## Reference Card

## Use Cases

**Zoals bij natuurlijke taal komen er ook in de UML veel dialecten voor. In zo’n dialect zijn er net iets andere afspraken, aangepaste layout of andere naamgeving van toepassing. De UML artifacten van bedrijven, organisaties e.d. zijn dan ook vaak verschillend maar met enige moeite voor ons wel te begrijpen.**

**Omwille van de leesbaarheid en de eenduidigheid spreken we daarom een aantal zaken af in deze Reference Card.**

**Use Cases:**

* De naam van een use case bevat een werkwoord. En is uniek en duidelijk. De naam is betekenisvol voor de Actor.
* Formuleer alles altijd actief en niet passief, zodat duidelijk is wie welke handelingen doet. Dus niet "Het *wordt* ingevuld", maar "De *actor* vult dit in: ..".
* We nummeren nooit dieper dan drie niveaus in een stappenplan. Als dit nodig blijkt, klopt de complexiteit van de use case niet en moet deze gesplitst worden.
* Meerdere pre- of postcondities scheiden we altijd met het woord *'of'* of *'en'*.
* De pre/postcondities zijn relevant voor het systeem. Dus geen dingen als "Er moet een printer aanwezig zijn".
* Het einde van een use case geef je aan met het woord "Stop". Dit kan na een (niet geslaagde) voorwaarde zijn of aan het einde van de use case, maar dit laatste is triviaal dus mag weggelaten worden.
* In het Use Case Diagram spreken we van een of meerdere Actors en niet van Acteurs.
* Voor het aanroepen van een secundaire use case gebruiken we:   
  "Voer uit **Invoeren NAW**", waarbij 'Invoeren NAW' een secundaire use case is.

**Fully dressed Use Cases (ook wel: Use Case Outline):**

* Het stappenplan in een use case beschrijving is altijd genummerd.
* We onderscheiden apart de Basic Flow en de Alternative Flow
* Ook de Alternative flows, uitzonderingen en faalscenario’s beschrijven we (indien mogelijk) in 2 kolommen. Maak duidelijk waar deze flows “inhaken” op de Basic Flow.
* We geven links in een tabel de acties van de gebruiker en rechts de acties van het systeem.
* De Basic Flow start altijd met een actie van de gebruiker en eindigt met een actie van het systeem.
* Zijn actief geschreven, zodat duidelijk is wie welke handelingen doet. Dus niet "Het *wordt* ingevoerd", maar "De *actor* voert “bla” in: ..". Vermijdt dus de werkwoorden worden en zijn.
* Trigger moet beschreven zijn in de eerste stap
* Het is altijd duidelijk welke Actor de stap uitvoert.
* Alle mogelijke postcondities zijn toegevoegd.

Template Fully Dressed UseCase

(*Het grijs gearceerde gedeelte is de een HighLevel UseCase beschrijving / Brief Use Case)*

|  |  |  |
| --- | --- | --- |
| Naam | Naam van de UseCase | |
| Actor(s) | Actor(s) die geassocieerd zijn aan deze UseCase. | |
| Purpose | Doel van de UseCase | |
| Requirements | Requirements waaraan de UseCase voldoet cq. invulling geeft.  Leg hier een koppeling naar de geïnventariseerde requirements | |
| Scenarios | Scenario’s die de UseCase ondersteunt. | |
| WireFrame | De gekoppelde wireframe’s | |
| Preconditie(s) | Conditie vooraf waaraan voldaan is/moet zijn | |
| Postconditie(s) | Conditie waarin het systeem verkeert na voltooiing van de UseCase | |
| **Basic Flow (Main Success Scenario)** | | |
| *Actor action* | | *Systems response* |
|  | |  |
|  | |  |
| **Exceptions** | | |
|  | |  |
|  | |  |
|  | |  |
|  | |  |