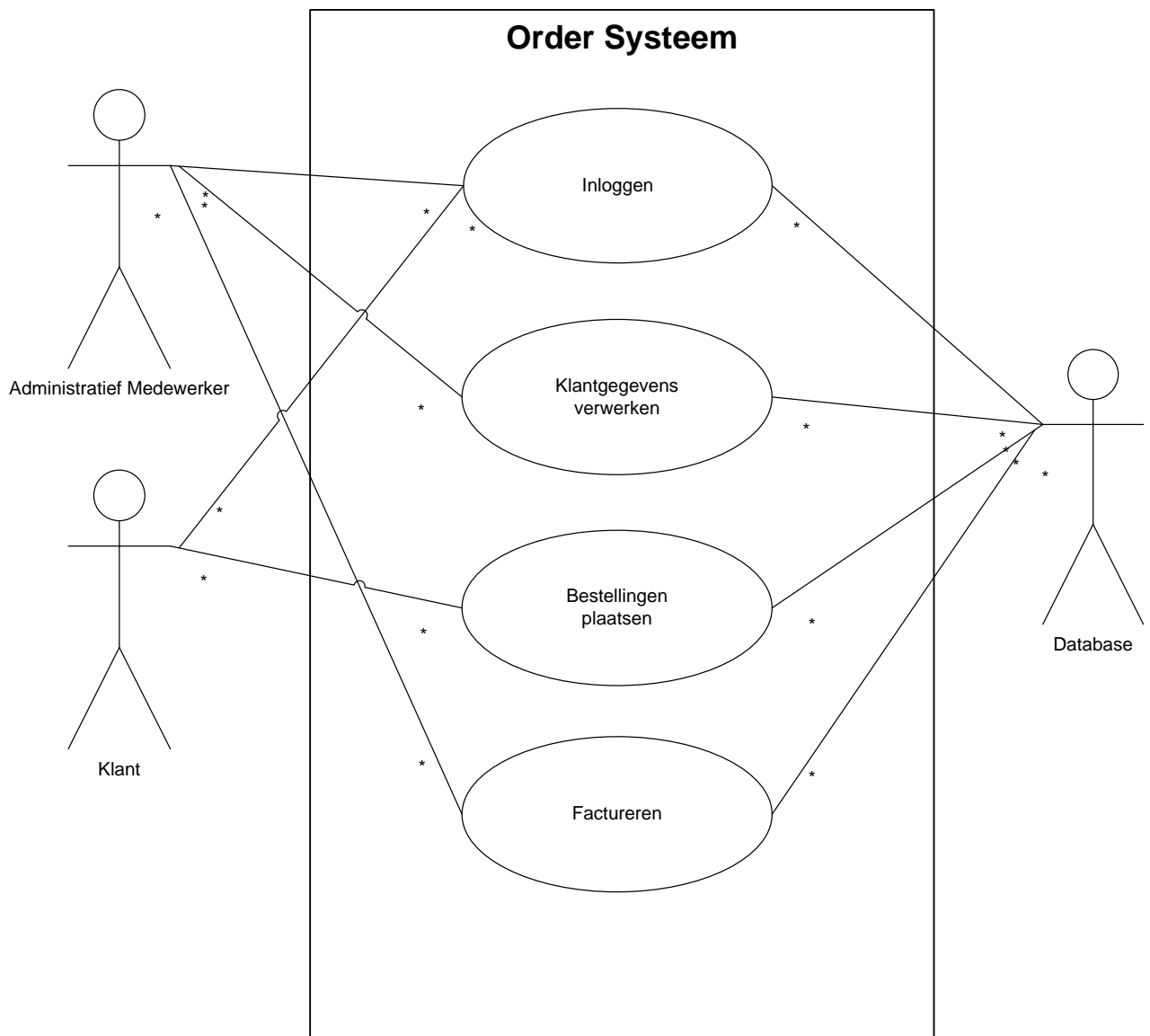
1. deskresearch: bestuderen van beschikbare informatie
2. opstellen van interviews:
 - a. wie wil je interviewen (wie zijn de actoren)?
 - b. welk resultaat wil je bereiken met het interview?
 - c. opstellen van vragen
3. interviews houden
4. interviews uitwerken en laten goedkeuren
5. scope vastleggen m.b.v. een diagram in UML.
- 6.

Interview-vragen

- a. Open vragen
- b. Gesloten vragen

UML: Use-cases

De functionele requirements worden vastgelegd m.b.v. een use-case diagram. Elke functionele eis plaatsen we in of onder een ovaal = **use-case**. Elke gebruiker die betrokken is bij zo'n use-case wordt aangegeven door een poppetje = **actor**, waarbij de functie van de gebruiker wordt vermeld. Actieve actoren (de actoren die het systeem in werking stellen) worden links geplaatst, passieve actoren (ondersteunende actoren) rechts. Een rechthoek om de verzameling use-cases geeft de systeemgrens aan, de **scope**. De lijnen tussen actoren en use-cases geven aan welke actoren bij welke use-cases betrokken zijn.



Niet-functionele requirements

Deze requirements zeggen niets over wat het systeem moet kunnen, maar meer welke beperkingen er zijn (**business constraints**). Bijv. een order kan pas geplaatst worden als deze bij een klant hoort; het toevoegen van een nieuwe klant mag niet meer dan 120 seconden duren.

Eisen als **gebruikersvriendelijk** en **gemakkelijk** moeten in een interview nader gespecificeerd worden.

Pseudo-requirements

Dit zijn requirements die te maken hebben met het systeem of de organisatie, zoals:
“De nieuwe applicatie moet gaan draaien op het bestaande computersysteem”.
“Het geheel moet geprogrammeerd worden in Java”.

User stories

Als er geen bestaand systeem is dan zal het beeld wat de gebruiker heeft van het te ontwikkelen systeem (de user story) omgezet moeten worden in een use-case diagram.

Use Case Template

Bij elke use-case (elk bolletje in je use-case diagram) hoort ook een beschrijving. De beschrijving wordt vastgelegd in een Use Case Template. Deze heeft een vaste indeling:

Naam	
Versie	
Actor	
Preconditie	
Beschrijving	
Uitzonderingen	
Non-functionele requirements	
Postconditie	

Naam	Toevoegen klant
Versie	1.0
Actor	Administratief medewerker, database
Preconditie	Medewerker is ingelogd, systeem is beschikbaar voor invoer
Beschrijving	Medewerker selecteert „Klant toevoegen” Voert naam van klant in Voert adresgegevens in Voert regiocode klant in Voert eventueel ordergegevens klant in Drukt op OK-knop om klant toe te voegen, systeem toont melding “Klant toegevoegd”
Uitzonderingen	Klant al bekend in systeem, melding “Klant komt al voor”, terug naar eerste scherm Klantgegevens incompleet, melding gegevens incompleet, aangeven welke gegevens ontbreken, terug naar invoerscherm
Non-functionele requirements	Invoer klantgegevens moet te realiseren zijn in vijf minuten.
Postconditie	Klant is toegevoegd aan systeem, systeem wacht op actie gebruiker.

Oefening 1

Gegeven is de volgende context:

- Maak een Use-Case Diagram in Visio waarin je de hoofdfuncties verwerkt. Druk deze af om te bewaren in je portfolio.
- Maak van elke hoofdfunctie ook een Use-Case Template. Druk deze af om te bewaren in je portfolio.
- Bij de afsluiting zal je gevraagd worden om je Use-case diagram en Use-case templates te presenteren, waarbij je duidelijk moet aangeven, waarom je het op jouw manier hebt getekend/beschreven.

De Fietsfabriek

Een voormalige fietsenmaker is een fietsenfabriek “De Fietsfabriek” begonnen. De fabriek koopt de onderdelen voor de fietsen in en assembleert zelf de fietsen. De ondernemer vraagt om de automatisering van de nieuwe organisatie op te zetten: Hij wil een applicatie om de inkoop, voorraad en verkoop bij te houden. Ook moeten klanten via Internet producten (fietsen en accessoires) kunnen bestellen en betalen. Klanten kunnen zowel privé-personen zijn, als fietswinkels.

Om kort te gaan wil “De Fietsfabriek” de volgende functies:

- Inkopen van onderdelen bij leveranciers:** De Fietsfabriek heeft verschillende leveranciers waarbij ze producten (fietsonderdelen) inkoopt. Hierbij gaat het om nieuwe voorraad, maar ook voor het aanvullen van bestaande voorraad. Als er besloten wordt om een nieuw product aan te schaffen door de afdeling Marketing, wordt er contact gelegd met de betreffende leverancier en een bestelling gedaan. De leverancier stuurt een bevestiging van de bestelling en de bestelling wordt verzonden samen met de factuur. Zodra de bestelling is aangekomen, wordt deze door de afdeling Voorraadbeheer opgeborgen en de gegevens van de bestelling worden geregistreerd in het systeem. De factuur gaat naar de afdeling Financiën, die deze factuur betaalt. Op vrijdag draait het systeem een lijst uit van alle producten waarvan het aantal onder de vastgestelde limiet komt. Deze worden dan bijbesteld en verwerkt op dezelfde wijze als nieuwe producten.
- Aanmaken klantaccounts:** Een klant kan via internet zijn eigen account aanmaken. Deze heeft hij nodig om fietsen te kunnen kopen. Hierbij worden NAW- en rekeninggegevens geregistreerd. Elke klant heeft een unieke combinatie van inlognaam en wachtwoord.
- Verkopen van producten aan klanten:** Een klant kan via internet producten (fietsen en accessoires) kopen. Nadat de klant heeft ingelogd, kan hij producten selecteren om in zijn winkelwagentje te doen. Zodra hij klaar is met selecteren, kan hij de bestelling afronden. Hierbij kan hij eventueel het afleveradres veranderen. Betalingen kunnen gedaan worden via creditcard, paypal of ideal. Zodra de betaling binnen is, geeft de afdeling Financiën een bericht door aan de afdeling Voorraadbeheer. Deze verzamelt de bestelde producten, verpakt ze samen met een verpakingsbon en laat ze ophalen door het vervoersbedrijf. De voorraden worden direct bijgewerkt in het systeem.
- Marketing:** Deze afdeling houdt zich bezig met het in en uit de handel nemen van producten. Als het om een nieuw product gaat, registreert deze afdeling de basisgegevens (naam, kleur, maat, minimumvoorraad, etcetera) in het systeem. Wekelijks wordt er een verkooplijst uitgedraaid. Aan de hand van deze lijst, beslist de afdeling Marketing of een product behouden of afgestoten wordt. Dit is afhankelijk van de verhouding verkoopvoorraad en de snelheid waarmee de voorraad verkocht wordt. Wordt een product uit de handel genomen, dan wordt de afdeling Voorraadbeheer hiervan op de hoogte gesteld en wordt het product van de website verwijderd.

Oefening 2

Gegeven is de volgende context:

- Maak een Use-Case Diagram in Visio waarin je de hoofdfuncties verwerkt. Druk deze af om te bewaren in je portfolio.
- Maak van elke hoofdfunctie ook een Use-Case Template. Druk deze af om te bewaren in je portfolio.
- Bij de afsluiting zal je gevraagd worden om je Use-case en Use-case template te presenteren, waarbij je duidelijk moet aangeven, waarom je het op jouw manier hebt getekend/beschreven.

Internetbedrijf SMALL

SMALL is een bedrijf wat kleding wil gaan verkopen voor baby's en peuters (leeftijd 0 tot en met 4 jaar) via internet. Hiervoor moet een systeem ontwikkeld worden. Het systeem van SMALL moet de volgende functionaliteit gaan bevatten:

- Inkopen van kleding bij leveranciers:** SMALL heeft verschillende leveranciers waarbij ze kleding inkoopt. Hierbij gaat het om nieuwe voorraad, maar ook voor het aanvullen van bestaande voorraad. Als er besloten wordt om een nieuw product te gaan verkopen door de afdeling Marketing, wordt er contact gelegd met de betreffende leverancier en een bestelling gedaan. De leverancier stuurt een bevestiging van de bestelling en de bestelling wordt verzonden samen met de factuur. Zodra de bestelling is aangekomen, wordt deze door voorraadbeheer opgeborgen en de gegevens van de bestelling worden geregistreerd in het systeem. De factuur gaat naar de afdeling Financiën, die deze factuur betaalt. Op vrijdag draait het systeem een lijst uit van alle producten waarvan het aantal onder de vastgestelde limiet komt. Deze worden dan bijbesteld en verwerkt op dezelfde wijze als nieuwe producten.
- Aanmaken klantenaccounts:** Een klant kan via internet zijn eigen account aanmaken. Deze heeft hij nodig om kleding te kunnen kopen. Hierbij worden NAW-gegevens en rekeninggegevens geregistreerd. Elke klant heeft een unieke combinatie van inlognaam en wachtwoord.
- Verkopen van kleding aan klanten:** Een klant kan via internet kleding kopen. Nadat de klant heeft ingelogd, kan hij producten selecteren om in zijn winkelwagentje te doen. Zodra hij klaar is met selecteren, kan hij de bestelling afronden. Hierbij kan hij eventueel het afleveradres veranderen. Betalingen kunnen gedaan worden via creditcard, paypal of ideal. Zodra de betaling binnen is, geeft de afdeling Financiën een bericht door aan de afdeling Voorraadbeheer. Deze verzamelt de bestelde producten, verpakt ze samen met een verpakkingsbon en laat ze ophalen door het vervoersbedrijf. De voorraden worden tevens bijgewerkt in het systeem.
- Marketing:** Deze afdeling houdt zich bezig met het in en uit de handel nemen van producten. Als het om een nieuw product gaat, registreert deze afdeling de basisgegevens (naam, kleur, maat, minimumvoorraad, etcetera) in het systeem. Dagelijks wordt er een verkooplijst uitgedraaid. Aan de hand van deze lijst, beslist de afdeling marketing of een product behouden of afgestoten wordt. Dit is afhankelijk van de verhouding verkoopvoorraad en de snelheid waarmee de voorraad verkocht wordt. Wordt een product uit de handel genomen, dan wordt de afdeling Voorraadbeheer hiervan op de hoogte gesteld en wordt het product van de website verwijderd.