**Diagrama cazurilor de utilizare**

Orice program software trebuie sa interactioneze cu exteriorul fie cu alti oameni, fie cu alte sisteme.

Diagrama cazurilor de utilizare (use case diagram) este o diagrama UML, utilizata in faza de proiectare si analiza a unui sistem software pentru a defini functionalitatile sistemului si pentru a clarifica cerintele si nevoile impuse atunci cand acestea trebuiesc transmise altor persoane intr-un mod usor de inteles, independent de tehnologia utilizata in implementarea aplicatiei.

Programatorii, pot urmarii aceste diagrame pe parcursul intregului proces de dezvoltare pentru a se asigura ca sistemul este construit conform specificatiilor si cerintelor initiale.

O diagrama use case are cateva componente specifice: actori, use case-uri si relatiile dintre ele.

**Actorii**

Acestia reprezinta niste entitati externe care pot avea roluri diferite si care interactioneaza cu sistemul descris. Actorii pot fii niste simplii utilizatori, sisteme software sau dispozitive hardware si se reprezinta grafic printr-o persoana.

Shape

Description automatically generated with low confidence

Figura 2.1. Reprezentarea grafica a unui actor

.

**Use case-uri**

Acestea sunt actiunile pe care sistemul le ofera utilizatorilor. Un use case descrie o functionalitate a sistemului, relevanta pentru actor insa nu se specifica alte detalii precum modul in care se face aceasta actiune si nici modul in care aceasta a fost implementata.

Grafic, un use case se descrie printr-o elipsa in interiorul careia apare numele acestuia.

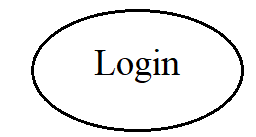


Figura 2.2. Reprezentarea grafica a unui use-case care descrie actiunea de logare

Numele cazului de utilizare trebuie sa fie unic, simplu si sa descrie o actiune sau o secventa de actiuni usor de inteles pentru utilizatori.

**Relatii**

Intre use case-uri si actori se stabilesc relatii care sunt importante pentru descrierea modului in care sitemul functioneaza si cum interactioneaza actorii cu acesta.

Exista cateva tipuri de relatii: asociere, dependenta si generalizare.

**Relatia de asociere** se poate defini intre use case-uri sau intre actori si use case-uri. Aceasta sugereaza comunicarea intre componentele pe care le uneste si se poate reprezenta grafic printr-o linie.

Shape, rectangle

Description automatically generatedFigura 2.3. Reprezentarea grafica a unei relatii de asociere intre doua use case-uri

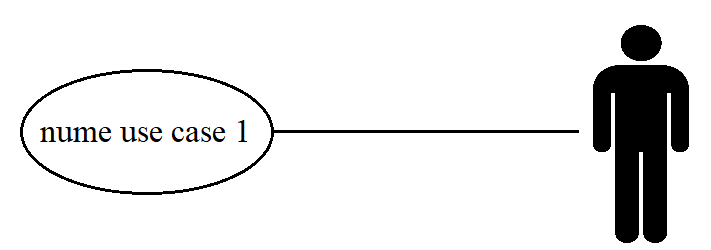


Figura 2.4. Reprezentarea grafica a unei relatii de asociere intre un use case si un actor

**Relatia de dependenta** se poate stabili intre doua cazuri de utilizare si este de doua tipuri: include si extend.

Relatia de tip **include** este folosita atunci cand un caz de utilizare depinde de alt caz de utilizare pentru a functiona.

In urmatorul exemplu, use case-ul B este inclus de catre use case-ul A. Cu alte cuvinte, atunci cand se executa cazul A si cazul B se executa inclusiv.

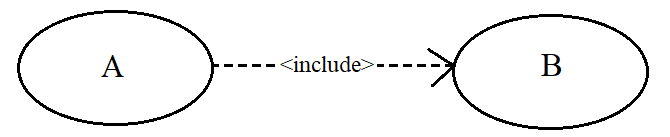


Figura 2.5. Depeneta de tip **include**

Relatia de tip **extend** se foloseste atunci cand cele doua cazuri de utilizare sunt de sine statatoare adica nu depind unul de calalalt. In exemplul urmator, executia use case-ului A determina daca B se executa sau nu.

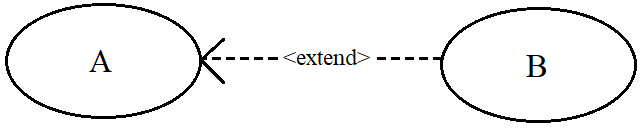


Figura 2.6. Dependenta de tip **extend**

**Relatia de generalizare** este folosita pentru a arata ca un caz de utilizare este o varianta sau specializare a altui caz de utilizare iar spre deosebire de celelalte tipuri de relatii, poate avea loc numai intre doi actori sau intre doua cazuri de utilizare. Un actor va mosteni rolul altui actor, impreuna cu cazurile de utilizare atribuite acestuia.