



Unified Modeling Language

USE CASE DIAGRAMMEN

Inleiding

Bij de aanvaarding van een opdracht om een applicatie te bouwen, is het in de eerste plaats van belang om de eisen (wensen) van de opdrachtgever te inventariseren. In vaktermen wordt dit de “requirements analysis” genoemd. Om na te gaan of de analist de eisen en wensen van de opdrachtgever goed heeft begrepen, kan niet worden volstaan met een uitdraai van programma-code. De opdrachtgever zal hier helemaal niets van begrijpen. En zoals het gezegde “een afbeelding zegt meer dan duizend woorden” al aangeeft, is een diagram het meest aangewezen middel om de opdrachtgever uit te leggen of zijn eisen en wensen goed zijn begrepen. Het diagram dat daartoe uitermate geschikt is, is het use case-diagram.

Use case-diagram

Een use case-diagram is een diagram die aangeeft

- welke functionaliteit de applicatie bevat
- welke rollen bepaalde personen, organisaties of andere (informatie)systemen spelen; het gaat dus om de rollen en niet om de personen, organisaties of systemen
- welke rol wat mag, kan of moet

Opdracht 1

Iedereen maakt regelmatig gebruik van een geldautomaat, van welke bank dan ook. Wat kan iemand bij zo'n automaat doen, met andere woorden welke functionaliteit biedt de geldautomaat?

Opdracht 2

Welke personen en organisaties spelen een rol bij het gebruik van een geldautomaat en welke rol spelen zij?

Opdracht 3

Geef op basis van de antwoorden op de vorige twee opdrachten aan welke rol wat mag, kan of moet.

Onderdelen van een use case-diagram

Een use case-diagram kan de volgende onderdelen bevatten:

- één of meer actoren
- één of meer use cases
- associaties tussen actoren en use cases
- één of meer include-relaties tussen use cases
- één of meer extend-relaties tussen use cases
- een systeemgrens
- één of meer generalisaties tussen actoren

Actor

Een actor is de rol van een natuurlijke persoon, een organisatie of een ander (informatie)systeem, die normaal gesproken communiceert met het systeem. De actor staat dan buiten het systeem, buiten de grenzen van het systeem. In uitzonderlijke gevallen kan een actor deel uitmaken van het systeem.

Een actor kan in het diagram op twee manieren worden aangegeven:



- een “stokmannetje”; de naam van de actor wordt onder het stokmannetje geplaatst
- een rechthoek, prototyping de actor als «actor»; het prototype wordt boven de naam van de actor geplaatst; de naam van de actor wordt in de rechthoek onder het prototype geplaatst

Beide elementen kunnen door elkaar worden gebruikt, maar het geniet de voorkeur om het stokmannetje te gebruiken voor natuurlijke personen en de rechthoek voor organisaties en systemen. Dit laatste is niet mogelijk in Visual Paradigm. In deze CASE-tool moet ook gebruik worden gemaakt van het “stokmannetje” voor systemen of organisaties, die als actor fungeren in het diagram.

Use Case

Een use case wordt weergegeven door middel van een ovaal, waarin de naam wordt geplaatst.

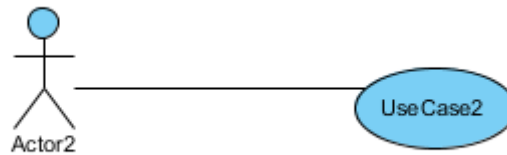


Een use case geeft een bepaalde functionaliteit van het systeem weer, een functie die de applicatie moet kunnen verwezenlijken. Het is niet de bedoeling dat dit in eerste instantie te gedetailleerd gebeurt.

Use cases die geassocieerd zijn met een actor worden primaire use cases genoemd. Use cases die gerelateerd zijn aan andere use case door middel van een include- of extend-relatie (zie hierna) worden secundaire use cases genoemd.

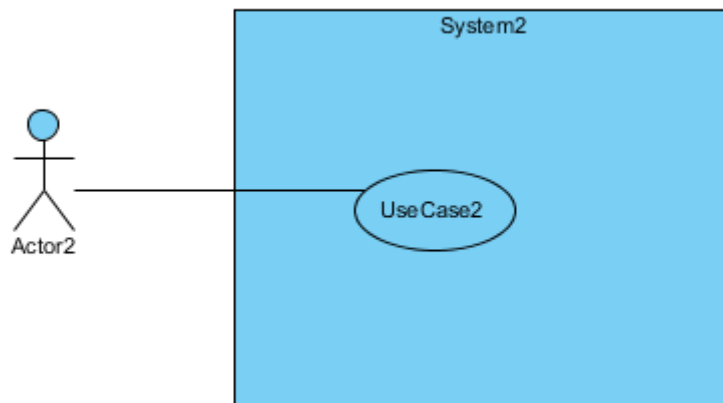
Associatie

Een associatie is een relatie tussen een actor en een use case, en wordt weergegeven door een ononderbroken lijn. Beter nog, een associatie komt alleen voor tussen een actor en een use case. Het geeft aan wat een bepaalde actor mag, kan of moet doen.



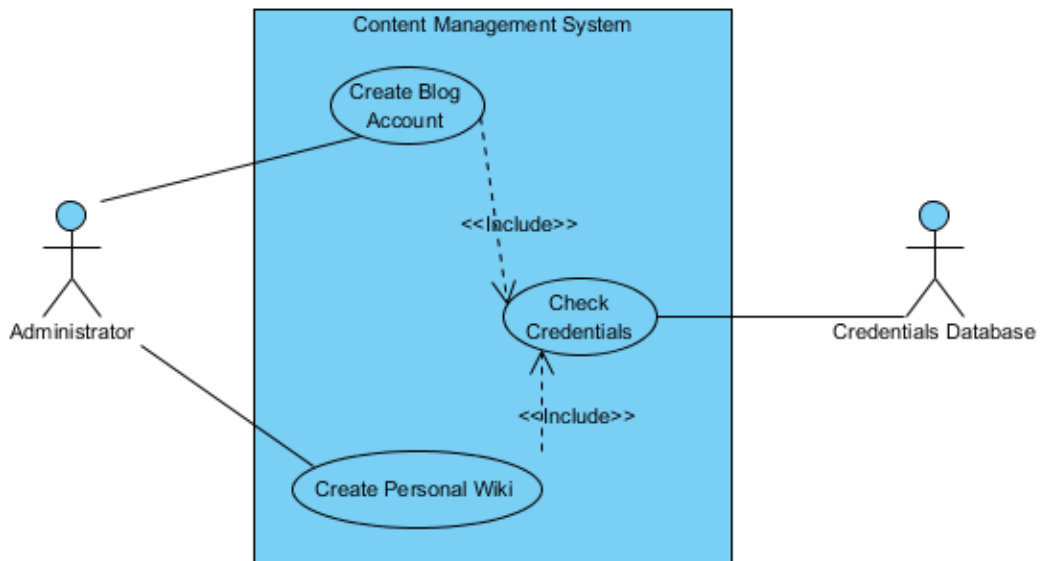
Systeemgrens

De systeemgrens geeft aan wat zich binnen en buiten het systeem (of de applicatie) bevindt.



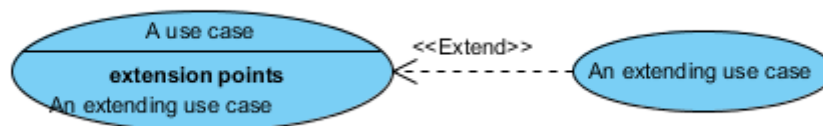
De actoren bevinden zich (over het algemeen) buiten de systeemgrens en zijn dus geen onderdeel van het systeem. Use cases bevinden zich binnen de systeemgrens en zijn dus onderdeel van het systeem.

Include-relatie



Een include-relatie houdt in dat de ingevoegde use case altijd moet worden uitgevoerd tijdens de uitvoering van de use case die de andere use case invoegt. De pijl van de include-relatie wijst altijd naar de use case die wordt ingevoegd.

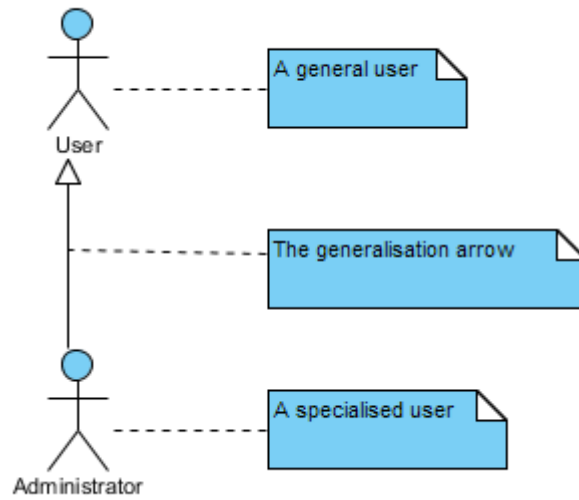
Extend-relatie



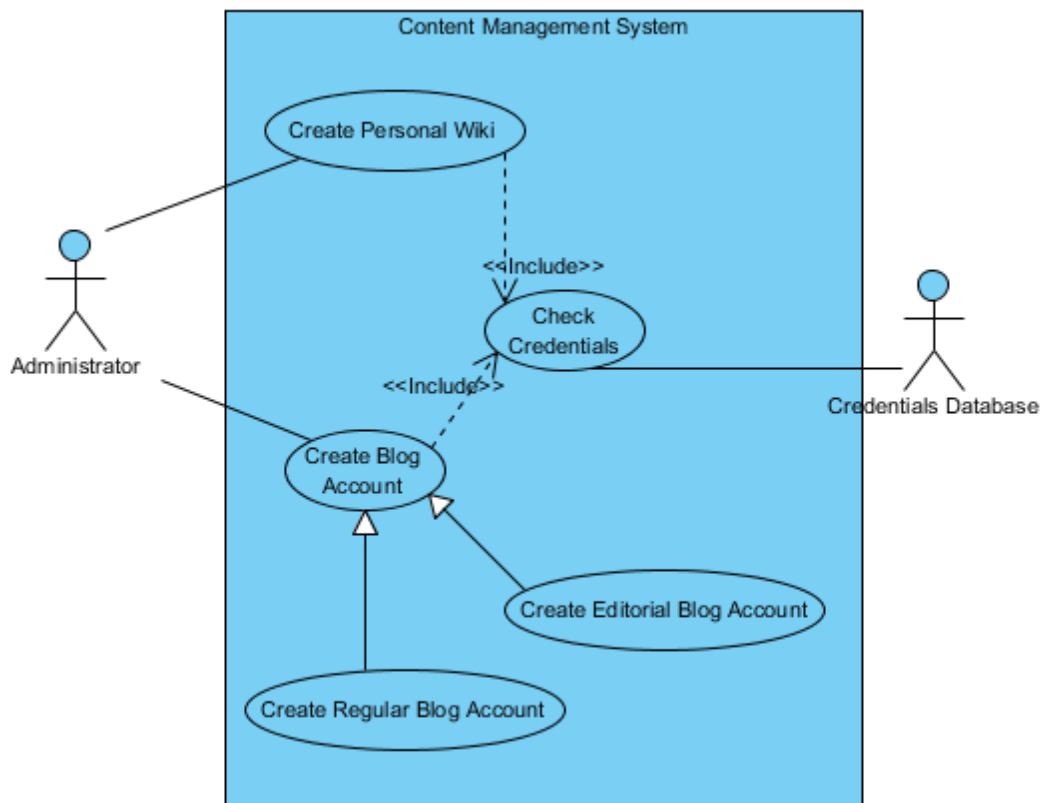
Een extend-relatie houdt in dat de use case met uitgebreide functionaliteit alleen wordt uitgevoerd, wanneer in de aanroepende use case aan bepaalde voorwaarden is voldaan. De pijl van de extend-relatie wijst altijd naar de aanroepende use case. Een aantal UML applicaties plaatst in de aanroepende use case een zogenaamd “extension point”, een horizontale lijn in de aanroepende use case en daaronder de gegevens van de “extending use case”.

Generalisatie

Generalisatie komt voor bij zowel actoren als use cases. Het betekent dat er sprake kan zijn van een algemene vorm en (steeds meer) gespecialiseerde vormen van actoren en use cases. In het voorbeeld van een blog kan sprake zijn van niet-leden, leden, moderators en beheerder, misschien zelfs een super-beheerder. Elk van deze actoren is een generalisatie van de navolgende.



De pijl wijst altijd naar de generaliserende actor of use case.



Opdracht 4

Maak een use case-diagram op basis van de uitkomsten van de opdrachten 1 tot en met 3.

Use case specificaties

Een use case diagram wordt vaak vergezeld van use case specificaties worden gemaakt. Voor elke use case wordt een dergelijk specificatie opgemaakt. De use case specificatie bestaat uit:

- naam van de use case
- een eventuele ID
- het belang van de use case (prioriteit)
- de primaire actor
- de belanghebbenden en hun belang
- een korte beschrijving
- door wie en of wat wordt de use case geactiveerd (getriggerd)
- welke relaties heeft de use case (associatie, include, extend en generalisatie)
- wat is de normale volgorde van de activiteiten binnen de use case (wanneer de use case succesvol wordt doorlopen)
- wat zijn eventueel de alternatieven in de volgorde van de activiteiten binnen de use case (de use case zal niet altijd met succes worden doorlopen)
- is er sprake van pre-condities (moet voor de uitvoering van de use case zijn voldaan aan bepaalde voorwaarden)
- is er sprake van post-condities (aan welke voorwaarden moet zijn voldaan nadat de use case is uitgevoerd)

Een aantal van de genoemde onderdelen zal duidelijk kunnen zijn aan de hand van het voorgaande in dit hoofdstuk. Andere onderdelen verdienen enige verduidelijking.

Normale volgorde van activiteiten

De normale volgorde van activiteiten binnen een use case, is de volgorde waarbij er geen enkele abnormaliteit optreedt en de use case met succes wordt doorlopen. In het geval van de geldautomaat betekent dit bijvoorbeeld voor de use case “Geld opnemen” dat alle activiteit om het gewenste bedrag uit de geldautomaat te ontvangen zonder enig probleem verloopt: de bankpas wordt herkend, de pincode is juist, er is voldoende saldo, er is voldoende geld in de automaat aanwezig, de rekeninghouder wacht niet te lang met het uitnemen van de pas (de pas wordt niet ingeslikt), het geld komt uit de automaat, de rekeninghouder wacht niet te lang met het uitnemen van het geld.

Deze normale volgorde van activiteiten wordt in de juiste volgorde geplaatst en genummerd met 1, 2, 3, enzovoorts.

Alternatieve volgorde van activiteiten

De alternatieve volgorde van activiteiten betreffen de abnormaliteiten die kunnen optreden in de normale volgorde. Stel dat de normale volgorde voor het opnemen van geld met een bankpas als volgt is:

1. De geldautomaat toont het startscherm
2. Rekeninghouder plaatst de bankpas in de sleuf van de geldautomaat
3. De bankpas wordt geïdentificeerd
4. De geldautomaat toont het hoofdmenu
5. Rekeninghouder kiest voor geld opnemen
6. De geldautomaat toont het menu ‘geld opnemen’
7. Rekeninghouder kiest standaard geldbedrag zonder transactiebon

8. Geldautomaat toont melding 'pas uitnemen'
9. Geldautomaat verwijdert de pas uit de sleuf
10. Geldautomaat toont melding 'geld uitnemen'
11. Geldautomaat geeft het gewenste bedrag in biljetten uit

Bij elke van deze afzonderlijke activiteiten zou er iets mis kunnen gaan. Neem als voorbeeld activiteit 3, de identificatie van de bankpas. Wat kan hier misgaan? Bijvoorbeeld:

- de bankpas wordt niet herkend, omdat deze beschadigd is
- de bankpas is geblokkeerd in verband met verlies, diefstal of skimming

Op deze problemen moet actie worden ondernomen. Dit worden de alternatieve activiteiten, die als alternatieven worden opgenomen met een subnummering onder 3:

- 3a1. Bankpas niet herkend
- 3a2. Bankpas uitnemen
- 3a3. Startscherm tonen

- 3b1. Bankpas geblokkeerd
- 3b2. Bankpas wordt ingeslikt
- 3b3. Speciale melding tonen
- 3b4. Startscherm tonen

Zoals hier is te zien, wordt voor elk alternatieve een aparte subnummering gebruikt, in dit geval a, b, c, enzovoorts. Voor elke activiteit binnen een alternatief wordt een nieuwe subnummering gebruikt, hier 1, 2, 3, enzovoorts. Maar, in plaats van 3a1, 3a2, kan ook 3.1.1, 3.1.2 enzovoorts worden gebruikt.

De vorm is niet belangrijk, de volgorde wel. Ook van belang is het herkennen van de normale en de alternatieve volgorde van de activiteiten binnen een use case. Er kan echter ook nog sprake zijn van subactiviteiten. Wanneer de normale volgorde van activiteiten te complex of te gedetailleerd wordt, kan worden gekozen om normale activiteiten als subactiviteiten aan te merken. Zo kan in plaats van met een bankpas ook met een creditcard geld worden opgenomen. De activiteiten verlopen dan net iets anders dan bij een bankpas, zoals het kiezen van een taal voor weergave van de schermen die door de geldautomaat worden getoond.

Pre- en postcondities

Het is ook heel belangrijk om de voorwaarden te kennen en te herkennen die van belang zijn voor en na de uitvoering van een use case. Voor de use case 'geld opnemen' zal dat duidelijk zijn:

- voorwaarde vooraf: een bankrekening, een (niet geblokkeerde) bankpas en voldoende saldo hebben
- voorwaarde achteraf: het gewenste geldbedrag ontvangen, de pas terugkrijgen en eventueel de gevraagde transactiebon

Opdracht 5

Schrijf een use case specificatie voor één van de use cases uit het diagram van opdracht 3. Kies je voor de use case "Geld opnemen", doe dit dan voor de variant met de creditcard.