

Extragerea cerințelor (Requirements elicitation)

Prof. univ. dr. ing. Florica Moldoveanu

Curs Ingineria programelor – UPB, Automatică și Calculatoare
2020-2021

Extragerea cerintelor

Scopul: identificarea si definirea cerințelor utilizator

- Se poate efectua în fazele de început ale procesului de dezvoltare sau pe întreg parcursul procesului de dezvoltare - în funcție de modelul de dezvoltare folosit.
- Metoda folosită pentru extragerea cerintelor este adaptată la nivelul fiecărei organizații și în funcție de caracteristicile fiecărui proiect.
- Rezultatul, “Cerintele utilizator” (User Requirements) este sintetizat în “**Documentul de specificare a cerintelor**”.
- **O cerință este:**
 - o functionalitate pe care viitorul produs trebuie sa o ofere – cerinta functionala
 - o constrângere pe care produsul trebuie să o satisfacă pentru a fi acceptat de client – cerinta nefunctionala

Participanți la extragerea cerintelor (*stakeholders*)

- Analistul (Requirements engineer) – coordoneaza extragerea si definirea cerintelor
- Utilizatorii finali, beneficiarii viitorului produs
- Clientul / posibili cumparatori
- Analisti de marketing (pentru produse destinate pietei) – studiul pietei
- Reprezentanti ai autoritatilor din domeniul de operare al produsului software (probleme legislative, reguli in cadrul organizatiei)
- Ingineri software
 - analizeaza fezabilitatea cerintelor exprimate de ceilalti participanti, cerintele tehnice, costurile.
 - definesc scenarii și cazuri de utilizare ale viitorului sistem informatic

Analistul negociaza între cerintele diferitelor participanti, cerintele tehnice, bugetare, legislative, etc.

Sursele cerințelor

- **Studiul de piață**: exista produse similare? Ce va aduce nou viitorul produs?
- **Domeniul in care va opera produsul**: de ex. bancar, clinic, controlul in timp real al proceselor, etc; anumite cerinte sunt specifice domeniului.
- **Participantii la procesul de extragere a cerintelor** – fiecare participant are interese/cerinte specifice
- **Fluxul de activități** (procesul business) existent într-o organizatie, **pe care noul sistem informatic trebuie sa-l îmbunatateasca** – introducerea noului sistem nu trebuie sa produca schimbari neplanificate.
- **Studiul de fezabilitate**: este produsul realizabil cu satisfacerea cerintelor specificate, d.p.d.v. tehnic si financiar (costurile prevazute/negociate)?

Tehnici de extragere a cerintelor

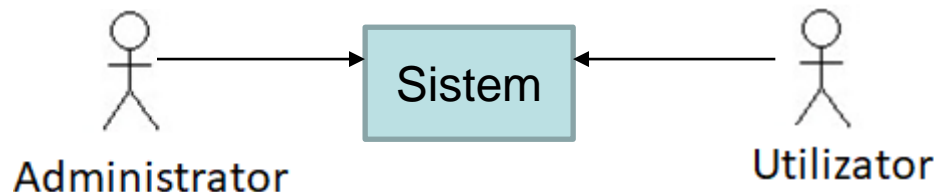
Difera, în funcție de natura produsului software:

- Produs nou destinat pieței
 - **Studiul produselor software similare existente** – noul produs trebuie să ofere ceva în plus: funcționalitate, ușurința în utilizare, tehnologie nouă, etc.
- Sistem informatic
 - **Studiul fluxului de activități al organizației în care va fi implementat sistemul** informatic, entitățile, taskurile, etc, modul în care poate fi îmbunătățit fluxul de activități prin introducerea unui sistem informatic.
- **Interviuri** cu viitorii utilizatori, clientul, reprezentanți ai autorităților din domeniul de operare al produsului software.
- **Descriere scenarii și cazuri de utilizare ale viitorului sistem**
- **Utilizarea de prototipuri executabile ale viitorului sistem**

Activități de definire a cerintelor

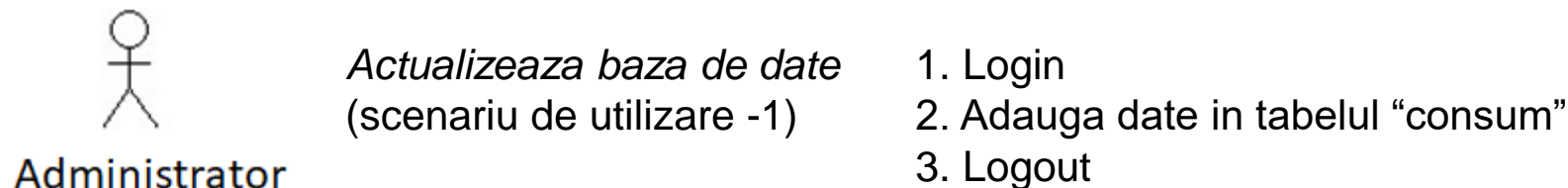
Definirea cerințelor utilizator include următoarele activități efectuate de dezvoltator:

1. Identificarea actorilor: entitățile externe sistemului care vor interacționa cu sistemul



2. Identificarea și descrierea scenariilor de utilizare: se definesc scenarii pentru functionalitățile tipice care vor fi furnizate de viitorul sistem.

- Scenariile sunt exemple concrete de utilizare a viitorului sistem
- Exprimate în limbaj natural: ușurează comunicarea dezvoltatorului cu utilizatorii viitorului sistem și înțelegerea domeniului aplicației



Activități de definire a cerintelor



Administrator

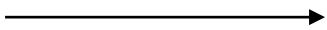
Actualizeaza baza de date
(scenariu de utilizare -2)

1. Login
2. Modifica date in tabelul "consum"
3. Eroare la salvarea tabelului

3. Definirea cazurilor de utilizare ale sistemului: plecand de la scenarii, dezvoltatorii definesc un set de cazuri de utilizare care descriu toate posibilitatile de utilizare a viitorului sistem.



Administrator



Actualizeaza
baza de date

Cazul de utilizare
"Actualizeaza baza de date"
unifica scenariile de utilizare in
interactiunea dintre Administrator si sistem

4. Rafinarea cazurilor de utilizare: se detaliaza cazurile de utilizare descriind comportarea sistemului în prezența erorilor și a condițiilor exceptionale

5. Identificarea relatiilor dintre cazurile de utilizare

6. Identificarea și definirea cerintelor nefunctionale: constrangeri privind performanta sistemului, consumul de resurse, cerinte de securitate, etc.

Identificarea actorilor (1)

➤ Un actor este un rol pe care o entitate externa îl joaca in raport cu sistemul:

- Un utilizator direct al sistemului (un utilizator direct poate juca mai multe roluri)
- Un echipament extern sau alt sistem care comunica cu sistemul analizat

1. Se determina tipurile de persoane care vor avea o legatura cu noul sistem informatic:

- Care vor utiliza direct sistemul
- Care vor utiliza rezultatele produse de sistem (de ex. rapoarte din baza de date)
- Care nu vor utiliza sistemul dar pot influenta conceptia sa (impun reguli in organizatie)
- Se determina **tipurile de utilizatori directi si operatiile pe care acestia le vor efectua cu sistemul**, sarcini, abilitati de lucru cu calculatorul, cunostintele lor tehnice.

2. Se determina entitatile externe sistemului (alte sisteme sau echipamente), care vor comunica cu sistemul.

Identificarea actorilor (2)

STUDIU DE CAZ: SGCB - Sistem de gestiune electronica a cartilor din mai multe biblioteci

Se doreste realizarea unui sistem informatic pentru gestiunea centralizata a cartilor existente in mai multe biblioteci.

Sistemul trebuie sa ofere urmatoarele functionalitati:

- Sa permita inregistrarea persoanelor ca abonati
- Sa permita abonatilor sa caute si sa imprumute sau sa restituie carti
- Sa permita inregistrarea de noi carti si scoaterea din evidenta a unora existente
- Sa pastreze evidenta abonatilor si a cartilor imprumutate de fiecare abonat.

Deoarece sistemul realizeaza o gestiune centralizata, pentru accesarea sa se propune o interfață Web.

Identificarea actorilor (3)

Din descrierea anterioara rezulta ca vor exista doua tipuri de utilizatori directi ai sistemului:

- Persoanele care vor accesa sistemul pentru a căuta, împrumuta sau restitui cărți
- Persoanele care vor gestiona abonatii și cartile din biblioteci: vor înregistra noi carti sau vor elimina pe unele existente, vor actualiza numarul de carti din fiecare exemplar, etc.

➤ **Identificarea actorilor: pe baza operatiilor efectuate cu sistemul de utilizatorii direcți.**

Se identifica două roluri:

- Rolul de abonat
 - Rolul de bibliotecar
- O persoana poate juca atat rolul de abonat cat si rolul de bibliotecar.
- Fiecare dintre cele 2 roluri poate fi jucat de mai multe persoane.
- Actorii identificati sunt **Abonat** și **Bibliotecar**, reprezentati în UML astfel:



Identificarea actorilor (4)

Intrebări care pot ajuta la identificarea actorilor:

- Care grupuri de utilizatori vor folosi direct în munca lor noul sistem?
- Care grupuri de utilizatori vor utiliza principalele functii ale sistemului?
- Care grupuri de utilizatori vor efectua operatii secundare, cum ar fi cele de mentenanta sau de administrare?
- Care sunt sistemele externe hardware sau software care vor comunica cu sistemul?

Scenarii de utilizare (1)

- Un **scenariu** este o descriere narativa a unei singure functionalitati a unui sistem informatic, prin prisma unei interacțiuni concrete dintre un actor și sistem.
- Scenariile ușurează extragerea cerințelor, fiind un instrument ușor de înțeles de utilizatori si clienți.

Exemple de scenarii de utilizare a SGCB pentru functionalitatea “Imprumut cărți”:

Scenariul Imprumut -1

1. Un utilizator acceseaza interfata web a sistemului in sectiunea pentru imprumut carti si completează rubricile rezervate numelui de utilizator și parolei de acces, apoi apasa butonul “Submit”.
2. Sistemul preia datele și verifica identitatea utilizatorului.
3. Sistemul afișează mesajul: „Nume de utilizator inexistent. Nu sunteti înregistrat ca abonat. Efectuati procedura de înregistrare”.

Scenarii de utilizare (2)

Scenariul Imprumut -2

1. Un utilizator acceseaza interfata web a sistemului in sectiunea pentru imprumut carti si completează rubricile rezervate numelui de utilizator și parolei de acces, apoi apasa butonul "Submit".
2. Sistemul preia datele și verifica identitatea utilizatorului.
3. Sistemul afișează mesajul: „Ati depasit numarul maxim de cărți ce pot fi împrumutate. Restituiti o parte dintre ele”.

Scenariul Imprumut -3

1. Un utilizator acceseaza interfata web a sistemului in sectiunea pentru imprumut carti si completează rubricile rezervate numelui de utilizator și parolei de acces, apoi apasa butonul "Submit".
2. Sistemul preia datele și verifica identitatea utilizatorului.
3. Sistemul afișeaza formularul de împrumut.
4. Abonatul completeaza formularul de împrumut, cu titlul cartii, numele și prenumele autorului și codul ISBN al cartii apoi apasa butonul "Submit".
5. Sistemul preia datele și cauta cartea.
6. Sistemul afișează mesajul: „Cartea nu există în bibliotecile noastre”.

Scenarii de utilizare (3)

Intrebari care pot ajuta la identificarea scenariilor:

- Care sunt operatiile pe care un anumit actor doreste sa le efectueze cu sistemul?
- Care sunt datele pe care le acceseaza actorul? Cine creaza aceste date? Pot fi ele modificate sau eliminate? De catre cine?
- Care sunt schimbarile externe pe care actorul trebuie sa le comunice sistemului? Cand?
- Care sunt evenimentele pe care sistemul trebuie sa le comunice actorului? Cat de repede?

Cazuri de utilizare (1)

Definirea cazurilor de utilizare ale sistemului

- Numarul de scenarii de utilizare a unui sistem este foarte mare!
- Un set de scenarii de utilizare care descriu aceeasi functionalitate se abstractizeaza într-un **caz de utilizare**.
- Un scenariu este o instanta a unui caz de utilizare.
- Un caz de utilizare descrie o functionalitate a sistemului în raport cu un actor; este o abstractizare a tuturor scenariilor care descriu acea functionalitate.
- Totalitatea cazurilor de utilizare ale unui sistem reprezintă toate modurile în care poate fi utilizat sistemul respectiv.
- Un caz de utilizare este initiat de un actor.
- Dupa initierea sa, cazul de utilizare poate interactiona si cu alti actori.
- Un caz de utilizare reprezinta un flux complet de evenimente prin sistem, declanșat ca urmare a initierii sale.

Cazuri de utilizare (2)

Deși UML admite variații în descrierea cazurilor de utilizare, **în general un caz de utilizare este descris prin secvența tipică de pași (comportamentul de bază) și alternativele la secvența tipică.**

Exemplu: Cazul de utilizare "Împrumut" al sistemului de gestiune a cartilor (SGCB)

Fluxul de baza (secvența tipică de pași) – descrie scenariul în care împrumutul poate fi efectuat

1. Un utilizator accesează interfața web a sistemului în secțiunea pentru împrumut cărți și completează rubricile rezervate numelui de utilizator și parolei de acces, apoi apasă butonul "Submit".
2. Sistemul preia datele și verifică identitatea utilizatorului.
3. Sistemul afișează formularul de împrumut.
4. Abonatul completează formularul de împrumut, cu titlul cărții, numele și prenumele autorului și codul ISBN al cărții apoi apasă butonul "Submit".
5. Sistemul preia datele și caută cartea.
6. Sistemul înregistrează împrumutul.
7. Sistemul afișează mesajul "Puteți lua cartea de la ghișeul bibliotecii cu adresa".

Cazuri de utilizare (3)

Alternative:

La pasul 3:

3a) Utilizatorul nu este înregistrat ca abonat.

Sesiunea este încheiata de sistem, cu mesajul: „Nume de utilizator inexistent. Nu sunteți înregistrat ca abonat. Efectuați procedura de înregistrare”.

3b) Utilizatorul este înregistrat dar a depășit numărul maxim admis de cărți împrumutate.

Sesiunea este încheiată de sistem cu mesajul:

„Ați depășit numărul maxim de cărți împrumutate. Restituiți o parte dintre ele”.

La pasul 6:

6a) Cartea nu este găsită.

Sesiunea este încheiată de sistem cu mesajul: „Cartea nu există în bibliotecile noastre”.

Cazuri de utilizare (4)

Repetarile de comportament intr-un caz de utilizare pot fi descrise prin formulari de tipul:

n. **repetă**

n.1. -----

n.2.-----

până (conditie)

sau

n. **cat timp (conditie) repetă**

n.1.

n.2.

Sunt admise, de asemenea, formulari de tipul:

if conditie se continua cu pasul x

sau

if conditie se continua cu pasul x

else se continua cu pasul y

Cazuri de utilizare (5)

Descrierea tipică a unui caz de utilizare cuprinde următoarele elemente:

- **Descrierea comportamentului de bază al sistemului** (fluxul principal de evenimente și operațiile declanșate) **și alternativele** – o descriere pas cu pas a acțiunilor actorului și sistemului.
- **Pre-condiția (opțional)** – o constrângere asupra sistemului la inițierea cazului de utilizare; se specifică în limbaj natural.
- **Post-condiția (opțional)** – o constrângere asupra sistemului la terminarea execuției cazului de utilizare; se specifică în limbaj natural.
- **Cerințe speciale** – cerințe ce nu pot fi descrise cu ușurință în fluxul de evenimente.
- **O schiță a interfeței utilizator** la executia cazului de utilizare **(opțional)** .

In faza de analiza a cerintelor, cazurile de utilizare se rafinează si formalizează folosind diagrame de secvență UML pentru descrierea scenariilor și diagrame de activitate UML pentru descrierea operațiilor mai importante.

Cazuri de utilizare (6)

- De regula, preconditionia precede descrierea cazului de utilizare iar postconditia o succede, dar este posibil ca ambele sa fie specificate fie inaintea descrierii cazului de utilizare, fie dupa.

De exemplu, pentru cazul de utilizare anterior:

Preconditie:

Postconditie: In baza de date a sistemului exista o inregistrare a imprumutului catre abonat

In acest caz, preconditionia are valoarea *true* intotdeauna.

In UML, un caz de utilizare se reprezinta grafic printr-o elipsa in interiorul careia este scris numele cazului de utilizare.

Relatia dintre un actor si un caz de utilizare se reprezinta printr-o **linie de comunicare**:

