***Association antara actor dan use case***

Ujung panah pada association antara actor dan use case mengindikasikan ***siapa/apa*** yang meminta interaksi dan bukannya mengindikasikan aliran data

Sebaiknya gunakan garis tanpa panah untuk association antara actor dan use case



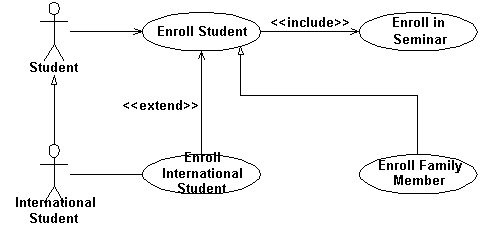
association antara actor dan use case yang menggunakan panah terbuka untuk mengindikasikan bila actor berinteraksi secara ***pasif*** dengan system anda.



***Generalization/inheritance***

* Generalization/inheritance digambarkan dengan sebuah garis berpanah tertutup pada salah satu ujungnya yang menunjukkan lebih umum
* Harus digambarkan secara vertical

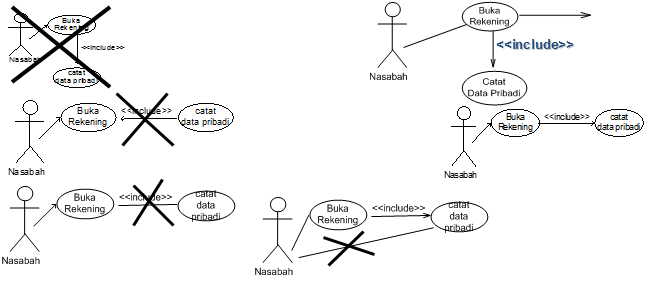
Contoh :



<<i**nclude**>> , yaitu kelakuan yang harus terpenuhi agar sebuahevent dapat terjadi, dimana pada kondisi ini sebuah use case adalah bagian dari use case lainnya.

*Association antara use case*

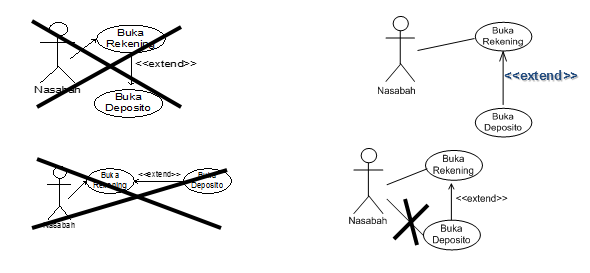
* <<include>>
  + termasuk didalam use case lain (required) / (diharuskan)
  + Pemanggilan use case oleh use case lain
  + contohnya adalah Pemanggilan sebuah fungsi program
  + *Gambarkan association <<include>> secara horizontal*
  + Tanda panah terbuka harus terarah ke sub use case
  + Tidak boleh actor dihubungkan pada use case <<include>>



<<**extends**>>, kelakuan yang hanya berjalan di bawah kondisi tertentu seperti menggerakkan alarm.

*Association antara use case*

* <<extend>>
  + Perluasan dari use case lain jika kondisi atau syarat terpenuhi
  + Kurangi penggunaan association Extend ini, terlalu banyak pemakaian association ini membuat diagram sulit dipahami.
  + Tanda panah terbuka harus terarah ke parent/base use case
  + *Gambarkan association extend secara vertical (picture extending use case below than base/parent use case)*
  + Tidak boleh actor dihubungkan pada use case <<extend>>



<<**communicates**>>, mungkin ditambahkan untuk asosiasi yang menunjukkan asosiasinya adalah communicates association . Ini merupakan pilihan selama asosiasi hanya tipe relationship yang dibolehkan antara actor dan use case.

**Dependency**

Merupakan relasi yang menggambarkan ketergantungan antara usecase yang satu dengan usecase yang lain. Ada dua macam dependency yaitu include dan extends. Include menggambarkan bahwa jalannya suatu usecase memicu jalannya usecase lain. Misalnya usecase login diinclude oleh usecase memilih menu, artinya usecase memilih menu akan memicu dijalankannya usecase login. Sebelum aktor menjalankan usecase memilih menu, aktor harus menjalankan usecase login dulu. Dalam penggambaran diagram usecase, parah mengarah kepada usecase yang diinclude. Sedangkan extends menggambarkan bahwa suatu usecase dijalankan karena ada persyaratan tertentu dari usecase lain. Misal, dalam sebuah sistem user tidak bisa menjalankan login sebelum dia mendaftar akun. Dalam diagram usecase, usecase daftar akun mengextends usecase login. Artinya aktor harus menjalankan usecase daftar akun dulu sebelum menjalankan usecase login karena usecase login memiliki syarat aktor yang melakukan login harus sudah melakukan pendaftaran akun. Arah panah dependency mengarah pada usecase yang memiliki syarat. (MSDN, n.d). Berikut adalah contoh Usecase Diagram : 

