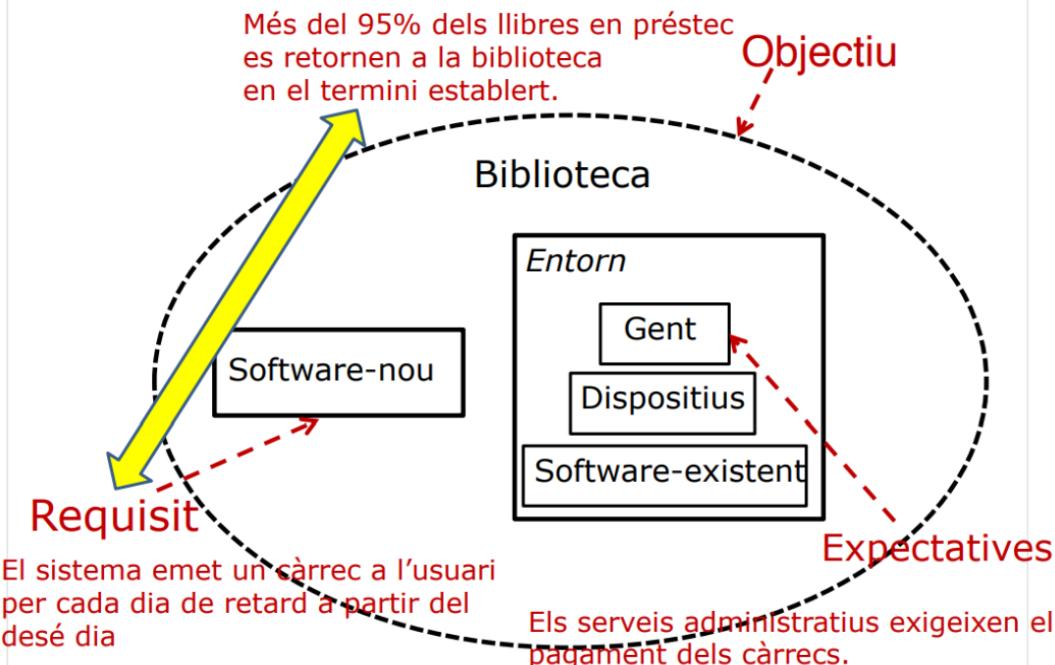


Príncipi de necessitat dels requisits:

Per desenvolupar un sistema cal definir els seus requisits

Requisit és una condició que ha de satisfer el software-nou (a desenvolupar). Es formula únicament en termes dels fenòmens compartits entre el software i el seu entorn



• Requisit funcional

Defineix els efectes funcionals que el software ha de tenir en el seu entorn (what the product has to do).

Ex: "*El sistema ha de proporcionar una llista de tots els llibres de la biblioteca sobre un tema determinat*"

• Requisit de qualitat

Estableix propietats de qualitat que el software o la seva execució han de tenir. (how well)

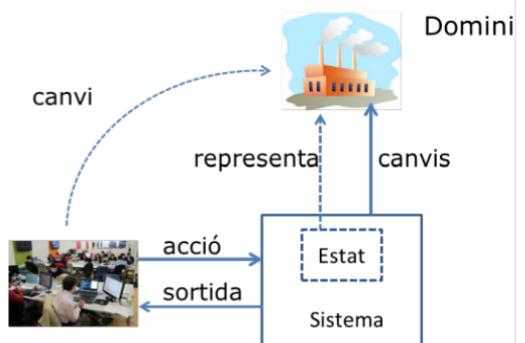
Ex: "*Un usuari no administratiu mai pot saber els llibres que han estat prestats a altres*"

• Restricció

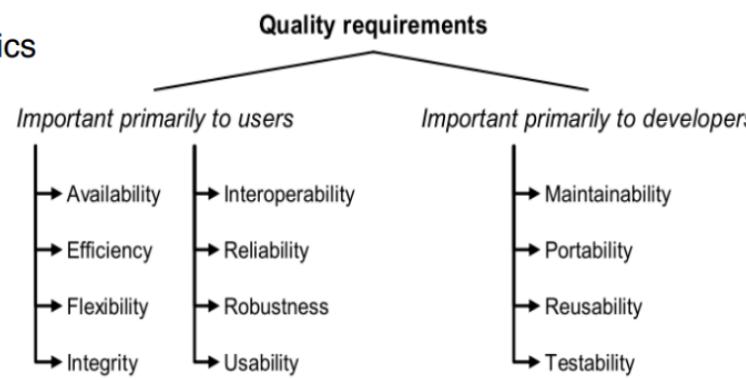
És una condició **exigida d'entrada** sobre com s'ha de desenvolupar el sistema, el context en el qual haurà de funcionar o les característiques que ha de tenir.

Ex: "*El cost total del software de la biblioteca no pot excedir X*"

- Conceptes del domini representats en el sistema
- Definicions dels conceptes
- Restriccions integritat
- Esdeveniments o accions:
 - Quan es produeixen
 - Restriccions
 - Efecte sobre l'estat
 - Sortida



- Percepció (Look and feel)
- Usabilitat i humanitat
- Rendiment
- Funcionament i ambientals
- Manteniment i suport
- Seguretat
- Culturals i polítics
- Legals



Propietats dels requisits

- **Abstracte.** Independent de la implementació.
- **No ambigu.** Es pot interpretar d'una sola manera.
- **Traçable.** Se sap d'on prové
- **Verifiable.** Hi ha una manera de comprovar que el sistema el satisfà.

Exemple:

El sistema permet a un usuari registrat reservar

amb el mòbil

(a qualsevol moment del dia)

Una taula en un restaurant registrat per una data determinada (dinar o sopar).

Si la taula està disponible,

El sistema confirma la reserva
immediatament.

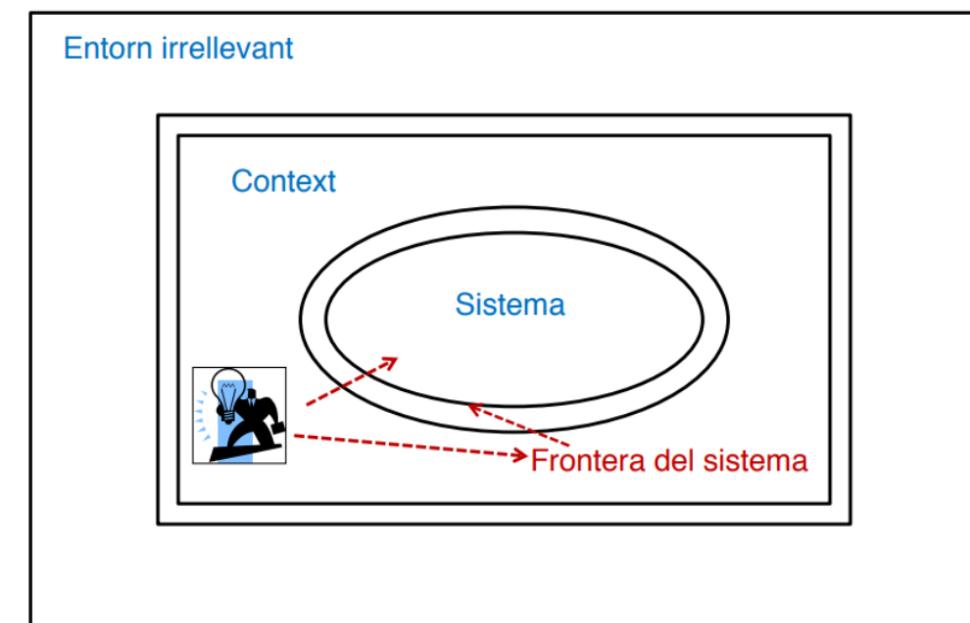
(funcional)
(restricció)
(qualitat)

- **Requisit emergent:** requisit (funcional o de qualitat) que sorgeix (“emergeix”) mentre es desenvolupa el sistema, o quan el sistema entra en funcionament.

Sistema: La part del món que podem dissenyar/canviar (software, procediments, dispositius, documents,...)

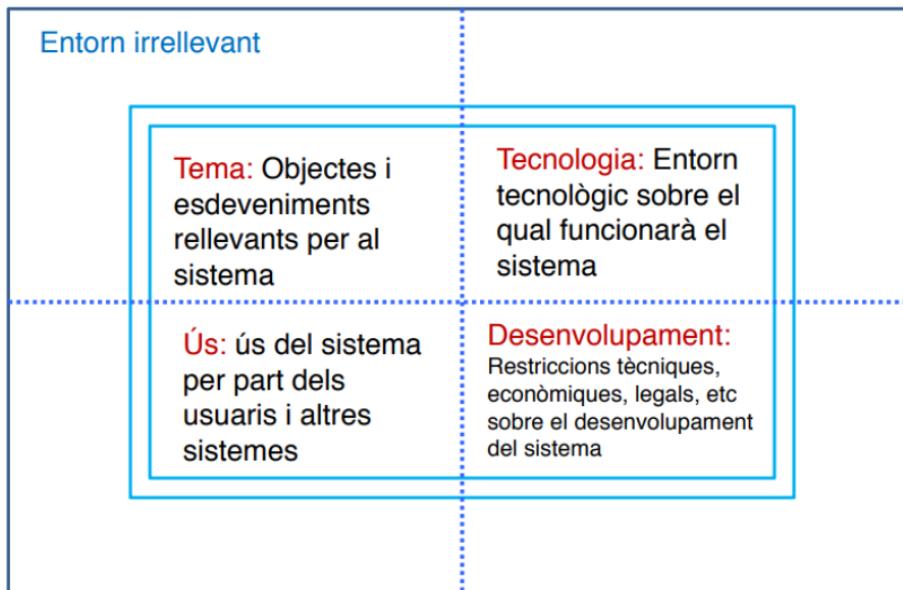
Entorn: Tota la resta

El **context** d'un sistema és la part de l'entorn del sistema, rellevant per definir, comprendre i interpretar-ne els seus requisits.



La **frontera del sistema** separa el sistema a desenvolupar del context del sistema.

Facetes del context



Context: objectes a analitzar

Tema

Persones
Productes
Processos
...

Tecnologia
Hardware
Software
Estratègia
...

Ús

Grups usuaris
Interfícies
Interaccions amb altres sistemes
...

Desenvolupament
Recursos
Eines a usar
...

Desenvolupament

Context: Parts interessades (stakeholders)

Tema

Experts domini

Experts legals

Usuaris directes

Usuaris indirectes

Experts interfícies

Ús

Tecnologia

Responsables
plataforma
tecnològica

Responsables
desenvolupament,
operació,
manteniment

Desenvolupament

Context: Documents

Tema

Models de referència

Lleis

Llibres de text

Ús

Tecnologia

Pla estratègic
Documentació APIs

Processos de negoci
Estàndards

Normes de
desenvolupament

Desenvolupament

Tema

Sistema actual

Sistemes similars

Sistemes anàlegs

Tecnologia

Sistemes existents

Ús

Sistemes existents
que s'usen de
manera semblantEines de
desenvolupament

Desenvolupament

Els stakeholders

Els *stakeholders* o parts interessades són...

- persones
- grups
- organitzacions

que d'una manera o altra...

- estan involucrats de forma activa en el projecte
- són afectats pel procés o el resultat
- poden influenciar el procés o el resultat

Els stakeholders

Algunes de les parts interessades tradicionals són:

- Promotor
 - Client
 - Usuaris
 - Consultors
 - Gestors
 - Equip principal
- de vegades són els mateixos
- Inspectors
 - Personal de marketing
 - Experts legals
 - *Stakeholders* negatius
 - Govern
 - Experts tècnics

Poden representar els interessos generals d'autoritats superiors. Per exemple

- Responsables de privadesa de dades
- Advocats
- Agents de patents
- Membres del comitè d'empresa

De vegades hi ha *stakeholders* que no coneixen l'existència del projecte, però dels que hem de comprendre el seu possible impacte en el sistema.

Quins són els *stakeholders* del sistema de compra d'entrades online d'un cinema?

- Espectadors
 - Usuaris directes → utilitzen el producte o servei per aconseguir un objectiu
- Director/a
 - Promotor → responsable de la inversió per fer el projecte
- Empleats de taquilla
 - Beneficiaris funcionals → es redueix la càrrega de treball a taquilla
 - Consultors interns → aporten informació sobre el funcionament actual
- Altres cinemes
 - *Stakeholder* negatiu → competidor que desitja el fracàs del sistema
- Equip de desenvolupament
 - Membres del nucli de l'equip → encarregats de dur a terme el projecte

L'enginyer de requisits

Les tasques que ha de desenvolupar són:

- Identificar les parts interessades crítiques o *stakeholders*
- Descobrir què esperen o desitgen (els objectius dels *stakeholders*)
- Determinar el conjunt de requisits del software

Les parts interessades en el context del sistema

Es poden classificar segons les diferents facetes del context:

- Tema: aporten informació sobre el domini del sistema
 - Experts domini, experts legals...
- Ús: usuaris, beneficiaris o influències sobre el sistema
 - Usuaris directes, usuaris indirectes, experts interfícies...
- Tecnologia: associats a la planificació, disseny i operació de l'entorn tècnic
 - Responsables plataforma tecnològica...
- Desenvolupament: construeixen i planifiquen el sistema
 - Responsables desenvolupament, operació i manteniment...

Ambdós són *stakeholders* de la faceta d'ús, l'empleat interactua directament amb el sistema i la cap de departament se'n beneficia alhora que anima el seu ús.



Usuari directe



Sistema



Usuari indirecte

Les parts interessades en el context del sistema

En un sistema de seguretat del cotxe que supervisi l'atenció del conductor i prengui mesures adequades per evitar que el conductor s'adormi....

- Tema
 - Físics, investigadors d'accidents, experts del sistema de frenada
- Ús
 - Conductor, copilot, dissenyador del *cockpit* (*cabina*)
- Tecnologia
 - Responsables de les centraletes, responsables del protocol de comunicació del vehicle
- Desenvolupament
 - Enginyer de software, enginyer elèctric, tècnic de centraletes



Les parts interessades en l'ER

Les 3 activitats principals de l'enginyeria de requisits són:

- Elicitació (o extracció)
- Documentació
- Negociació

i les 2 activitats transversals són:

- Validació
- Gestió

Però és a l'elicitació, la negociació i la gestió on els *stakeholders* prenen un paper més rellevant. La validació té el seu propi capítol fora de l'àmbit d'aquesta sessió.

Requisits - Elicitació

L'objectiu de l'activitat d'elicitació de requisits és:

1. Identificar les fonts d'informació rellevants
 - a. Parts interessades involucrades en el procés
 - b. Documentació existent
 - c. Sistemes existents
2. Obtenir els requisits de les fonts d'informació rellevants
3. Desenvolupar requisits nous i innovadors

Ens centrarem en l'identificació d'*stakeholders* rellevants (1a) i l'obtenció de requisits a partir d'ells (2).

Identificar els *stakeholders* rellevants és clau per aconseguir una bona especificació de requisits. Una elicitació incompleta es tradueix en un sistema de baixa qualitat.

Quan els recursos disponibles (temps, diners) són limitats, el nombre d'*stakeholders* que participen en el procés d'obtenció de requisits queda limitat.

- Tema
 - Físics, investigadors d'accidents, experts del sistema de frenada
- Ús
 - Conductor, copilot, dissenyador del cockpit
- Tecnologia
 - Responsables de les centraletes, responsables del protocol de comunicació del vehicle
- Desenvolupament
 - Enginyer de software, enginyer elèctric, tècnic de centraletes



Un cop s'han identificat els *stakeholders* rellevants es poden aconseguir requisits a partir de:

- **Converses** en entrevistes o tallers
- **Qüestionaris** on els mateixos *stakeholders* documenten els requisits
- **Observació** de rutines o comportament d'*stakeholders*

Requisits - Negociació

Les opinions de les parts implicades poden divergir i estar en conflicte. L'objectiu de l'activitat de negociació és:

1. Detectar i explicitar tots els conflictes
2. Analitzar la causa o arrel de cada conflicte
3. Resoldre els conflictes
4. Documentar la resolució del conflicte i el raonament utilitzat

Idealment el sistema considerarà i assolirà les necessitats i desitjos de tots els *stakeholders*. En la pràctica això no és senzill d'aconseguir.

Un conflicte es produeix quan...

- les necessitats o interessos de diferents parts interessades es contradueixen
- o...
- algunes d'aquestes necessitats o interessos no es poden tenir en compte.

No resoldre els conflictes pot provocar la pèrdua de suport dels *stakeholders* o inclus el fracàs del projecte.

Els conflictes també poden servir com a font de noves idees i requisits innovadors.

Requisits - Negociació

Els escenaris originats a la fase d'especificació exemplifiquen situacions concretes de l'ús del sistema.

Aquests escenaris es poden fer servir per:

1. Il·lustrar conflictes i detectar la causa real
2. Documentar alternatives o solucions al conflicte

Les parts tècniques involucrades (arquitectes, desenvolupadors, testers...) han de comunicar els conflictes detectats i evitar intentar resoldre'ls per elles mateixes.

Els conflictes poden aparèixer en diverses etapes de l'enginyeria de requisits:

- Durant una activitat en l'elicitació de requisits, dos *stakeholders* presenten requisits que es contradueixen
- Documentant requisits originats durant different entrevistes
- Prioritzant requisits, hi ha dos *stakeholders* amb diferents opinions
- Validant un requisit, hi ha un *stakeholder* que el considera correcte i un altre que no
- Resolent un conflicte n'apareix un de nou

Un conflicte d'informació o dades a l'exemple del sistema de seguretat al vehicle...

Un grup d'*stakeholders* demana una configuració inicial completa del sistema de forma manual i un altre grup rebutgen fer-ho perquè creuen que no és necessari que els conductors del vehicle facin aquesta feina.

El segon grup d'*stakeholders* ha interpretat que la configuració és per part dels conductors mentre que el primer grup d'*stakeholders* pensava en la configuració a fàbrica.



Un conflicte d'interessos a l'exemple del sistema de seguretat al vehicle...

L'equip de desenvolupament de la pantalla del vehicle demana mostrar la informació de seguretat a la pantalla del vehicle i els experts en accidents rebutgen fer-ho perquè creuen que distraurà al conductor.

Un conflicte de valors a l'exemple del sistema de seguretat al vehicle...

Uns *stakeholders* volen utilitzar senyals acústics per alertar al conductor del vehicle de qualsevol perill i uns altres volen utilitzar senyals visuals per la mateixa tasca.



Heurística per l'anàlisi de conflictes

1. Comprovar si és un conflicte d'informació
 - a. Demanar als *stakeholders* una explicació del requisit en conflicte
 - b. Si cap de les explicacions divergeix del requisit, hi ha un conflicte d'informació
2. Preguntar sobre els interessos dels *stakeholders*
 - a. Esbrinar els objectius dels *stakeholders* en relació al requisit en conflicte
 - b. Si els objectius entren en contradicció, hi ha un conflicte d'interessos
3. Clarificar el context d'avaluació dels *stakeholders*
 - a. Preguntar als *stakeholders* sobre el context del requisit en conflicte
 - b. Si hi ha diferències en el context d'avaluació, hi ha un conflicte de valors

En trobem de diferents tipus:

- Conflicte de dades o informació
- Conflicte d'interessos
- Conflicte de valors
- Conflicte de relació
- Conflicte estructural

Requisits - Gestió

Control del procés i del canvi.

- Emmagatzematge persistent dels requisits
- Priorització dels requisits
- Gestió del canvi (traçabilitat)
- Planificació i control de totes les activitats

Requisits – Gestió – Priorització

Quan els recursos disponibles (temps, diners) són limitats, no tots els requisits prenen la mateixa importància o s'acaben desenvolupant amb el mateix nivell de realització.

La prioritització ens permet endreçar els diferents requisits segons la seva importància. Ocorre durant les diferents activitats de l'enginyeria de requisits.

Identificar els *stakeholders* rellevants per a la prioritització és el primer pas. S'han de **seleccionar en funció del criteri de prioritització** (importància, cost, durada, risc...) però almenys un representant de cada faceta del context ha d'estar representat.

Els escenaris originats a la fase d'especificació exemplifiquen situacions concretes de l'ús del sistema.

Els *stakeholders* poden determinar la prioritat dels requisits basant-se en la prioritat que donen a cada escenari. En altres paraules, prioritzar els escenaris per implícitament prioritzar els requisits associats.

Els objectius

Un objectiu és quelcom que volem aconseguir perquè n'obtenim un benefici.

En un projecte de software els defineix l'enginyer de requisits.

En trobem de dos tipus:

- Objectius dels *stakeholders*
- Objectius del projecte

Son la base per definir els requisits del software.

Objectius

Enginyer de requisits

Requisits

El promotor i la visió

El promotor és el responsable del projecte, aquell qui té l'autoritat i normalment paga el desenvolupament. Té l'última paraula sobre què fa el sistema, com ho fa, i com ha de ser.

- Defineix el projecte
- Coneix bé els objectius, les prioritats i l'impacte de negoci
- Proporciona els recursos i motiva l'equip

La visió és la idea principal que dona cos al projecte. Pot ser més o menys abstracta ja que defineix quin canvi aportarà però no com s'implementarà.

Serveix com a guia per a tots els que participen en el projecte, però no és suficient per definir tots els requisits amb el detall necessari.

Objectius i rol de les parts interessades

Possibles parts interessades del projecte per a modernitzar un diari tradicional:

- Subscriptors actuals
- Subscriptors digitals
- Lectors edició digital
- Anunciant
- Redacció
- Director de tecnologia



L'anunciant

té l'objectiu de...

- enviar anuncis específics a un públic objectiu, en el mitjà preferit (text, audio, video), i a un cost que és més atractiu que la millor alternativa que tenen els anunciant per aquell públic objectiu.

té el rol de...

- posar anuncis en la nova edició digital del diari.

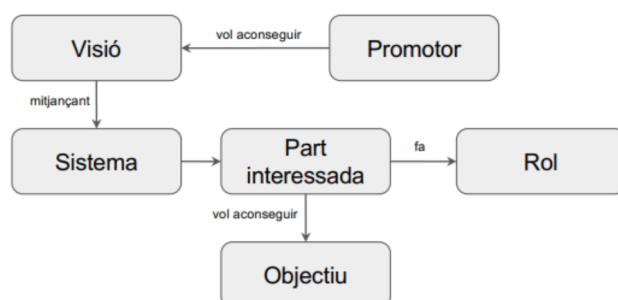
Un objectiu d'una part interessada en el projecte és:

- un benefici que espera aconseguir fent el rol que se li exigeix en el sistema
- el preu màxim que està disposada a pagar per això

Un *stakeholder* desenvoluparà el seu rol si el benefici (*V*) obtingut és igual o superior al desitjat i el cost (*P*) és igual o inferior al màxim que està disposat a assumir.

$$V \geq V_{\min} \quad \& \quad P \leq P_{\max}$$

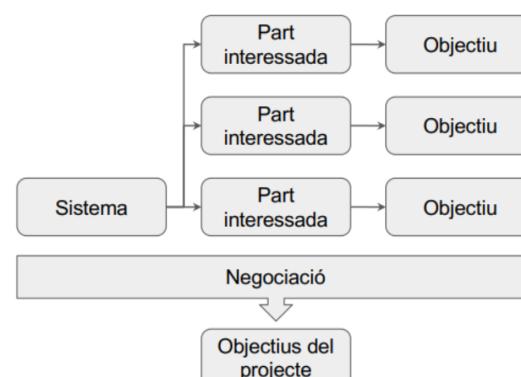
Una part interessada pot tenir diversos objectius. El preu màxim es pot referir a un conjunt de beneficis, no només a un d'ells.



Objectius del projecte

Possibles objectius d'un projecte per a modernitzar un diari tradicional:

- Oferir als subscriptors la versió en línia del diari
- Aconseguir nous subscriptors de l'edició digital
- Atreure lectors a l'edició digital
- No quedar enrere respecte de la competència
- Recuperar la inversió en uns anys
- Cobrir els costos d'explotació amb els anunciant



Els objectius del projecte són el resultat de la negociació dels diferents objectius de les diferents parts interessades.

No tots els objectius d'*stakeholders* esdevenen objectius del projecte. En funció de la rellevància de l'*stakeholder* i dels recursos disponibles es poden acabar ometent alguns o tots dels seus objectius.

Objectius opositius de diferents *stakeholders* es materialitzen en:

- Pèrdua dels objectius com a objectius del projecte → *lose - lose*
- Acceptació d'un objectiu i rebuig de l'altre → *win - lose*
- Creació d'un nou objectiu de projecte intermig → *win - win*

Objectius SMART

Definir objectius específics, mesurables i realistes requereix temps i esforç.

Definir objectius acordats i acotats temporalment requereix negociació entre diferents parts implicades.

Per tant, és preferible fer que els objectius del projecte (i no els dels *stakeholders*) esdevinguin SMART.

Specific

- ✗ Intentaré ser més fort
- ✓ Intentaré ser més fort aixecant peses

Measurable

- ✗ Conseguiré ser més fort aixecant peses
- ✓ Conseguiré ser més fort, aixecant peses fent 10 repeticions de 200 kg

Llista negra de termes: provar, podria, hauria, intentar, possiblement, suposadament, probablement, pot ser...

Agreed upon

- ✗ Conseguiré ser més fort, aixecant peses fent 10 repeticions de 200 kg
- ✓ Conseguiré ser més fort, aixecant peses fent 10 repeticions de 200 kg amb el suport de l'entrenador

U!! De vegades hi ha informació implícita, hem de valorar quan fer-la explícita.

Realistic

- ✗ Conseguiré ser més fort, aixecant peses fent 10 repeticions de 200 kg cada dia amb el suport de l'entrenador
- ✓ Conseguiré ser més fort, aixecant peses fent 5 repeticions de 90 kg cada dia amb el suport de l'entrenador



Time-based

- ✗ Conseguiré ser més fort, aixecant peses fent 5 repeticions de 90 kg cada dia amb el suport de l'entrenador
- ✓ Conseguiré ser més fort, aixecant peses fent 5 repeticions de 90 kg cada dia amb el suport de l'entrenador, abans del 17 de juliol

Llista negra de termes: aviat, en uns mesos, abans d'acabar l'any...



Objectius SMART

Objectiu base: Ser ric

S: Ser ric creant un software

M: Ser milionari creant un software

A: Ser milionari creant un software per empreses

R: Guanyar 100.000 euros creant un software per empreses

T: Guanyar 100.000 euros creant un software per empreses en menys de 5 anys

Aquestes qualitats són importants perquè:

- Ajuden a assolir i organitzar els objectius
- Faciliten la generació de requisits
- Eviten malentesos entre les diferents parts implicades

No tots els objectius han de tenir les 5 qualitats.

Validació d'objectius

És el procés de comprovar que els objectius (del projecte) recollits són els que realment volen assolir les diferents parts interessades. És a dir, són útils i prou ben definits.

Per a validar podem fer les següents tres preguntes:

1. Quina és la part interessada?
2. Quin és l'objectiu associat?
3. És SMART?



Un **cas d'ús** és
un **conjunt d'accions** realitzades per un sistema
que produeix un **resultat observable**
que és **valuós** per a un o més actors del sistema.

Un **escenari d'un cas d'ús** és una seqüència d'accions
que ocorre en unes certes condicions,
amb la intenció d'assolir l'objectiu de l'actor primari i
que obté un resultat particular.

Un escenari comença amb l'acció disparadora i continua
fins que s'asseoleix l'objectiu o s'abandona.

Primary Actor: Customer

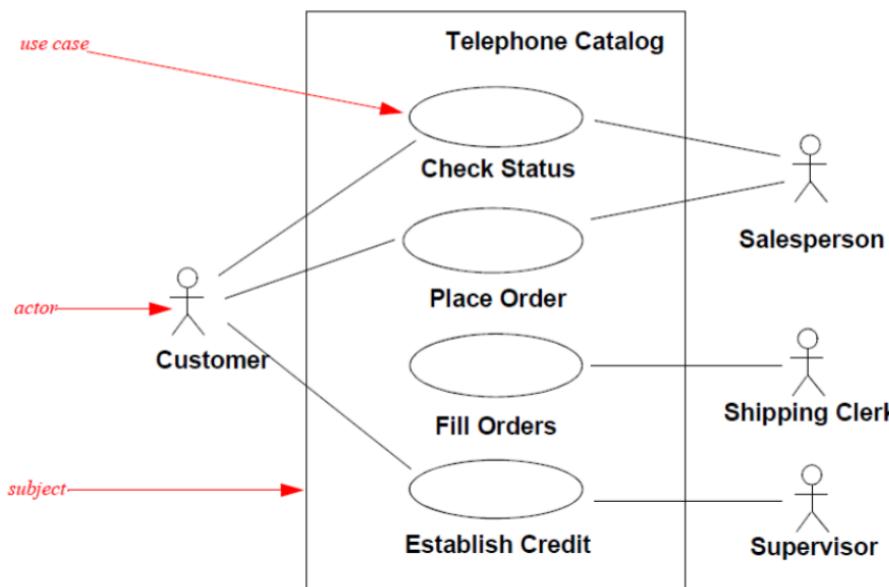
Precondition: None.

Trigger: A customer wants to place an order.

Main Success Scenario:

1. At any time before step 8 the customer logs in.
2. The customer browses the product catalog.
3. The customer selects a product to buy.
4. The system displays the contents of the shopping cart.
5. The customer changes the contents of the shopping cart.
6. The system updates the shopping cart.
7. The system displays the contents of the updated shopping cart.
The customer repeats steps 3 and 5 as necessary to build his order.
8. The customer checks out the order.
9. The system shows the shipping address and the available shipping methods.
10. The customer selects the preferred shipping method.
11. The system shows the billing address and the available payment methods.
12. The customer selects the preferred payment method.
13. The system displays a summary of the order.
14. The customer confirms the order.
15. The system saves the order.
16. The system sends an email to the customer with the information about the order.

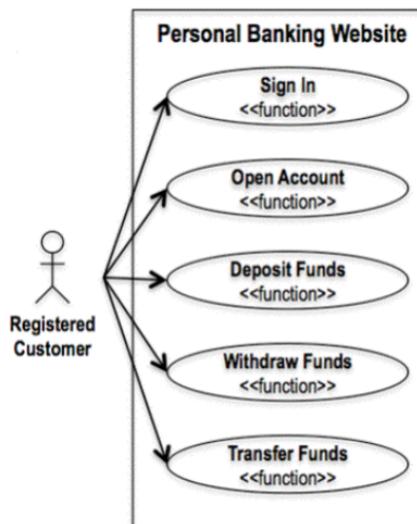
19



Extensions:

- 1a. The customer is new:
 - 1a1. Create customer.
- 14a. The customer wants to change the contents of the shopping cart:
 - 14a1. The customer changes the contents of the shopping cart.
 - 14a2. The customer continues with the checkout procedure at step 9.
- 11a, 14b. The customer wants to change the shipping address:
 - 11a1. The system shows the known addresses of the customer.
 - 11a2. The customer selects a different shipping address.
 - 11a3. The customer continues with the checkout procedure at step 11.

Use case diagram



Use case preconditions

None

1. User is signed in

1. User is signed in
2. User has at least one active account1. User is signed in
2. User has at least one active account
3. At least one of those accounts has a positive balance1. User is signed in
2. User has at least two active accounts
3. At least one of those accounts has a positive balance

Fer comanda

«include»

Crear client

Una **història d'usuari** descriu una funcionalitat que és valiosa per a un usuari o client d'un sistema.

Consta de tres parts:

- Card (tarja): Descripció escrita de la història d'usuari
- Conversation: Els detalls es conversen amb els stakeholders (owner)
- Confirmation: Condicions de satisfacció/acceptació que confirmen que la historia s'ha desenvolupat correctament

Three Cs

Card

- Stories are traditionally written on note cards.
- Cards may be annotated with estimates, notes, etc.

Conversation

- Details behind the story come out during conversations with product owner

Confirmation

- Acceptance tests confirm a story was coded correctly

As a **<type of user>**, I want **<some goal>** so that **<some reason>**.

Com a **usuari registrat del sistema** jo vull **poder demanar una nova contrasenya** de manera que jo pugui iniciar sessió si he **oblidat la que tenia**.

Detalls en les **Condicions de satisfacció/acceptació**

Com a **usuari**, puc cancel·lar una reserva meva d'hotel, per tal de no haver-ne de pagar el seu cost.

Condició de satisfacció:

- Un membre preferent pot cancel·lar fins al darrer minut sense penalització.
- A un membre no preferent se li carrega un 10% per una cancel·lació del mateix dia.
- S'envia un email de confirmació.
- S'informa l'hotel de la cancel·lació.

Detalls en **Subhistòries més petites**

Com a **usuari**, jo puc cancel·lar una reserva meva d'hotel, per tal de no haver-ne de pagar el seu cost.

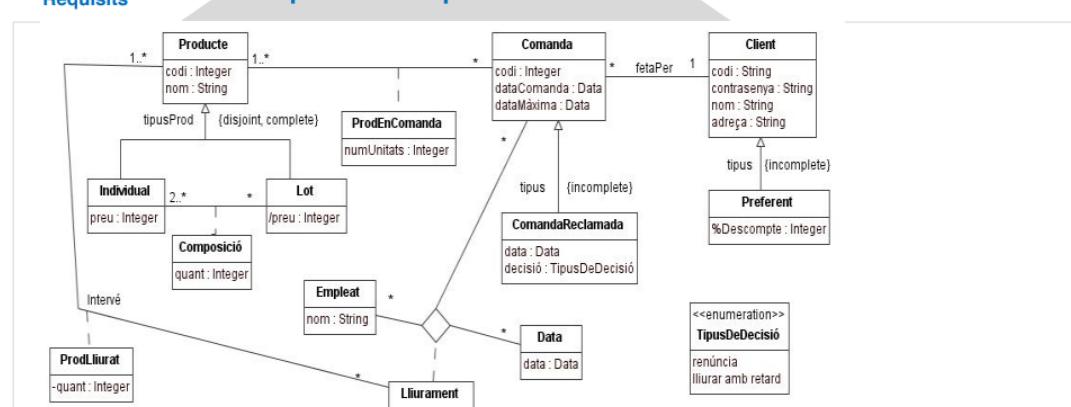
Com a **client preferent** puc cancel·lar la meva reserva d'hotel fins al darrer minut

Com a **client no preferent** puc cancel·lar la meva reserva d'hotel amb antelació mínima de 24 hores

Com a usuari he de rebre un email de confirmació de qualsevol reserva cancelada

Una bona **història d'usuari** hauria de ser:

- **Independent**: autocontinguda, no depèn de cap altra.
- **Negociable**: no és un contracte explícit i ha de deixar espai per a negociar.
- **Valuosa** per a usuaris o clients
- **Estimable**: S'ha de poder estimar el tamany
- Ben dimensionada (**Sized appropriately**): no ha de ser tan gran que sigui impossible planificar/prioritzar amb un cert nivell de certesa
- **Verificable (Testable)**: ha de proporcionar la informació necessària com perquè es puguin desenvolupar els tests.



Restriccions Textuals:

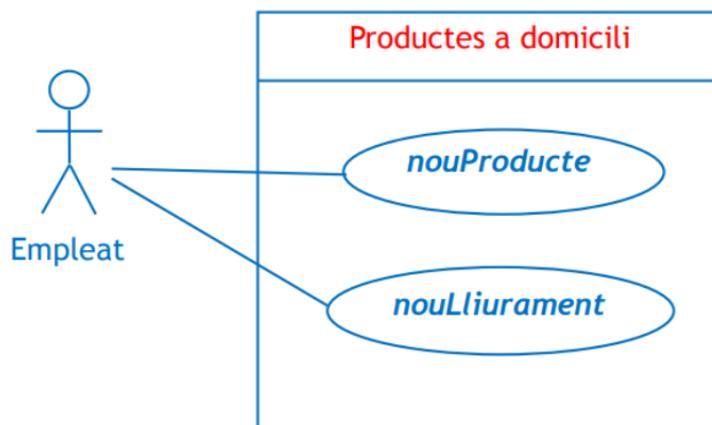
1. Claus externes de les classes no associatives: (Producte, codi); (Comanda, codi); (Client, codi i contrasenya); (Data, data); (Empleat, nom)
2. La data màxima d'una comanda ha de ser posterior o igual a la data de la comanda.
3. Un client no pot fer més de 5 comandes en una mateixa data.
4. La data de lliurament d'una comanda ha de ser posterior o igual a la data de la comanda.
5. Els productes d'un lliurament d'una comanda pertanyen a la seva comanda.
6. La suma de les quantitats dels productes lliurats d'una comanda no és superior a la suma de les quantitats dels productes demandats en la comanda.
7. No es pot fer un lliurament d'una comanda reclamada amb decisió="renúncia"
8. La data de reclamació d'una comanda ha de ser posterior a la data màxima de la comanda.

Informació derivada:

1. /preu d'un lot és la suma de: preu × quantitat dels productes individuals que el componen

Diagrama de casos d'ús:

Considerarem només dos casos d'ús: **nouProducte** i **nouLliurament**.

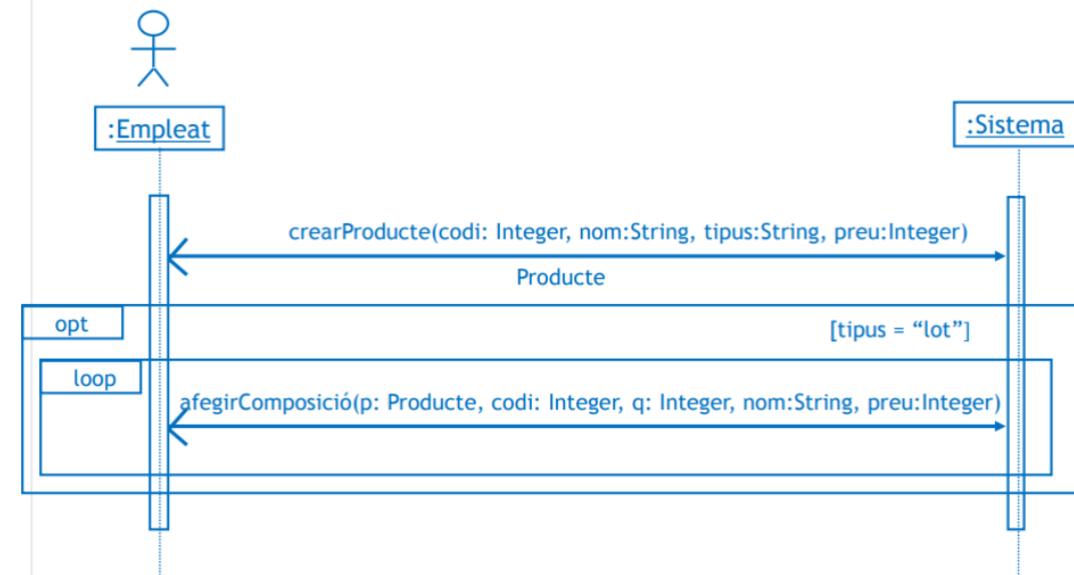


Especificació (brief style) dels casos d'ús:

nouProducte. Quan un empleat registrat al sistema vol crear un producte ha d'indicar el codi del producte i tota la informació necessària per a donar-lo d'alta. Si el producte que es crea és un lot, aleshores també caldrà indicar tota la informació dels productes individuals que el composen. Si un producte individual d'un lot no existeix en el moment de crear el lot, aleshores també caldrà crear aquest producte individual.

nouLliurament. Quan un empleat registrat al sistema decideix fer un nou lliurament indica totes les dades necessàries per identificar el lliurament i tota la informació necessària dels productes que conté. Un lliurament només es pot fer si tots els productes que conté ja existeixen en el moment de fer el lliurament. Un lliurament no té perquè fer referència a tota una comanda sencera.

Diagrames de Seqüència: nouProducte



Contractes de les operacions en OCL: nouProducte

context:

Sistema::crearProd(codi: Integer, nom:String, tipus:String, preu:Integer): Producte

post:

Producte.allInstances()->exists(p | p.ocllsNew() and p.codi=codi and p.nom=nom and
if tipus="Lot" then p.ocllsTypeOf(Lot) and result=p
else p.ocllsTypeOf(Individual) and p.ocllAsType(Individual).preu=preu endif)

context:

Sistema::afegirComposicio(p: Producte, codi:Integer, q:Integer, nom:String, preu:Integer)

pre:

post:

if not Individual.allInstances()@pre -> exists(i|i.codi = codi) then
Individual.allInstances()->exists(i | i.ocllsNew() and i.codi=codi and i.nom=nom
and i.preu=preu) endif
and Composicio.allInstances()->exists(c | c.ocllsNew() and c.quant=q and
c.individual.codi=codi and c.lot=p)

Hi ha tantes històries d'usuari com Casos d'ús

Requisits

(Històries d'Usuari i Escenaris)

DE BARCELONA ■ ■ ■ UPC BARCELONATECH

As a <type of user>, I want <some goal> so that <some reason>.

Feature: nouProducte

Com a Empleat autenticat al sistema, jo vull poder crear un nou producte de manera que jo pugui augmentar l'oferta de l'empresa

Criteris d'acceptació:

- Ha de poder crear un producte (individual o lot) sempre que vulgui
- No podrà crear un nou producte si ja n'hi ha un altre que tingui el mateix codi
- Ha de poder assignar un producte individual ja existent a un lot sempre que vulgui
- Hi ha tants escenaris de cada història d'usuari com criteris d'acceptació té.
- Els escenaris han de considerar també els casos en que es violen les restriccions d'integritat.
- Un escenari seria un camí d'execució determinat (dels molts possibles) del contracte d'una operació.

Requisits

Escenaris de nouProducte

DE BARCELONA ■ ■ ■ UPC BARCELONATECH

Escenari: alta de producte individual

Donat un empleat autenticat que vol donar d'alta un producte individual
Quan introduceix un codi de producte inexistent, un nom i un preu de producte
Aleshores es crea una instància de producte individual amb aquests atributs

Escenari: alta de producte ja existent

Donat un empleat autenticat que vol donar d'alta un producte individual o lot
Quan introduceix un codi de producte existent
Aleshores el sistema retorna l'excepció "producte ja existeix"

Escenari alta de producte lot

Donat un empleat autenticat que vol donar d'alta un producte lot
Quan introduceix un codi de producte inexistent i un conjunt de com a mínim dos productes individuals amb la seva quantitat de composició
Aleshores es crea una instància de producte lot al que se li associen els productes individuals amb la quantitat corresponent

Escenari afegir producte a lot

Donat un empleat autenticat que vol afegir una composició del lot
Quan entra un producte lot existent, un producte individual existent que no pertany al lot i una quantitat
Aleshores es crea una instància de composició entre el lot i el producte individual amb la quantitat corresponent

Escenari: alta de lot amb composició insuficient

Donat un empleat autenticat que vol donar d'alta un producte lot
Quan indica un conjunt de productes individuals inferior a dos
Aleshores el sistema retorna l'excepció "composició de lot insuficient"

ER
Enginyeria de Requisits

Functional vs Non-Functional

FACULTAT D'INFORMÀTICA DE BARCELONA FIB UPC UNIVERSITAT POLITÈCNICA DE CATALUNYA BARCELONATECH

Functional requirements define what a software product must do: its features and functions

"A user must be able to edit messages after they are sent."

Non-Functional requirements specify how the system must perform

"The message must be updated for all users in a chat within 0.1 seconds, given that all users are online and have LTE connection or better"

Non-Functional Requirements

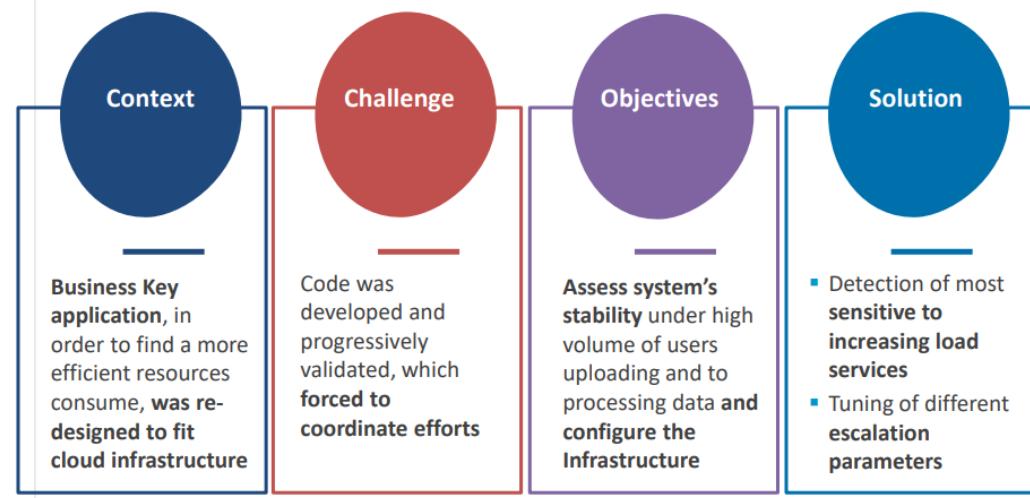
Performance	How fast a system should respond to requests
Scalability	How well a system can handle an increase in users or workload
Security	How well a system protects against unauthorized access and data breaches
Usability	How easy a system is to use
Maintainability	How easy it is to update and modify the system
Accessibility	Sustainability

11

NFRs - Inputs

- What is the system architecture and how is it built?
- How many end users will actually use the application?
- Where are these end users located?
- How many of these end users will use it concurrently?
- How will the end users connect to the application?
- How many additional end users will require access to the application over time?
- What will be the environment under test and what relationship (in terms of capacity) will it have with Production?
- What effect will the application have on key servers' consume of resources?

Real examples NFR Assessment



Continuous NFRs Assessment in parallel to developing and infrastructure's configuration allowed to take an initial Baseline, analyze metrics, size the infrastructure and configure scalation rules

Conclusions

