

Títol: Eina de suport per a anàlisi de dominis

Volum: 1/1

Alumne: Meseguer Marcobal, Fernando

Director/Ponent: Franch Gutiérrez, Xavier

Departament: L.S.I.

Data:

DADES DEL PROJECTE

Títol del Projecte: Eina de suport per a anàlisi de dominis.

Nom de l'estudiant: Meseguer Marcobal, Fernando.

Titulació: Enginyeria Informàtica.

Crèdits: 37.5

Director/Ponent: Franch Gutiérrez, Xavier.

Departament: L.S.I.

MEMBRES DEL TRIBUNAL (*nom i signatura*)

President: Quer Bosor, Carme.

Vocal: Alvarez Martínez, Carlos.

Secretari: Franch Gutiérrez, Xavier.

QUALIFICACIÓ

Qualificació numèrica:

Qualificació descriptiva:

Data:

***Eina de suport per a anàlisi
de dominis.***

Agraiments

Em primer lloc, vull escriure les primeries línies dedicades a totes aquelles persones que han confiat i ajudat en fer possible aquest treball final de carrera:

En primer lloc, donar les gràcies a Xavier Franch i Claudia Ayala, al confiar en mi per realitzar aquet projecte, així com pels seus seguiments, a la vegada que m'ha permès dependre de les seves explicacions.

A Xavier López i Jordi Pradel, per la seva ajuda tècnica desinteressada amb el component reutilitzat de la eina de taxonomies i l'eina l'Hiberante, respectivament.

No voldria oblidar-me de la meva família i tota la gent que sempre m'ha acompanyat durant la carrera, tant de dins de la universitat com amics de la infància, que en tot moment s'han interessat i m'han animat abans i durant aquest projecte.

Gràcies a tots.

Índex de Continguts

1. INTRODUCCIÓ	1
1.1. INTRODUCCIÓ.....	1
1.2. ÀMBIT DEL PROJECTE.....	3
1.2.1 Àmbit del Projecte General.....	3
1.2.2 Àmbit acadèmic	4
1.3. MOTIVACIÓ.....	6
1.4. DESCRIPCIÓ DELS OBJECTIUS.....	7
1.5. ORGANITZACIÓ DE LA MEMÒRIA.....	9
2. ANTECEDENTS	11
2.1. EXPLICACIÓ DEL/S PROBLEMA/ES.....	12
2.2. MÈTODES PUBLICATS	13
2.3. CONCLUSIÓ	15
3. ANÀLISI.....	17
3.1. ANÀLISI DEL PROBLEMA	18
3.2. ANÀLISI DE REQUERIMENTS	22
3.2.1 <i>Requeriments Generals</i>	22
3.2.2 <i>Requeriments Específics</i>	24
3.2.3 <i>Conclusió</i>	29

4. ESPECIFICACIÓ	31
4.1. MODEL DE CASOS D'ÚS	32
4.1.1 Casos d'ús	32
4.2. MODEL CONCEPTUAL	45
4.2.1 Fragments del Model Conceptual	47
4.3. MODEL DE COMPORTAMENT	52
4.3.1 Diagrames de Seqüència del sistema	52
4.3.2 Contractes de les operacions	62
5. DISSENY	73
5.1. ARQUITECTURA DEL SISTEMA	74
5.1.1 Arquitectura de l'aplicació	74
5.1.2 Estratègia de Persistència	76
5.1.3 Tecnologia emprada	77
5.2. MODEL CONCEPTUAL NORMALITZAT	78
5.2.1 Fragments del Model Conceptual Normalitzat	80
5.3. ASSIGNACIÓ DE RESPONSABILITATS A CAPES	83
Exemple 1: Cas d'ús Alta Participant	84
Exemple 2: Cas d'ús Alta Font d'Informació	85
5.4. ESPECIFICACIÓ DE CADA CAPA	87
Exemple 1: Cas d'ús: Alta Participant	87
Exemple 2: Cas d'ús: Alta Font d'Informació	88
5.5. DIAGRAMES DE SEQÜÈNCIA	89
Exemple 1: Cas d'ús: Alta Participant	90
5.6. DISSENY DE LA INTERFÍCIE	92
5.6.1 Disseny Extern	92
5.6.2 Disseny Intern	93
5.7. STORY BOARD	94
5.7.1 Mapes Navegacionals	94
5.8. DISSENY DE LA BD	102
5.7.1 Traductor de Dades	103
6. IMPLEMENTACIÓ I PROVES	105
6.1. ASPECTES GENERALS DE LA IMPLEMENTACIÓ	106
6.1.1 Eines i Hardware utilitzats	106
6.1.2 Aspectes específics de la implementació ha destacar	107
6.2. PROVES	111
6.2.1 Interacció amb pantalles	111
6.2.2 Visualització de pantalles	111
6.2.3 Entrada de dades	112
6.2.4 Proves del sistema	112

7. PLANIFICACIÓ DEL PROJECTE	113
7.1. MÈTODE RUP	114
7.2. PLANIFICACIÓ DE TREBALL	115
7.2.1 Planificació de cada fase RUP	116
8. ESTUDI ECONÒMIC	121
8.1 CÀLCUL DE COSTOS	122
8.1.1 Recursos humans	122
8.1.2 Altres	122
8.1.3 Cost final	122
9. CONCLUSIONS.....	123
10. BIBLIOGRAFIA	127
10.1 LLIBRES	127
10.2 MATERIAL DOCENT	128
10.3 ARTICLES	128
10.4 PÀGINES WEB	129

1. Introducció

1.1. Introducció

En els temps en que ens trobem la informàtica s'estén per tot el món a una velocitat vertiginosa, aquesta cada cop es troba més present tant en els llocs de treball com en les llars, convertint-se, quasi bé, en una eina de primera necessitat; provocant una major demanda de nous productes software per a poder satisfer les necessitats que deriven d'aquest fenomen, cada cop més accentuat, que es l'expansió de la informàtica. Actualment es desenvolupen nombroses quantitats, gairebé incalculables, de noves aplicacions software, sent les noves versions cada cop més freqüents. D'aquí que ens centrem en la construcció dels productes software.

La construcció de sistemes basats en paquets ja solucionats [Carney-Long2000], [Meyers-Oberndorf2002] esta creixent cada cop més i més, i ha arribat a ser una estratègia necessària per al desenvolupament de diferents àrees d'aplicació. Aquesta tendència ve provocada per un increment en l'adopció de tecnologies de desenvolupament de software basat en components adquirits per diferents proveïdors, i la continua creació de nous

canals de marketing que facin de pont entre proveïdors i consumidors de aquest productes. El principal benefici d'aquesta tecnologia es la reducció del cost i el temps de desenvolupament mentre es manté la qualitat.

Aquest components se denominen Off-The Shelf (OTS) i estan caracteritzats principalment per 2 tipus: COTS (Commercial-Off-The-Shelf), que són distribuïts a algun cost per companyies comercials, que generalment també proveïxen suport; i OSS (Open-Source-Software) proveïts per comunitats d'open source, generalment sense cost ni promesa de suport però amb certes obligacions respecte a les llicències.

Així, un component OTS es defineix com: “un producte de software que està disponible públicament a algun cost o amb certes obligacions respecte a les llicències, i altres projectes de software poden reusar i integrar dins dels seus propis productes.

La selecció de components OTS és una de les activitats més crítiques i esta composta bàsicament per dos activitats: la recerca de candidats i l'avaluació d'aquests respecte als requeriments del sistema. Fins ara, la majoria dels esforços de la comunitat de l'Enginyeria del Software s'han centrat en els aspectes de la tècnica d'avaluació, a la vegada que, deixen de costat els mecanismes per trobar els components adequats en el creixent mercat de components; sobre tot, tenint en consideració l'enorme quantitat d'informació no estructurada que hi ha en el mercat, indispensable per avaluar i caracteritzar els components. Per un altra banda, hi ha proveïdors de paquets que no proporcionen la classe d'informació estructurada que permetria una recerca automatitzada o al menys assistida en l'execució. Per un altra, no es realista pensar que la situació canviarà en el futur, sobre tot, per exemple, quan es tracta d'informació relacionada amb paquets com ERP, CRM o CMS; on la situació es agreujada pel fet de que la informació tendeix a destacar les fortalezes, i a ocultar febleses dels paquets llicenciats.

Així, suposat que la reutilització es un objectiu crucial, es demana una estratègia eficaç per la recerca de paquets software que confiarà en un anàlisi

sòlid del domini per poder dissenyar infraestructures reutilitzables del software. Es fa identificant els elements bàsics del domini, organització i comprensió de les relacions entre els elements, i representat aquest enteniment d'una manera útil. No obstant això, en el camp de la selecció de paquets software no hi ha estudis de com determinar i avaluar les fonts d'informació, cosa en que es deuria de basar aquest domini. La comunitat de l'Enginyeria del Software a comunicat que aquesta es una important tasca oberta [Cechich 2006].

En aquest punt, surt el repte d'intentar classificar i caracteritzar les diferents fonts d'informació que proporcionen informació rellevant sobre el mercat de components. Aquest repte és important per al suport a l'anàlisi de domini requerit per crear una infraestructura de reus eficient, per poder facilitar, en un futur, l'elecció dels producte software que més s'adapti a les característiques.

1.2. Àmbit del projecte

En aquest apartat es situa el projecte dins de la seva temàtica general, relaciona aquest amb el pla d'estudis de E.I. Per altra banda, la relació d'aquest projecte dintre del projecte general (UPIC) d'on parteix aquest treball fi de carrera.

1.2.1 Àmbit del Projecte General

Aquest projecte es situa dins d'un projecte que estan portant a terme per un grup d'investigadors del *Departament de Llenguatges i Sistemes Informàtics de la UPC (LSI)*, més concretament en un projecte anomenat UPIC, que mira de determinar una estratègia per la selecció, anàlisi i estructuració de la informació de components.

El grup busca una estratègia eficaç, per lo qual estan treballant en el desenvolupament d'un mètode anomenat GOTHIC, per poder construir una infraestructura de la reutilització que es pugi fer servir en processos de la recerca de OTS, amb l'ajuda de taxonomies.

Fins al moment, aquest grup tractava la informació, dit d'alguna manera, manualment, i arribat el moment en que la informació es tal que costa treballar-hi. Es aquí on entra en joc aquest treball final de carrera, que mira de desenvolupar una eina que permeti la classificació i organització de la informació relacionada amb components OTS i així ajudar en el posterior anàlisi del domini.

1.2.2 Àmbit acadèmic

En el pla d'estudis de Enginyeria Informàtica de la U.P.C, aquest projecte es troba en el Departament de Llenguatges i Sistemes Informàtics, en concret, en l'àrea de l'Enginyeria del Software. Esta directament relacionat amb els coneixements impartits en diferents assignatures del pla docent, principalment amb l'Enginyeria del Software I i II, PESBD, Programació, Estructures de Dades i assignatures relacionades amb les Bases de Dades.

Tot aquest conjunt d'assignatures tenen com a objectiu final determinar quins són els problemes que planteja el projecte i, conseqüentment, quines són les tècniques per a solucionar-los.

Els coneixements adquirits en les assignatures de Enginyeria del Software s'utilitzaran per a analitzar, dissenyar una solució per a la implementació de l'eina a realitzar.

Les assignatures de Programació i Estructures de dades permetran el anàlisi i disseny de les estructures que tenen que seguir les classes i els mètodes que constitueixen l'aplicació en que es basa aquest projecte.

Les assignatures relacionades amb les Bases de Dades ens ajuden a dissenyar i crear la BBDD que mantindrà la persistència de la informació.

Per tot ell, per delimitar l'àmbit del estudi en el present projecte deuríem atendre als coneixements adquirits en les classes teòrics i, sobre tot, a les pràctiques, com són les realitzades amb els projectes de Programació, de Enginyeria de Software i Bases de Dades.

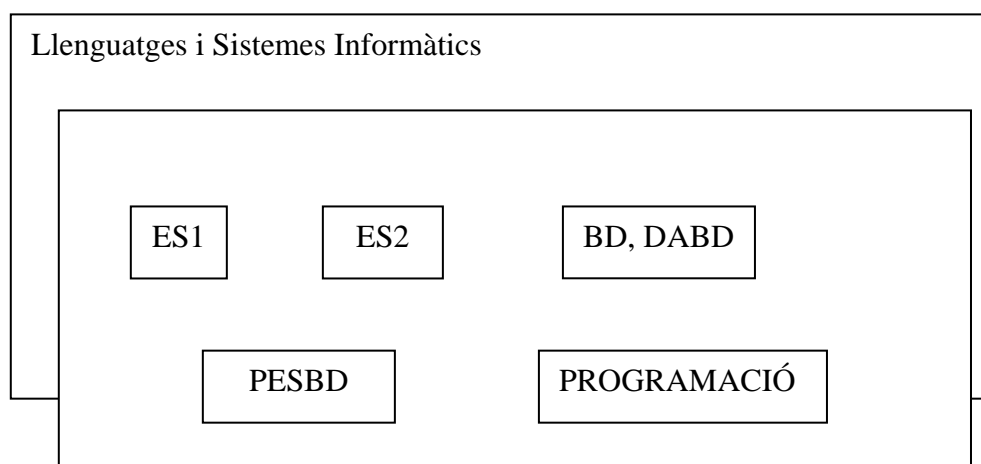


Fig. 1: Relació d'assignatures, departaments i àrees del Projecte.

1.3. **Motivació**

Son varies les motivacions que ens porten a realitzar aquest projecte, un conjunt de les quals es podrien classificar com acadèmiques i les altres com a personals.

A efectes acadèmics l'obtenció d'un producte Software com a eina per a facilitar l'anàlisi de dominis. Els punts motivadors més importants són:

- Principalment, proporcionar una aplicació d'escriptori que faciliti l'anàlisi de dominis. Per poder fer l'extracció i estructuració de la informació dels dominis dels productes software.
- Obtenir una aplicació de codi lliure i obert, feta a partir de tecnologies gratuïtes i potents.

A efectes personals la realització d'un projecte d'aquestes característiques, tan properes a un projecte real, hem reportarà una sèrie de beneficis que completaran la formació que he adquirit al llarg d'aquest cinc anys. Caldria esmentar:

- Implicació en un projecte de recerca a temps complet durant quatre mesos.
- Pas per totes les etapes de producció d'un sistema software, obtenint una visió més propera del procés de creació i manteniment d'un producte software.
- Incrementar la meva fluïdesa com a programador.
- Tenir la responsabilitat de portar endavant un projecte i ajustar-me a una planificació molt rigorosa en tractar-se d'un projecte final de carrera.
- Planificar reunions amb els clients i saber interpretar les seves necessitats.

Desenvolupar una presentació pública del resultat final i saber defendre-la amb èxit.

1.4. Descripció dels Objectius

L'objectiu principal d'aquest PFC es estudiar, dissenyar e implementar una eina per ajudar en l'extracció i estructuració dels diferents components software que hi ha al mercat

Per poder completar aquest objectiu, surt un conjunt d'objectius com a requeriments previs.

- 1 Recerca i estudi de fonts d'informació. Realització d'una recerca de informació d'un cert nombre de components software pels diferents àmbits. Realitzant un posterior estudi per extreure l'esquelet dels diferents components que hi ha al mercat.
 - 1.1 Determinar la informació important, la qual es voldrà guardar dels diferents productes, per poder extreure l'estructura comú.
 - 1.2 Estudiar i determinar com classificar els diferents proveïdors de les fonts.
 - 1.3 Estudiar i classificar les diferents fonts d'informació i la relació entre elles.
 - 1.4 Definició d'atributs: Extreure i establir els diferents tipus d'informació i proveïdors.
 - 1.5 Així com les respectives valoracions de les fonts i dels proveïdors.

- 2 Anàlisi del problema que se'ns planteja a partir de la classificació de les fonts i de les necessitats de l'usuari. Determinant els requeriments generals que ha de satisfer el sistema, saber les finalitats de la aplicació, així com es desitja treballar i mantenir la informació. Necessitat de establir comunicació amb un altra base de dades, ja que aquesta té informació sobre taxonomies, les quals serà informació per classificar la informació dintre del domini.

- 3 Especificació del sistema, “Què ha de fer el sistema?”: determinant quins són els requeriments accessibles, així com dels no funcionals, definint el model de dades i de processos. Així com també especificar la interacció amb l'usuari.

- 4 Dissenyar del producte software. A partir de l'especificació del sistema, dividir el sistema en parts i establir les relacions entre elles i el “què farà” cadascuna d'elles, identificant solucions tecnològiques.

- 5 Implementació de l'eina. Programació de l'aplicació dissenyada utilitzant les eines establertes: llenguatge de programació, Sistema de gestió de la base de dades (SGBD), etc. realitzant, si es necessari, ajustaments en el disseny.

- 6 Comprovació del producte software creat. Comprovar que es el producte desitjat, cerciorant-ne que funciona correctament.

1.5. Organització de la memòria

En aquest apartat es fa una breu descripció del contingut de cada un dels capítols de la memòria, per tracta de que el lector tingui una visió global de la seva estructura. La present memòria esta dividida en 10 capítols sense tenir en compte l'índex, el títol i la secció d'agraïments.

- ❖ El primer capítol es presenta una introducció del projecte mencionant l'àmbit del projecte, per una banda l'enfocament general i per l'altre en l'entorn acadèmic, i la motivació que ens porta a realitzar aquest projecte. A la vegada que també s'enumeren els objectius que busca el projecte i aqueta part on es mostra l'organització de la memòria.
- ❖ En el segon capítol es troben els antecedents del tema que tracta el projecte, és a dir, el punt de partida del mateix, situant el projecte dins d'un context més general i determinar-ne l'àmbit d'aplicació.
- ❖ El tercer capítol tracta l'abordament del problema, analitzant els requeriments del projecte, tant els generals com els específics.
- ❖ En el capítol quart es treballa l'especificació, on es defineixen els artefactes de casos d'ús, esquema conceptual i de comportament (diagrames de seqüència i contractes de les operacions).
- ❖ El cinquè, presenta el disseny, descrivint l'arquitectura general del sistema, a la vegada que les etapes inicials del disseny (normalització del model conceptual i contractes, assignació de responsabilitats a les capes de l'aplicació, especificació de cada una de les capes, amb els diagrames de seqüència de les capes, a la vegada que es dona el disseny de la interfície i Story Board, finalment es presenta el disseny de la Base de Dades.

- ❖ En sisè lloc s'explica aspectes remarcables de la implementació, així com aspectes, generals i específics, rellevants com les eines utilitzades. Tant mateix, es detallen les proves realitzades.
- ❖ En el setè capítol s'indica la planificació seguida pel projecte, a la vegada que les desviacions que a sofert aquest.
- ❖ El capítol vuitè presenta el estudi econòmic realitzat sobre aquest treball final de carrera.
- ❖ El penúltim capítol és la conclusió on es formularan les conclusions personals, explicant els objectius arribats pel projecte, detallant quines millores o projectes finals de carrera poden sorgir d'aquest treball.
- ❖ A més, en aquesta memòria s'adjunten dos annexes. En el primer d'ells es defineixen les referències bibliogràfiques de llibres, articles i documentació en línia consultats per poder realitzar aquest projecte. L'altre esta compost per un manual d'instal·lació de l'aplicació desenvolupada.

2. Antecedents

En aquest apartat es realitza un anàlisi del treball que hi ha del tema amb que es relaciona el projecte que ens ocupa. Mirant de situar aquest dintre d'un context més general i determinar-ne l'àmbit d'aplicació.

De primer s'explica la problemàtica que envolta al projecte, és a dir, quines són les causes que provoquen la necessitat d'aquest projecte. Posteriorment, es presenta els mètodes publicats amb els quals contem per al inici d'aquest projecte.

2.1. *Explicació del/s problema/es*

En el mercat hi ha molts productes software, i cada cop més. Depenent de la finalitat que es demana, cada un dels productes presenta unes coses beneficioses, però també d'altres no tant, respecte a la resta de productes.

Avui en dia, la construcció d'un sistema es basa amb components ja solucionats i a la vegada si es possible gratuïts, comunament denominats Off-the-Shelf (OTS). Els beneficis d'utilització d'OTS són principalment la reducció de costos i de temps empleat amb el desenvolupament, mantenint la qualitat. Malgrat això, la selecció de components per integrar aquests sistemes es un problema.

Quan necessitem un software per a realitzar una determinada tasca, seria ideal que tinguéssim algú a qui preguntar-li, saber quines opcions tenim i quina es la que més ens convé, segons les nostres necessitats i capacitats.

La selecció de components candidats no es una tasca fàcil, especialment en el cas de components OTS. Aquesta recerca té que fer front a una sèrie de característiques desafidores:

- La creixent quantitat de OTS en el mercat.
- Els ràpids canvis en els OTS.
- La Dependències entre components.
- Tipus de descripcions disponibles per als components: no es proporciona una classe d'informació estructurada que permeti una recerca automatitzada o al menys assistida en l'execució.

En aquest context, el grup GESSI ha proposat una metodologia anomenada GOTHIC (Goal-Oriented and reuse Infraestructure Construction) [Ayala-Franch2006] amb la finalitat d'afrontar els desafiaments que imposa la selecció de components OTS.

2.2. Mètodes publicats

Aquest projecte de final de carrera s'emmarca dins del mètode GOTHIC (Goal-Oriented Taxonomy and reuse Infrastructure Construction), proposat pel grup GESSI en el mercat del projecte UPIC (com es menciona en la secció 2.1, explicació del problema). El mètode GOTHIC intenta oferir una solució per resoldre els problemes detectats en els processos de selecció de components OTS en diferents dominis d'aplicació. Així doncs, aquest està compost per 7 activitats interrelacionades:

El mètode GOTHIC es troba dividit en 7 activitats:

1. Exploració de les fonts d'informació.
2. Anàlisi del domini del mercat dels COTS.
3. Identificació, refinament i declaració de objectius.
4. Establiment de dependències.
5. Estructuració de la taxonomia.
6. Validació de la taxonomia.
7. Gerència de la base de coneixements.

El model conceptual està donat per:

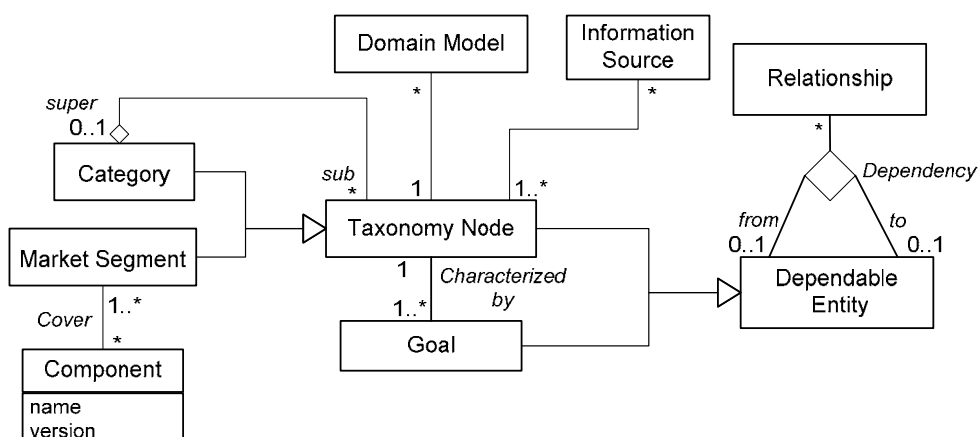


Fig. 2: Model Conceptual del GOTHIC.

Específicament, aquest treball de final de carrera està destinat a donar suport a l'activitat n^o1: "Exploració de Fonts d'Informació", així aquest procés d'exploració de les fonts d'Informació estaria dividida en tres parts: Identificació

de fonts d'informació; Anàlisi de les fonts (determinar els criteris rellevants); Priorització de les fonts, per la classificació en la taxonomia.

La taula 3 resumeix les fonts d'informació comunament útils per realitzar l'anàlisi de dominis relacionats amb components OTS. Es evident que el volum d'informació a considerar és críticament gran. Fins al moment, aquest grup tractava la informació, dit d'alguna manera, manualment, i arribat el moment en que la informació es tal que consta treballar-hi en el posterior anàlisi del domini. A causa d'això, s'ha considerat prioritari desenvolupar un programa adequat d'adquisició de coneixements la base del qual es l'anàlisi de les fonts d'informació.

Information Source		Information Type	Language	Examples	Utility	
Existing Taxonomies and Ontologies		Classifications; Categories; Glossaries	Natural Language (NL); Tree-like diagrams	Gartner, IDC, eCOTS, ComponentSource	They help not only for understanding domains and refining goals, but also for getting insights for organising goals.	
Related Standards		Descriptions; Glossaries	NL	IETF, SIP, ITU H.323, ISO 9126-1	They are considered the most confident of the sources, so the high-level goals are based on them.	
Vendors Information		Brochures; Evaluation forms; Benchmarks	NL; Values for attributes	Microsoft	They are helpful to know functionalities, trends and interactions among components	
Independent Reports	Scientific	Academic Events, Journals, Textbooks	Precise and rigorous descriptions	NL; Models; Formulas; Schemas	ICCBSS, ICSE, TSE	The information enclosed in this kind of sources, generally helps to understand domains and refining goals into sub goals.
	Dissemination	Magazines, Forums and Websites	Descriptions and tips for the general public	NL; Schemas; Tables	PCWorld, IEEE Software, COCOTS website,	
	Technical	White Papers, Surveys and Comparatives	Papers, Comparative Tables	NL, Tables; Figures	Gartner, INCOSE, eCOTS	
Oral Information	Interviews		Knowledge; Tips; Practical Info.	NL	ICCBSS panels, SEI courses, Business luncheons	
	Talks, seminars and courses					
Test of Tools and Systems		Test results; User's manuals	Visual data; NL	ICQ, MSN Messenger, CommuniGate		
Real Experiences on the field		Knowledge; Technical reports	Knowledge; NL	Past projects made		
Others		Any	Any			

Fig. 3: Taula Fons d'Informació per al cas RTCS.

Així aquest procés d'exploració de les fonts d'informació estaria dividida en tres parts:

- Identificació de fonts d'informació.
- Anàlisi de les fonts (determinar els criteris rellevants).
- Priorització de les fonts, per la classificació en la taxonomia.

2.3. Conclusió

La necessitat d'organitzar les diferents font d'informació relacionades amb components OTS, es una tasca prioritària per realitzar un anàlisi de domini satisfactori i eficaç. Així doncs, el desenvolupament d'un programa adequat per a l'anàlisi d'aquestes fonts d'informació que permeti la seva avaluació i prioritització d'acord a diferents criteris, és un dels objectius actuals del mètode GOTHIC.

Aquest projecte pretén posar el seu granet d'arena. En aquests moments ens trobaríem en la fase de crear una eina que ens ajudi en l'organització i valoració de les diferents fonts d'informació relacionades amb components OTS, mitjançant la creació una base de dades (o al menys una part) sobre els productes software que hi ha al mercat, tant dels components software com de les corresponents organitzacions o persones que els creen.

Aquest projecte permetrà l'organització i actualització d'informació relacionada amb els components OTS, per donar suport al procés de anàlisi de dominis, informant també al sistema del que busquem mitjançant un qüestionari.

3. Anàlisi

En aquest capítol es realitza l'anàlisi del problema a solucionar, amb l'objectiu de determinar i especificant els requeriments generals com específics, per poder contestar a la qüestió *quin sistema cal construir?*.

Dit d'un altra manera, s'analitza les condicions o capacitats necessitades per l'usuari, per definir el conjunt de propietats i de característiques del sistema (producte) que podrien satisfer aquestes necessitats (explícites com implícites) o objectius a assolir, triant el sistema més adient.

La qualitat del software es el grau amb el qual un sistema o component compleix els requeriments específics i les necessitats o expectatives del client. Per lo que aquest anàlisi és important, ja que a més qualitat, es passa de bon control de la qualitat (detectant errors) a permetre tenir una garantia de qualitat (prevenint errors) i en el temps una milloria contínua.

3.1. Anàlisi del problema

Com s'ha esmentat anteriorment, en el mercat actual hi ha molts productes software, aquests components pertanyen a dominis, a títol d'exemple podríem enumerar Real Time, Eines d'Enginyeria de Requeriments. L'anàlisi de dominis (*domain analysis*) extreu informació rellevant de dominis de productes software.

Per a la recerca i estudi, utilitzem com a cas d'estudi el domini de Temps Real. Es realitza una recerca pels diferents àmbits del domini, com son: Taxonomies i Ontologies, estàndards del domini, reports independents, tests d'eines, experiències personals, informació oral, ... variant els diferents tipus de proveïdors, així com buscant també valoracions sobre els components i els seus creadors.

De entre les característiques a tenir en consideració per classificar un component software, es troben, la versió d'aquest, quina persona/s o organització/ns tenen l'autoria, la disponibilitat, traduccions en que es troba el component, el cost d'adquisició, ...

A continuació, es presenta un exemple de l'artefacte que ajuda a classificar els diferents productes software, l'artefacte esta compostat per un conjunt de qüestions. En la columna de l'esquerra es troben els atributs que determinen les característiques que els diferencien, en la dreta, es troben les qüestions conjuntament amb les respostes de l'exemple en negreta.

ISO-9126-1

Name	Nom? Iso-9126-1
Version	Versió del producte? 1^oEd
Reliability	Valoració? high
Cost	Preu del producte? (en euros) 68,65 for document
Description	Per a que es usat? Tema principal relacionat Paraules Clau Standard about evaluation of software. Part 1: Quality Model

Location	Localització de la font? (direcció web,llibre) www.iso.org
Language	Idiomes en que es troba publicat English
Organization	Participant responsable del producte International Organization for Standarizations
Organization Type	Tipus de participant (existeix una enumeració: Academy, Standard, Company,...): <ul style="list-style-type: none"> - Es un centre acadèmic o de investigació? - Es companyia amb ànim de lucre? - L'organització es dedica principalment al desenvolupament o manteniment d'estandards? Standard
Information Source Type	La informació continguda representa (Existeix una enumeració:Hierarchy, Standard, VendorInfo,IndependentReport, OralInfo, TestOfTool, Experiencie, Other): <ul style="list-style-type: none"> - Jerarquies, Taxonomies o Ontologies? - Es considerat un estàndard en el àrea? - Reflexa informació o opinions d'un venedor? <ul style="list-style-type: none"> o Es un Manual d'Usuari? o Altres (Especificar). - Es considerat un Report independent? <ul style="list-style-type: none"> o Es d'àmbit científic (events acadèmics, journals, textbooks). o Es d'àmbit divulgatiu (relacionat amb revistes, websities o forums). o Es d'àmbit tècnic(relacionat amb comparatives, whitepapers o surveys). - Es informació oral (entrevistes, seminaris, cursos...)? - Test de eines y/o sistemes (resultat de proves específiques). - Experiències (coneixements sobre projectes realitzats anteriorment) - Altres (Especificar). Standard
Date	Data del producte 2001
Scope Covered	Quines són les àrees que cobreix? Exemple: Classifies software quality in a structured set of factors as follows: Functionality, Reliability, Usability, Efficiency, Maintainability and Portability.
Intended Audience	A qui va dirigit la informació? <ul style="list-style-type: none"> o Public en general? o Experts en el tema? Experts
Last Revised	Última revisió? 2001
Description Type	Tipus de representació? (NL, models, diagrames,...) NL (Natural Language)

Format	Quins són els formats del document? (pdf, doc, html...) Pdf
References	Referència a altres fonts? SQuaRE, ISO 25000:2005
Depends-on	Depèn d'altres fonts? No
Related-to	Està relacionat amb altres fonts? ISO 9000, ISO 9126-2, ISO 9126-3, ISO 9126-

En funció del tipus d'informació, per cada component software es interessant determinar un conjunt d'atributs, alguns cops diferents. En la següent pàgina es presenta una taula, relacionant el tipus d'informació amb el conjunt de propietats o atributs. Es marca amb una "X" els atributs que representen informació rellevant dins dels components software. Caldria destacar l'atribut *others*, el qual podrà tindre un valor dintre d'un conjunt concret en funció del tipus d'informació: serà *userManual* o *others* si el tipus és *vendor Info*, però sols podrà ser *scientific*, *divulgation*, *technical* si el tipus és *independent Report*, en canvi, si el tipus és un dels altres, aquest atribut no serà classificatori.

Aquesta relació d'atributs amb els tipus ajudarà en el posterior desenvolupament de l'eina software.

Les fonts d'informació es troben relacionades amb les taxonomies, les quals estan àmpliament catalogades en una base de dades ja existent en el Departament de Llenguatges i Sistemes Informàtics de la Universitat Politècnica de Catalunya. Per evitar produir incoherències, seria interessant que la eina que es pretén crear en aquest projecte es comuniqui amb la dita BBDD per obtenir les taxonomies amb les quals es relacionarà els components software

Es podria treballar en el plantejament que la BBDD, que persistirà les dades de la eina, es trobes en un lloc extern a l'aplicació, és a dir, que l'aplicació es comunicarà a distància. Permeten a l'usuari, per exemple, treballar des de diferents llocs.

TABLE OF ATTRIBUTES

			{userManual, others}	{scientific, divulgation, technical}				
Others								
Relation	X	X	X	X	X	X	X	X
Dependency	X	X	X	X	X	X	X	X
References	X	X	X	X	X	X	X	X
Valoration*				X	X	X	X	X
Last Revised		X						X
Versión		X						X
Date	X	X	X	X	X	X	X	X
Name	X	X	X	X	X	X	X	X
Intended Audience	X	X	X	X	X	X	X	X
Scope Covered	X	X	X	X	X	X	X	X
Reliability	X	X	X	X	X	X	X	X
Organization	X	X	X	X	X	X	X	X
Language	X	X	X	X	X	X	X	X
Cost	X	X		X	X	X	X	X
<i>Attributes</i>	Heriarchy	Standard	Vendor Info	Independent Report	Oral Info	Test of Tools	Experiences	Others
<i>Information Source Type</i>								

* Valoration → {reliability, description}.

Fig. 4: Taula relaciona els atributs amb els tipus de Informació.

3.2. Anàlisi de requeriments

Aquest capítol presenta les necessitats dels clients i usuaris finals capturades en una detallada especificació de requeriments, tant funcionals com no funcionals.

3.2.1 Requeriments Generals

El desenvolupament de l'aplicació es veu limitada per la restricció de utilitzar únicament software lliure de llicència pública i/o no lliure però amb la excepció de ser llicències comprades per la Universitat.

De cara a l'usuari estàndard no hi ha cap restricció d'instal·lació ni d'execució.

Les plataformes seran on es trobi instal·lada una de les diferents versions del sistema operatiu UNIX, com també de Microsoft Windows. A més, existeixen quantitats de recursos de lliure distribució per la xarxa d'Internet (sobre tot, en les universitats) per poder satisfer els requeriments de software d'aquest projecte.

3.2.1.1 Requeriments de Hardware

En un principi i tal com es menciona, el projecte es podrà instal·lar i executar en plataformes que suportin el sistema operatiu UNIX o adaptació. Per a us personal es pretén utilitzar un PC convencional amb el sistema operatiu LINUX, mentre que per a un us docent o corporatiu es recomana un host amb el sistema operatiu UNIX, amb terminals gràfiques de X-Windows.

Hi ha aspectes del hardware que condicionen la velocitat i quantitat de dades a tractar en les execucions de l'aplicació:

- El Processador: Segons el tipus de processador i velocitat que té.

- Memòria volàtil: A més memòria es donarà més marge al sistema operatiu per poder carregar totes les coses que pugui necessitar i, així, tenir que fer els mínims accessos possibles a disc mitjançant el ús de buffer, ja que la memòria de disc es molt més lenta que la memòria volàtil.
- Memòria en disc: Avui en dia, aquest no es un punt molt problemàtic, però es recomanable tenir un bon espai de disc per memòria de swap per als sistemes operatius i amb marge de maniobra, espai per l'aplicació i les dades.

3.2.1.2 Requeriments de Software

Els requeriments de software per poder instal·lar i executar l'aplicació són: el sistema operatiu, el llenguatge de programació, per poder crear els executables en la plataforma que farem servir; a més de l'entorn de treball, així com, les versions adequades de les diferents llibreries per la plataforma.

- Sistema Operatiu; UNIX o derivat, per exemple, LINUX per PC.
- Llenguatge de programació: Java.
- Entorn de treball: X-Windows.
- Llibreries:

3.2.2 Requeriments Específics

3.2.2.1 Requeriments funcionals

L'especificació de requeriments funcionals que es detallen a continuació en un grau precís, indicant subfuncionalitats, i ordenat.

1. Participants: Permetre al usuari mantenir una base de dades del les persones o organitzacions que escriuen i/o realitzen valoracions, sobre fonts d'informació i altres participants. Realitzant altes, baixes, consultes, modificacions... com permetre també llistats dels participants que existeixen en la base de dades.
2. Fonts d'Informació: Facilitar que l'usuari pugui treballar les fonts d'informació dels diferents components software, realitzant altes, baixes, consultes, modificacions, i llistats d'aquestes sobre una base de dades. Permeten a l'usuari veure les valoracions que rep una font d'informació.
3. Les taxonomies, com ja s'ha comentat anteriorment, caldrà obtenir-les d'un altra base de dades,
4. Facilitar a l'usuari el treball mitjançant una interfície gràfica intuïtiva.
5. Mantenir de manera persistent tota la informació (fonts d'informació, participants, valoracions, ...) que manipularà l'usuari mitjançant la interfície gràfica, , en una base de dades.

3.2.2.2 Requeriments no funcionals

Defineixen les qualitats generals que ha de tenir el sistema en realitzar la seva funció. A continuació, es llisten tots els requeriments de qualitat, característiques i subcaracterístiques, que ha de tenir el sistema en realitzar la seva funció .

Funcionalitat:

Correctesa	<p><i>Fa el que volem?</i> El producte final haurà de tenir la capacitat de realitzar amb exactitud les tasques, tal i com es defineixen en les especificacions.</p>
Adequació Precisió	<p>Es deurà de realitzar el conjunt de funcionalitats necessàries per a les diferents parts del problema, i així poder aconseguir els objectius de l'eina software de una manera precisa.</p>
Interoperabilitat	<p><i>Podrem fer-lo interactuar amb un altre sistema?</i> El producte esperat deurà de poder comunicar-se amb altres eines, per obtenir el llistat de taxonomies. Aquesta comunicació es realitzarà mitjançant formats XML.</p>
Seguretat (integritat)	<p><i>És segur?</i> El sistema final tindrà que ser capaç de protegir les dades contra modificacions i accessos no autoritzats.</p>

Fiabilitat: ho fa de manera fiable tot el temps?

Maduresa	Es deu de seguir rigorosament el pla proposat per al desenvolupament del software esperat. Treballant amb pressuposts basats amb estimacions reals.
Robustesa	El sistema tindrà que respondre el mes apropiat possible front a condicions extremes, és a dir, casos fora dels definits en l'especificació. Produint missatges d'error apropiats, evitant events catastròfics. Garantint a l'usuari que esta treballant sense fallides.

Usabilitat:

Facilitat d'aprenentatge i d'operació	<i>Esta dissenyat per se usat?</i> Persones amb diferents formacions i aptituds podran aprendre i fer ús fàcilment de l'eina. Ja que serà una eina intuïtiva. Proveint un fàcil i ràpid accés a les funcionalitats (amb barres d'eines i/o amb accessos amb el botó dret del ratolí).
Comprensió	L'eina té una comprensió proporcional al aprenentatge. No implicarà un re-estudi de l'aplicació, sinó que sols caldrà fer-ho sobre les noves funcionalitats.
Atractivitat	El producte manté uns requisits d'interfície, tindrà que ser visualment agradable treballar amb ell.

Mantenibilitat:

Analitzabilitat	Pensant en un futur treball d'extensió del programa, el codi haurà de ser fàcil de llegir i comprendre.
Canviabilitat (flexibilitat)	<i>És pot canviar?</i> El producte promourà la facilitat de canvis en l'especificació.
Estabilitat	L'eina serà completament estable, garantit que canvis futurs no comportin grans trastorns, promovent
Facilitat de Proves	<i>És pot provar?</i> S'haurà de donar facilitat per preparar procediments de acceptació, especialment dades de prova i procediments per detectar errades, tant del codi existent com de noves funcionalitats, per a que l'aplicació continuï funcionant.

Portabilitat: podré usar-lo en un altra màquina?

Adaptabilitat	El producte final tindrà que adaptar-se a l'entorn en el qual es farà servir. Tant a nivell software com a nivell hardware, en desenvolupament com usant el producte.
Facilitat de substitució	El disseny del software tindrà que permetré una fàcil substitució de part del software, atenent canvis en l'especificació
Instalabilitat	El producte tindrà que ser fàcil d'instal·lar en una màquina que compleix els requeriments. Així com de supervisió.

És important saber que existeixen múltiples interaccions entre els factors de qualitat, apareixen sinèrgies, quan un factor afavoreix un altre, o conflictes, quan un factor perjudica un altre. En el disseny de la eina, ens interessa que els factors de qualitat es corresponguin amb els mencionats anteriorment. Constatar de que dos dels principals requeriments de qualitat d'aquest projecte són la Usabilitat i Mantenibilitat. En canvi, l'eficiència (*s'ajusta al nostre hardware el millor possible?*) és un dels factors menys important alhora de desenvolupar el projecte, aquest és el que més conflictes té amb la resta de factors de qualitat.

3.2.3 Conclusió

En aquest capítol s'ha fet un anàlisi del problema que ens planteja, definint una taula de la relació entre els tipus de fonts d'informació i els atributs que ens ajuden a classificar les fonts d'informació, així com s'ha determinat una plantilla per facilitar la classificació dels components software obtinguts en la recerca realitzada com per a futures recerques. Posteriorment s'han definit els requeriments generals, tant de hardware com de software, del projecte a desenvolupar, així com els requeriments específics.

4. Especificació

Aquest apartat podria ser part del anterior, però s'ha separat per tenir una visió més clara dels passos que s'han anat realitzant en aquest treball final de carrera. En aquest apart, segons les etapes del desenvolupament del software, es mira de respondre a la qüestió: *Què ha de fer el sistema?*. Com a llenguatge de representació s'utilitza el llenguatge UML 2.0 (Unified Modeling Language).

Es presenta els artefactes de l'especificació. Concretament el model de casos d'ús, on s'analitzarà cada cas d'ús individualment. Després es presenta el model conceptual de dades, i l'explicació d'aquest per seccions, per facilitar-ne la seva comprensió per part del lector. Finalment, es mostra el model de comportament, on es presenten els diagrames de seqüència del sistema i els corresponents contractes de les operacions. Els diagrames de seqüència mostren les seqüències d'esdeveniments entre els actors, aquí l'usuari i el sistema, permeten identificar les operacions del sistema. Pel que fa als contractes, descriuen l'efecte de les operacions del sistema.

4.1. Model de casos d'ús

En aquest artefacte de casos d'ús es presenten al client solucions al conjunt d'escenaris que descriuen de quina manera els usuaris interaccionen amb el sistema per assolir totes les necessitats els objectius especificats a l'anàlisi de requeriments. En el moment que els clients fan l'aprovació del model de casos d'ús es pot passar al següent nivell de desenvolupament del producte, ja que quedaran definides a alt nivell totes les funcionalitats que s'hauran d'implementar en l'aplicació software.

4.1.1 Casos d'ús

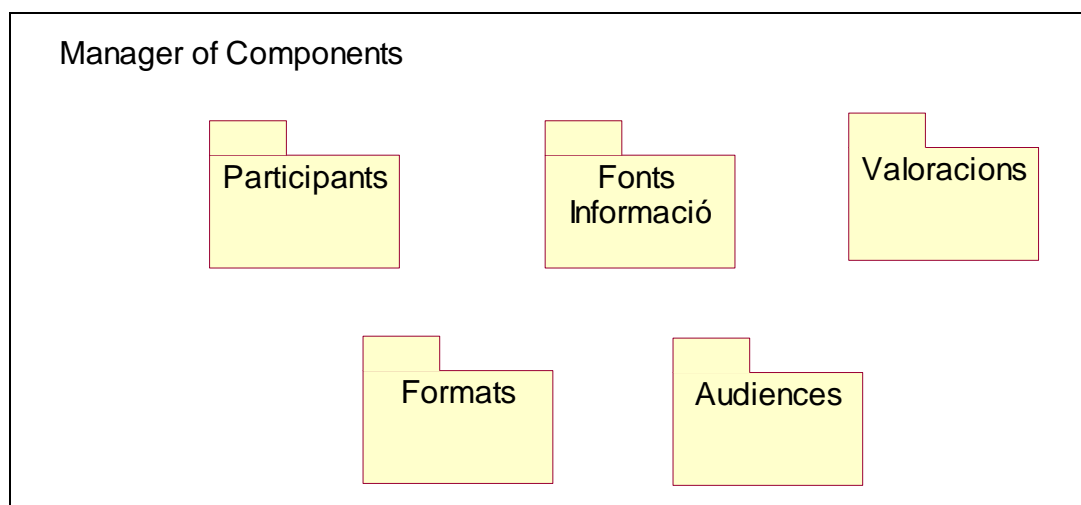


Fig. 5: Paquets de les funcionalitats de l'eina.

En la figura 5 es poden diferenciar cinc grans paquets per representar els casos d'ús. Per cada un dels paquets, de primer es presenta el diagrama de casos d'ús i posteriorment, es detallen cada un d'aquests en format complet i concret.

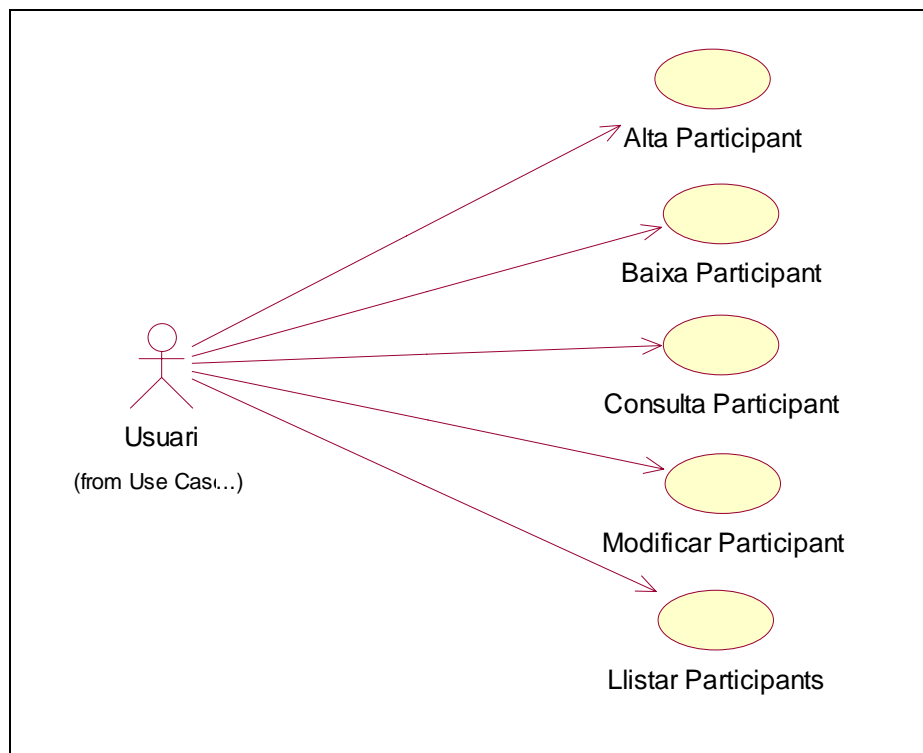
Paquet 1: Participants

Fig. 6: Diagrama casos d'ús del paquet Participants.

Cas d'ús : Alta Participant**Actors** Usuari**Propòsit** Permetre al usuari donar d'alta persones i/o organitzacions.**Resum** El usuari dona d'alta un participant.**Curs Principal**

- 1.-El usuari proporciona al sistema el Name del Participant.
- 2.- U selecciona en el desplegable el reliability.
- 3.- U introdueix la description.
- 4.- U especifica en el opcional que el participant és Person.
(del 1 al 4 en pestanya Info Bàsica)
- 5.-U prema "next".
- 6.-S activa la pestanya de Person.
- 7.- U introdueix el dni, City i email del Person .
- 8.- U prema en afegir un tipus de Person (AddType).
- 9.- U selecciona el un typePeron i un reliability desitjats i prema "OK".
- 10.- U clicka en afegir en llista de Rols.
- 11.- El sistema mostra llista de Organitzacions.
- 12.- U selecciona la organització e introdueix el nom del Rol i prema "OK".
- 13.- U pot donar d'alta evaluate-items i evaluateParticipants
- 14.- U prema "finish" donant per acabat l'alta de Participant.

Extensions

E name : Ja existeix algun participant amb el mateix nom (4)

- 1.- el sistema ho comunica al U i continua.

E el Type: Ja existeix el tipus de Person amb aquest type (9)

- 1.- el sistema ho comunica al U.
- 2.- el sistema dona l'opció d'introduir un altre type o la seva cancel·lació.

E Rol: E Organization amb el nom de Rol indicat. (12)

- 1.- el sistema ho comunica al U i demana un altre rol o la seva cancel·lació.
-

Cas d'ús : Baixa Participant

Actors Usuari

Propòsit Permetre al usuari donar de baixa Participant de la base de dades.

Resum El usuari dona de baixa un participant.

Curs Principal

- 1.-El U selecciona del desplegable el identificador del participant
- 2.-El S el desenregistra

Extensions

E relacions: El participant té relacions amb valoracions o fonts (1)

- 1.-El S pregunta si desitja esborrar-ho en cascada.
- 2.-U indica que Si.
- 3.-El S el desenregistra tot.

No esborrar res: U indica que no vol fer esborrat en cascada(2 de E relacions)

- 1.- Cas d'ús acaba.
-

Cas d'ús : Consulta Participant

Actors Usuari

Propòsit Permetre al usuari consultar dades dels Participants.

Resum El usuari consulta les dades d'un participant.

Curs Principal

- 1.-El U selecciona del desplegable el identificador del participant a consultar.
- 2.-El S mostra les dades d'aquest, així com les corresponents relacions.

Cas d'ús : Modificar Participant

Actors Usuari

Propòsit Permetre al usuari la modificació de les dades dels Participants.

Resum El usuari modificar les dades d'un Participant.

Curs Principal

- 1.-El Usuari selecciona del desplegable el identificador del participant.
- 2.-El Sistema busca i mostra les dades actuals del participant.
- 3.-El U modifica les dades del participant.
- 4.-El U indica al sistema que ja ha acabat prement el botó "Accept".
- 5.-El S guarda les modificacions realitzades al participant.

Extensions

Cancel·lar. El U prema "Cancel". (4)

- 1.-El sistema no guarda els canvis realitzats i dona per acabat el cas d'ús.
-

Cas d'ús : Llistar Participants

Actors Usuari

Propòsit Permetre al usuari veure un llistat de tots els Participants.

Resum El usuari consulta un llistat de tots els Participants.

Curs Principal

- 1.-El usuari sol·licita al sistema un llistat de tots els participants.
- 2.-el sistema mostra un llistat de tots els participants.

Extensions

No E cap Participant. No hi ha cap participant. (1)

- 1.- El sistema mostra una llista buida.

Paquet 2: Fonts d'Informació

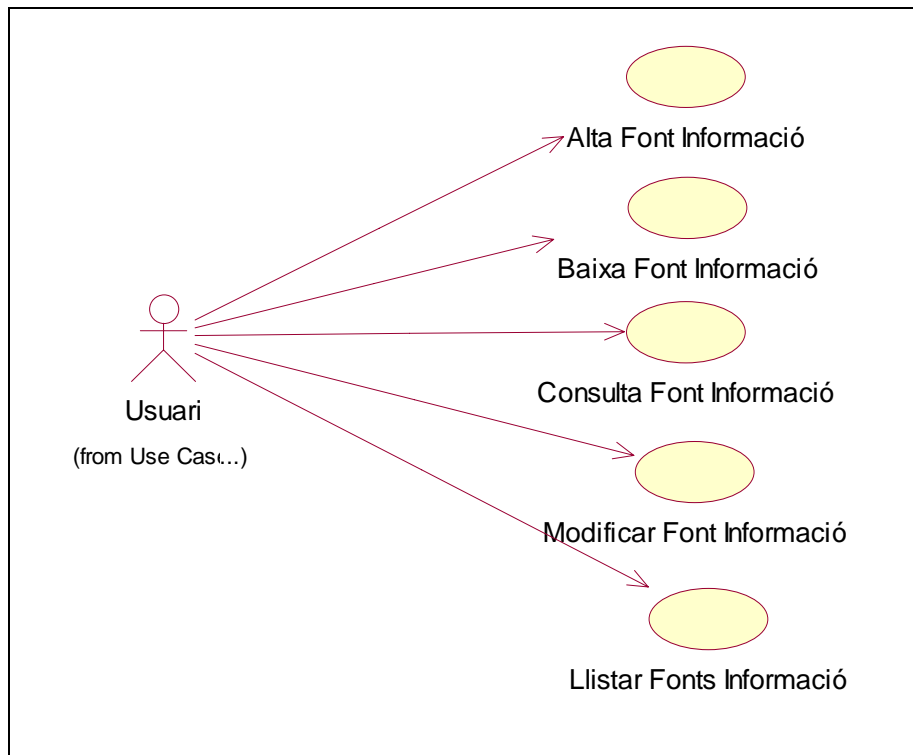


Fig. 7: Diagrama casos d'ús del paquet Fonts d'informació.

Cas d'ús : Alta Font d'Informació

Actors Usuari

Propòsit Permetre al usuari donar d'alta fonts d'informació de participants.

Resum El usuari dona d'alta una font d'informació.

Curs Principal

- 1.-El usuari proporciona al sistema el name de la font.
- 2.-U introdueix la versió de la font.
- 3.- U selecciona en el desplegable el reliability.
- 4.- U introdueix la data, la localització de la font, així com el cost.
- 5.- U introdueix la description, el descriptionType i el Scope Covered.
(del 1 al 5 en pestanya Info Bàsica)
- 6.-U prema "next".
- 7.-S activa la pestanya InfoSourcetype.
- 8.- U Selecciona el reliability del dessorcetype
- 9.-U indica el type of Source en el segon desplegable.
- 10.-El sistema genera el opcional de subtype en funció del tipus elegit per l'U.
- 11.-U indica opcionalment el subTypus.
(en la pestanya InfoSourceType, del 6 al 11)
- 12.-U prema "next".
- 13.-S activa la pestanya de format.
- 14.-U veu en un llistat els formats de la font.
15. U prema "add".

-
- 16.-U introdueix el nou format i prema "accept".
 - 17.-S actualitza el llista dels formats de la font.
(13 i 17 en pestanya format)
 - 18.-S activa la pestanya de Taxonomies.
 - 19.-S mostra el llistat de taxonomies i també un llistat amb les que té la font.
 - 20.-U prema en "add" .
 - 21.-U selecciona una taxonomia i prema "Accept".
 - 22.-S actualitza el llista de les taxonomies de la font.
 - 23.-U prema "next".
 - 24.- S activa la pestanya Languages.
(del 18 al 22 es igual la pestanya Languages)
 - 25.-U prema "next" i S activa la pestanya "Audiences".
 - 26.-S mostra el llistat de audiències que té la font d'informació.
 - 27.-U prema "add".
 - 28.-U introdueix el name i selecciona d'un desplegable el AudienceType i "Ok".
 - 29.-S actualitza el llista dels Audience de la font.
 - 30.-U prema "next" i S activa la pestanya "Participants".
 - 31.-S mostra llistat de "Written By" de la font.
 - 32.-U prema "Afegir".
 - 33.-S mostra llista de participants.
 - 34.-U selecciona del desplegable el participant i prema "ok"
 - 35.-S actualitza el llista dels escriptors de la font.
 - 36.- U prema "next" i S activa la pestanya References.
 - 37.-S mostra el llistat de fonts referenciades a aquella.
 - 38.-U prema afegir en referències.
 - 39.-S mostra llistat de fonts.
 - 40.-U selecciona la font a referenciar i prema "ok".
 - 41.-S refresca la llista de fonts referenciades.
 - 42.-Semblant amb DependesOn i Related To

Extensions

- E name* : Ja existeix alguna font d'informació amb el mateix nom (6)
- 1.- el sistema ho comunica al U i continua.

Cas d'ús : Baixa Font d'Informació

Actors Usuari

Propòsit Permetre al usuari donar de baixa fonts d'informació.

Resum El usuari dona de baixa una font d'informació.

Curs Principal

- 1.-El usuari selecciona del desplegable el identificador de la Font d'informació.
- 2.-El Sistema el desenregistra.

Extensions

- E relacions*: La font té relacions amb altres fonts i/o valoracions (1)
- 1.-El S pregunta si desitja esborrar-ho en cascada.
 - 2.-U indica que Si.
 - 3.-El S ho desenregistra tot.

- No esborrar res*: U indica que no vol fer esborrat en cascada(2 de E relacions)
- 3.- Cas d'ús acaba.

Cas d'ús : Consulta Font d'Informació

Actors Usuari

Propòsit Permetre al usuari consultar les fonts d'informació.

Resum El usuari consulta les dades d'una font d'informació.

Curs Principal

- 1.-El usuari selecciona del desplegable el identificador de la Font d'informació.
 - 2.-El Sistema mostra les dades d'aquesta.
-

Cas d'ús : Modificar Font d'Informació

Actors Usuari

Propòsit Permetre al usuari modificar les dades de les fonts d'informació.

Resum El usuari modifica les dades d'una font d'informació.

Curs Principal

- 1.-El usuari selecciona del desplegable el identificador de la Font d'informació.
- 2.-El Sistema busca i mostra les dades actuals de la font d'informació.
- 3.-El U modifica les dades del que consideri.
- 4.-El U indica al sistema que ja ha acabat prement el botó "Accept".
- 5.-El S guarda les modificacions realitzades a la font.

Extensions

Cancel·lar: El U prema "Cancel.

- 1.- El sistema no guarda els canvis i acaba.
-

Cas d'ús : Llistar Font d'Informació

Actors Usuari

Propòsit Permetre al usuari veure un llistat de totes les fonts d'informació.

Resum El usuari realitzar visualitzar un llistat de les font d'informació.

Curs Principal

- 1.- El usuari sol·licita al sistema un llistat de totes les Font d'informació.
- 2.- El sistema les llista totes.

Extensions

no E cap Font d'informació: No hi ha cap font d'informació a la BBDD.

- 1.- El sistema mostra una llista buida.

Paquet 3: Formats

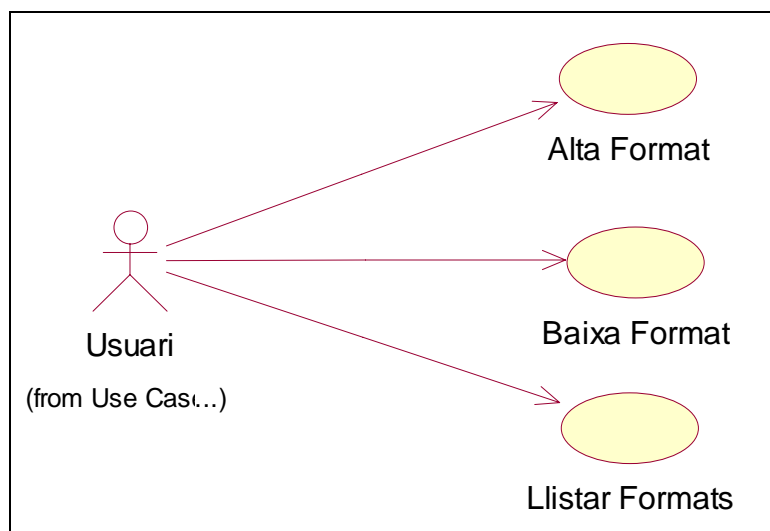


Fig. 8: Diagrama casos d'ús del paquet Formats.

Cas d'ús : Alta Format

Actors Usuari

Propòsit Permetre al usuari donar d'alta nous tipus de formats.

Resum El usuari realitzar l'alta d'un nou format.

Curs Principal

- 1.- El usuari proporciona al sistema el nom del nou format i prema "OK".
- 2.- El sistema l'enregistra.

Extensions

- E el format:* Si ja existeix algun format amb ell mateix identificador. (1)
- 1.- El sistema ho comunica al usuari i permet introduir un altre.

Cas d'ús : Baixa Format

Actors Usuari

Propòsit Permetre al usuari donar de baixa formats existents.

Resum El usuari realitzar la baixa d'un format.

Curs Principal

- 1.- El usuari selecciona del llistat de formats, el que desitja esborrar.
- 2.- El sistema el desenregistra.

Extensions

- El format té relacions:* Si el format seleccionat té relacions amb fonts d'inforamció. (1)
- 1.- El sistema ho comunica al usuari i no l'elimina.

Cas d'ús : Llista Formats

Actors Usuari

Propòsit Permetre al usuari veure un llistat dels formats existents.

Resum El usuari consultar un llistat amb tots els formats.

Curs Principal

- 1.- El usuari sol·licita al sistema un llistat de tots els Formats.
- 2.- El sistema els busca i els mostra tots.

Extensions

No E cap format:: Si no E cap format. (2)

- 1.- El sistema mostra una llista buida.

Paquet 4: Audiencies

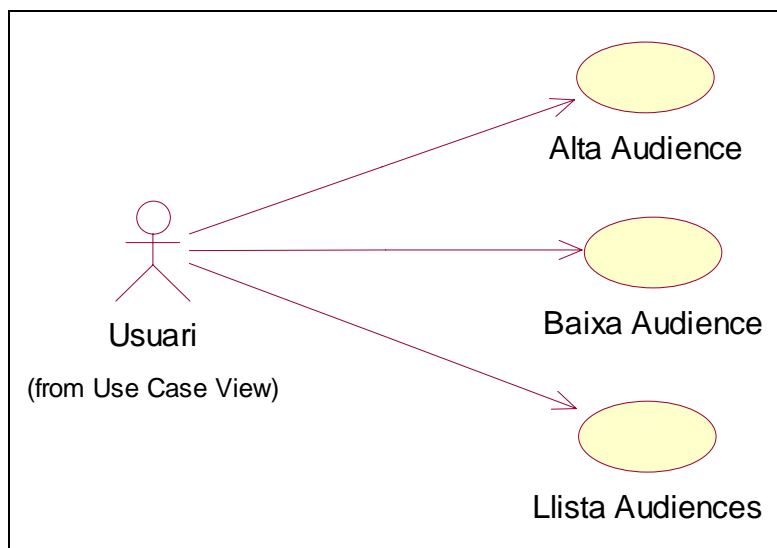


Fig. 9: Diagrama casos d'ús del paquet Audiencies.

Cas d'ús : Alta Audience

Actors Usuari

Propòsit Permetre al usuari donar d'alta nous *Audiencies*.

Resum El usuari realitzar l'alta d'un nou Audience.

Curs Principal

- 1.- El usuari proporciona al sistema un nom per al nou audience.
- 2.- El usuari selecciona del desplegable un type d'*audience* i preme "OK".
- 3.- El sistema l'enregistra.

Extensions

E el Audience: Si ja existeix algun *audience* amb ell mateix identificador. (2)

- 1.- El sistema ho comunica al usuari i permet introduir un altre.

Cas d'ús : Baixa Audience

Actors Usuari

Propòsit Permetre al usuari donar de baixa Audiences existents.

Resum El usuari realitzar la baixa d'un audience.

Curs Principal

- 1.- El usuari selecciona del llistat de Audiences, el que desitja esborrar.
- 2.- El sistema el desenregistra.

Extensions

L'audience esta relacionat. Si l'audience seleccionat té relacions amb fonts d'informació. (1)

- 1.- El sistema ho comunica al usuari i no l'elimina.
-

Cas d'ús : Llista Audiences

Actors Usuari

Propòsit Permetre al usuari veure un llistat de tots els Audiences existents.

Resum El usuari consultar un llistat amb tots els Audiences.

Curs Principal

- 1.- El usuari sol·licita al sistema un llistat de tots els Audiences.
- 2.- El sistema els busca i els mostra tots.

Extensions

No E cap Audience.: Si no E cap audience. (2)

- 1.- El sistema mostra una llista buida.

Paquet 5: Valoracions

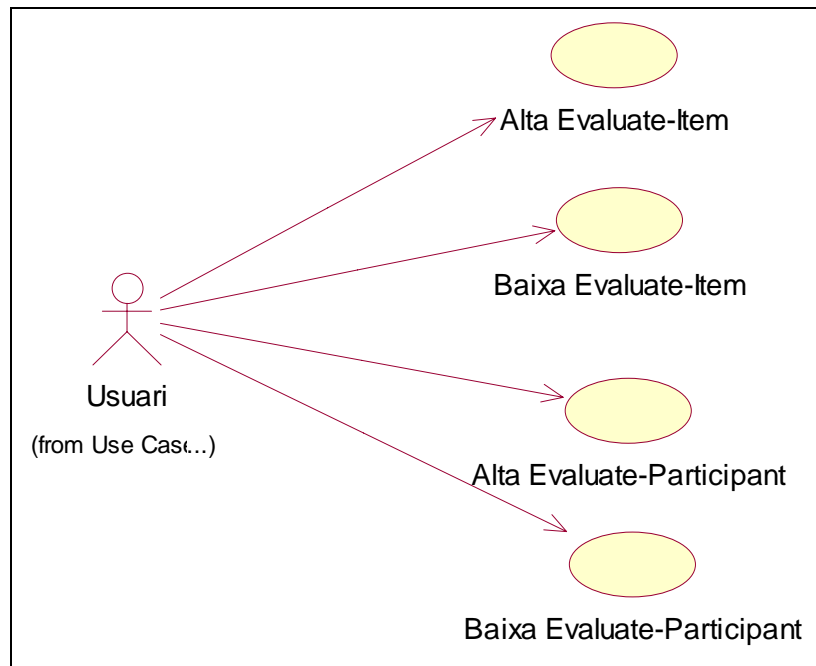


Fig. 10: Diagrama casos d'ús del paquet Valoracions.

Cas d'ús : Alta Evaluate-Ítem

Actors Usuari

Propòsit Permetre al usuari donar d'alta avaluacions sobre fonts d'informació, que no hagin segut realitzades per ell.

Resum El usuari realitzar l'alta d'una valoració sobre una font d'informació

Curs Principal

- 1.- El usuari selecciona es troba en editar (cas d'ús modificar) el participant *avaluador*, o en l'alta el participant que es vol registrar. (el usuari treballa en la pestanya de evaluate-Items)
- 2.-S mostra totes les valoracions que ha fet el participant sobre fonts (no pròpies)
- 3.-El Usuari prema el botó "add".
- 4.-El S prepara un llistat de les fonts que no són del participant.
- 5.-El U selecciona la font sobre la qual desitja fer l'avaluació.
- 6.- El U proporciona una descripció i selecciona el reliability del desplegable.
- 7.-El Sistema enregistra la avaluació.

Extensions

- E valoració:* La font ja té una valoració per part del participant. (5)
- 1.- El sistema ho comunica al usuari i el cas d'ús acaba

Cas d'ús : Baixa Evaluate-Ítem

Actors Usuari

Propòsit Permetre al usuari donar de baixa avaluacions existents sobre fonts d'informació, per part de participants.

Resum El usuari realitzar la baixa d'una avaluació sobre una font d'informació (Ítem).

Curs Principal

- 1.- El usuari es troba en editar (cas d'ús modificar) el participant avaluador, o en l'alta el participant que es vol registrar.
(el usuari treballa en la pestanya de evaluate-Items)
- 2.- El sistema actualitza la llista de fonts que tenen valoració del participant seleccionat.
- 3.- El U selecciona la font i l'avaluació que rep.
- 4.- El Sistema demana confirmació d'eliminació
- 5.- El U confirma l'esborrat.
- 6.- El Sistema la desenregistra.

Extensions

- Cancel·lar:* Es cancel·la l'eliminació de l'avaluació. (5)
- 1.- El sistema no elimina i el cas d'ús acaba.

Cas d'ús : Alta Evaluate-Participant

Actors Usuari

Propòsit Permetre al usuari donar d'alta avaluacions sobre altres Participants

Resum El usuari realitzar l'alta d'una valoració sobre un participant.

Curs Principal

- 1.- El usuari selecciona es troba en editar (cas d'ús modificar) el participant *avaluador*, o en l'alta el participant que es vol registrar.
(el usuari treballa en la pestanya de evaluate-Participants).
- 2.- S mostra totes les valoracions que ha fet el participant sobre altres participants.
- 3.- El Usuari preme el botó "add".
- 4.- El S prepara un llistat de tots els participants (exceptuant ell mateix).
- 5.- El U selecciona el participant sobre la qual desitja fer l'avaluació.
- 6.- El U proporciona una descripció i selecciona el reliability del desplegable.
- 7.- El Sistema enregistra la avaluació.

Extensions

- E valoració:* el participant avaluat ja té una valoració per part del participant.(5)
- 1.- El sistema ho comunica al usuari i el cas d'ús acaba

Cas d'ús : Baixa Evaluate-Participant

Actors Usuari

Propòsit Permetre al usuari donar de baixa avaluacions existents sobre participants.

Resum El usuari realitzar la baixa d'una avaluació sobre un participant.

Curs Principal

- 1.- El usuari es troba en editar (cas d'ús modificar) el participant avaluador, o en l'alta el participant que es vol registrar.
(el usuari treballa en la pestanya de evaluate-Participants).
- 2.- El sistema actualitza la llista de participants que reben valoració per part del participant seleccionat.
- 3.- El U selecciona el participant-avaluació a esborrar.
- 4.- El Sistema demana confirmació d'eliminació
- 5.- El U confirma l'esborrat.
- 6.- El Sistema la desenregistra.

Extensions

- Cancel·lar:* Es cancel·la l'eliminació de l'avaluació. (5)
- 1.- El sistema no elimina i el cas d'ús acaba.

4.2. Model conceptual

Aquí pretenem mostrar el model conceptual del sistema software, així com aprofundir en la seva comprensió, justificant les classes i decisions preses.

Primer de tot es mostra el model conceptual íntegre del sistema, en aquest moments no conté cap normalització, ni l'aplicació de cap patró de disseny, ja que estem en l'especificació. Però aquí es mostra les classes d'objectes, les associacions entre les classes, així com els atributs de les classes.

Posteriorment es presenta el model conceptual per seccions per tal de facilitar-ne la seva comprensió. Es presenten tres seccions forçament diferenciades, un fragment representa la part dels Participants, la segona lo relació amb les fonts d'informació i l'última que serien les valoracions tant de participants com de les fonts d'informació com la relació "escrit per", que relaciona participants amb fonts d'informació. Un altre punt a presentar serien les restriccions textuais.

4.2.1 Fragments del Model Conceptual

4.2.1.1 Fragment 1: Participants

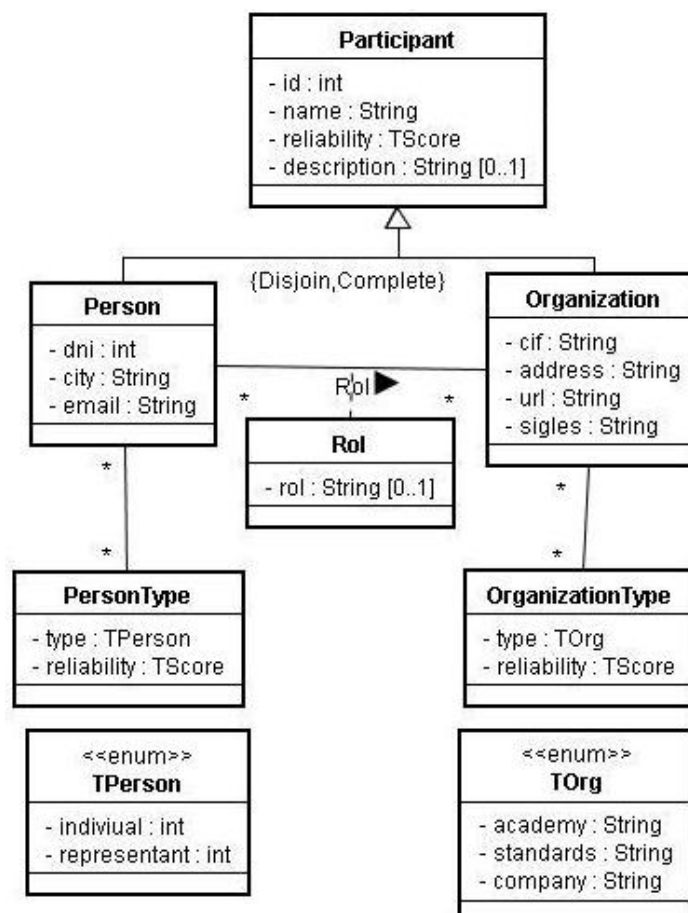


Fig. 12: Fragment del Model Conceptual relacionat amb Participants.

En aquest fragment es pretén ajudar a veure principalment com s'organitzen els participants, així com les possibles relacions entre ells.

Per una banda, un participant ha de ser una persona o una organització, però ni ninguna ni ambdues, de aquí l'especialització, de participant, sigui *disjoin* (sols pot ser un descendent i no més) i *complete* (ha de ser d'una de les subclasses especificada necessàriament).

Un tipus de persona es troba definida per un tipus i per un nivell de fiabilitat, aquets dos atributs provenen de uns conjunts de valors ja definits, concretament *TPerson* i *TScore* respectivament. Un *Person* pot tenir de 0 a

molts tipus de *Person* declarats i viceversa, d'aquí la relació “ * - * ” de la classe *Person* amb *PersonType*.

Semblant ocorre amb l'especialització *Organization*, pot tenir de 0 a molt tipus d'organitzacions, on el tipus de organització, esta compostat també per un tipus, en aquest cas *TOrg*, i per un nivell de fiabilitat, *TScore*.

Un objecte *Person* pot jugar un Rol en 0 o més *Organizations*, i viceversa; per lo que surt la classe associativa *Rol*.

Restriccions d'integritat textuals

- Claus externes: Participant (id:int), PersonType (type: TPerson, reliability: TScore).

4.2.1.2 Fragment 2: Fonts d'Informació.

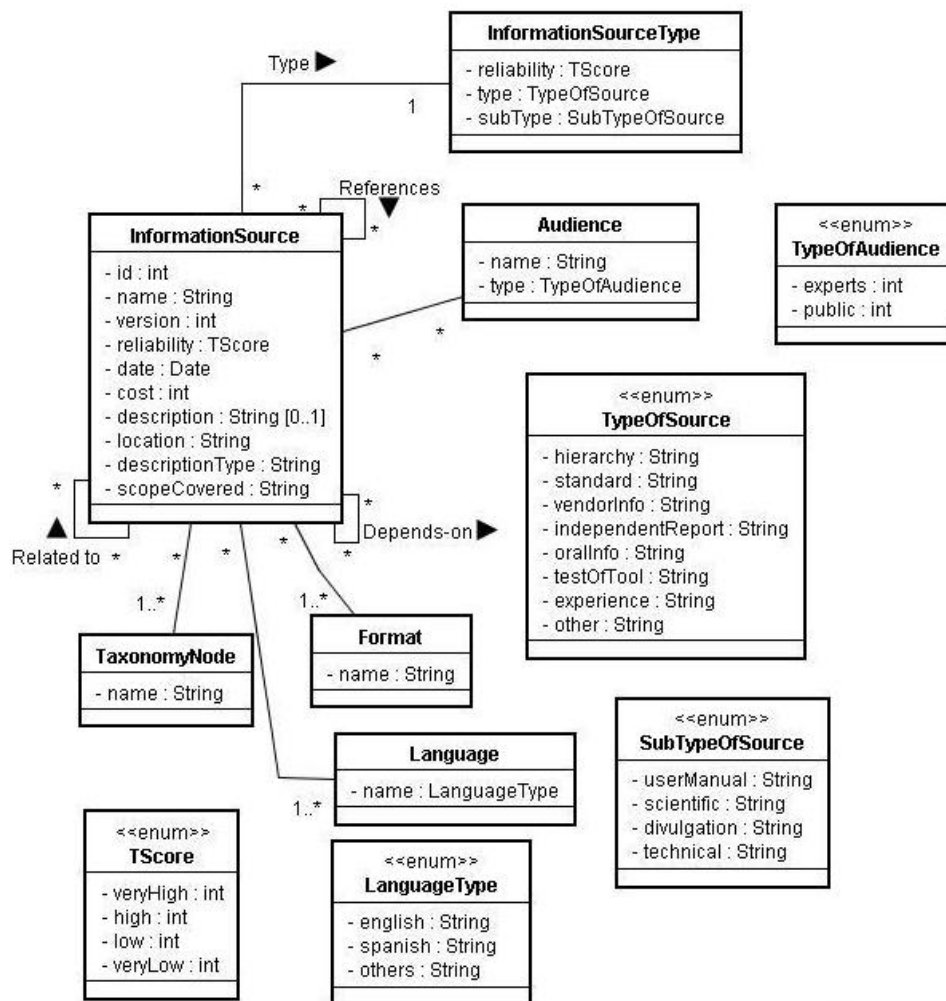


Fig. 13: Fragment del Model Conceptual relacionat amb les fonts d'informació.

En aquest fragment es vol mirar de que el lector pugui detectar clarament la relació de les fonts d'informació amb els diferents atributs definits en la taula d'atributs en el capítol anterior (Apartat 3.1), per la seva classificació, així com de les possibles relacions entre elles, ja sigui per relació, referència i/o dependència (constades també en la taula d'atributs).

Per una banda, una font d'informació manté una relació “* - 1” amb *InformationSourceType*, perquè una font d'informació té un únic tipus de font i es requereix, però un tipus de font pot ser el tipus de moltes fonts d'informació. El tipus de font es defineix per un atribut per al nivell de fiabilitat, un altre que indica el *type*, segons un conjunt de valors (columna de la taula d'atributs en el Apartat 3.1, figura 3), representat en l'enumeració *TypeOfSource*, i finalment, l'atribut *others* de la taula d'atributs que es veu reflectida en el camp *subType*, recordem que aquest pot ser d'un conjunt de valors els quals canvien segons el *type* seleccionat. Aquests conjunt de valors es troba en l'enumeració *SubTypeOfSource*.

Una font d'informació, pot estar relacionat com a mínim amb una taxonomia (*TaxonomyNode*), un Format, i un idioma(*Language*); però aquests poden estar relacionats amb diferents fonts d'informació. D'aquí les relacions “* - 1..*”. L'atribut *name* de *Language* correspon a un valor del conjunt *LanguageType*, el qual està compost pels llenguatges més significatius.

Després trobem que una font d'informació pot estar creada per a un/s tipus de públic (*Audiences*), aquest format per un nom i un tipus (*TypeOfAudience*), de on surt la relació “* - *”.

Finalment, en aquest fragment seria bo esmentar les relacions “* - *” entre la mateixa classe *InformationSource*, les quals representen:

related To: una font es troba relacionada amb altres fonts d'informació.

references: una font conté referències a altres fonts d'informació.

depends On: una font té requeriments d'altres fonts d'informació.

Restriccions d'integritat textuals

- Claus externes: InformationSource (id:int), Format (name:String), TaxonomyNode (name:String), Language (name:LanguageType), InformationSourceType (type.TypeOfSource, reliability:TScore, subType:SubTypeOfSource), Audience (name:String,type: TypeOfAudience).
- El valor de l'atribut subType de InformationSourceType dependrà del type que tingui.

4.2.1.3 Fragment 3: Valoracions.

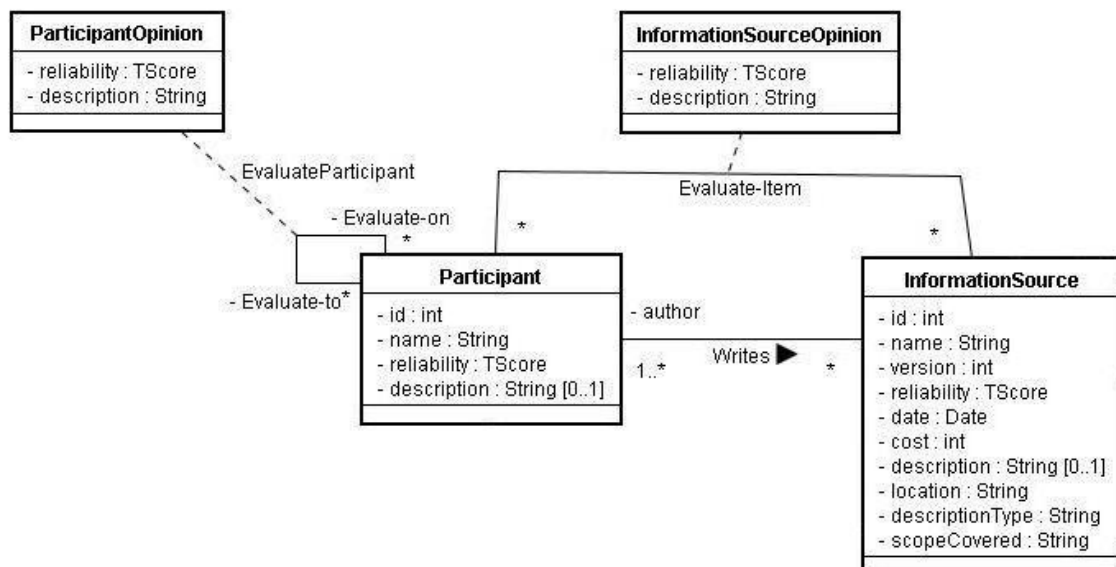


Fig. 14: Fragment del Model Conceptual relacionat amb les valoracions.

Un Participant escriu de 0 a molts InformationSources, i l'autor d'un InformationSource poden ser varis Participants, amb l'excepció de que al menys té que ser escrit per un Participant, és a dir, l'autor d'un InformationSource es de un o més Participants. D'aquí la relació *writes* "1..* - *".

Per altra banda, es descriu les valoracions sobre InformationSources i Participants. El fet que un Participant avalua una font d'informació es veu reflectit en la relació *Evaluate-Ítem*. La classe *InformationSourceOpinion* conté

el nivell de fiabilitat i descripció que un Participant dona sobre un InformationSource. La relació és “* - *” ja que un Participant pot avaluar diferents fonts, així com una font pot rebre diferents valoracions. Semblant es quan un Participant avalua altres Participants, cosa que es veu reflectida en la relació *Evaluate-Participant*. Aquesta classe conté el nivell de fiabilitat i descripció que un Participant realitza sobre un altre. Pel mateix raonament, aquesta relació es “* - *”.

Restriccions d'integritat textuais

- Un participant no pot avaluar un ítem que ha escrit ell mateix.
- Un participant no pot avaluar-se a ell mateix.

4.3. Model de Comportament

El propòsit d'aquest apartat es mostrar el model de comportament de l'eina, és a dir, com interactuen els actors amb l'eina *Manager of Components*.

4.3.1 Diagrames de Seqüència del sistema.

En aquest subapartat es mostren els diagrames de seqüència corresponents als casos d'ús descrits anteriorment.

Paquet 1: Participants

Cas d'ús : Alta Participant

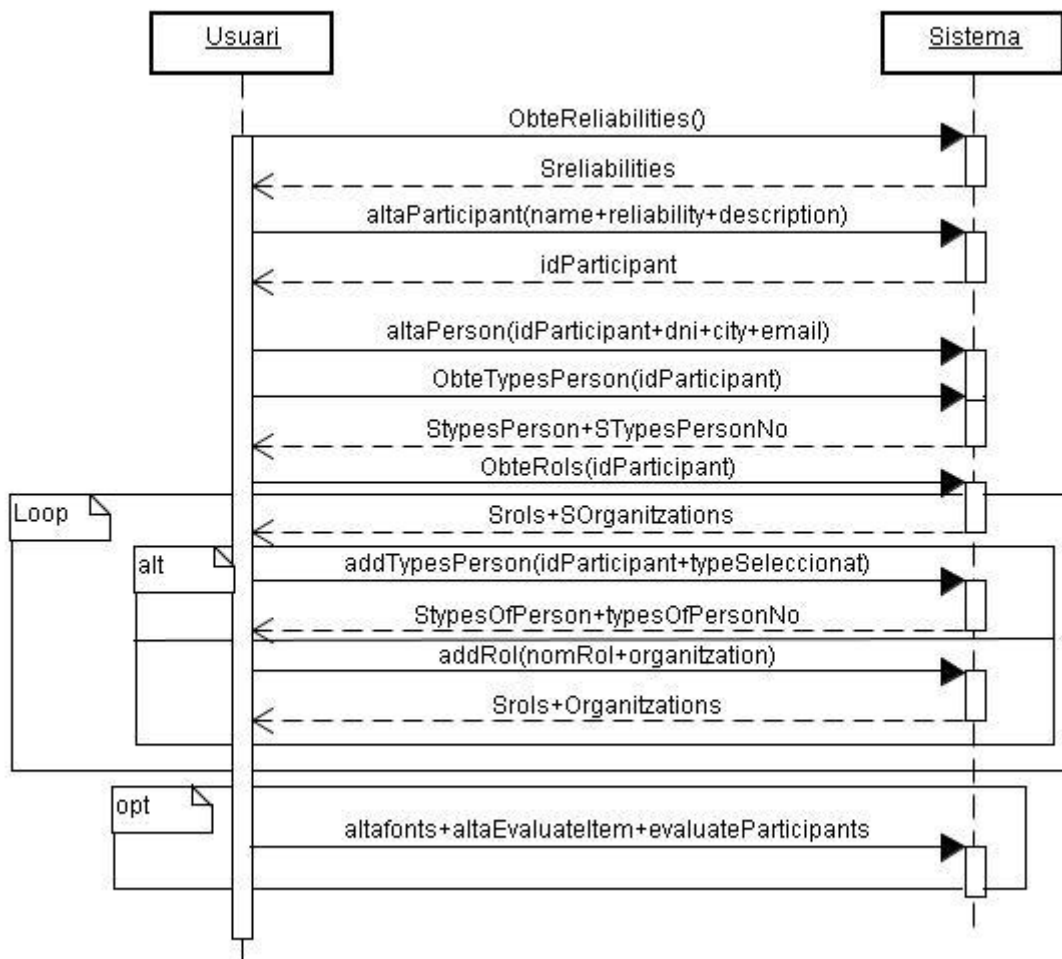


Fig. 15: Diagrama de seqüència d'alta Participant.

Cas d'ús : Baixa Participant

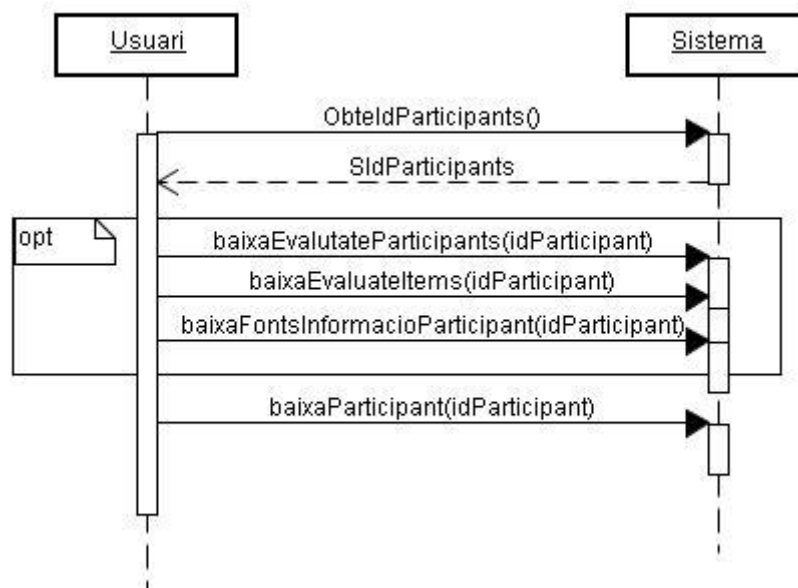


Fig. 16: Diagrama de seqüència de baixa Participant.

Cas d'ús : Consulta Participant

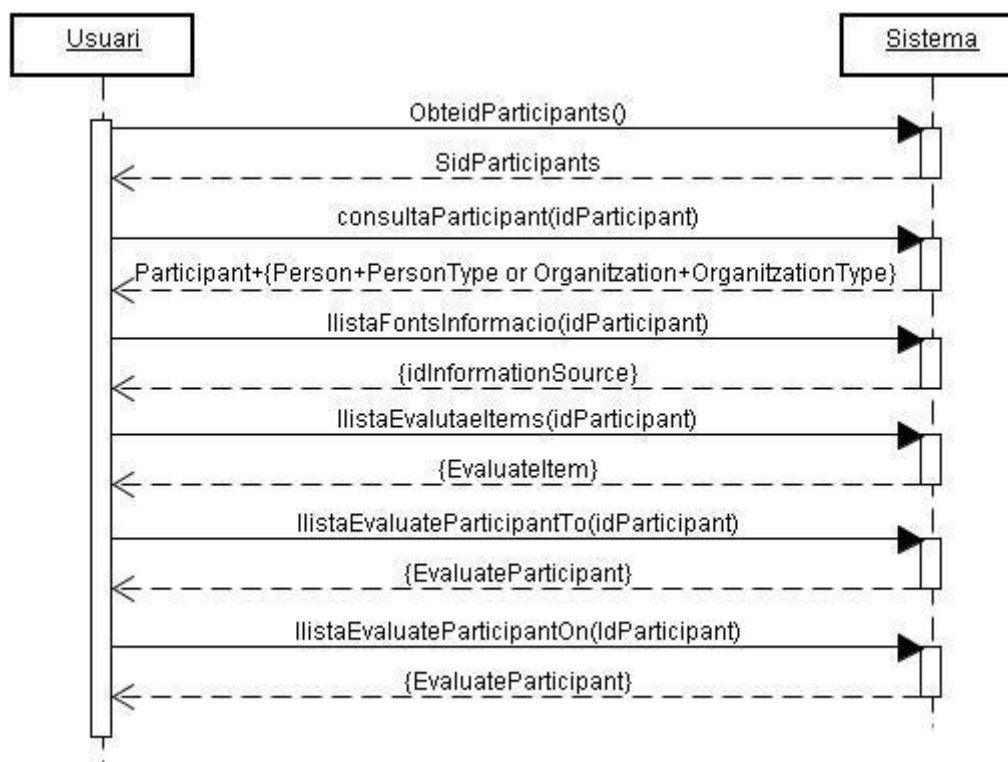


Fig. 17: Diagrama de seqüència de consulta Participant.

Cas d'ús : Modificar Participant

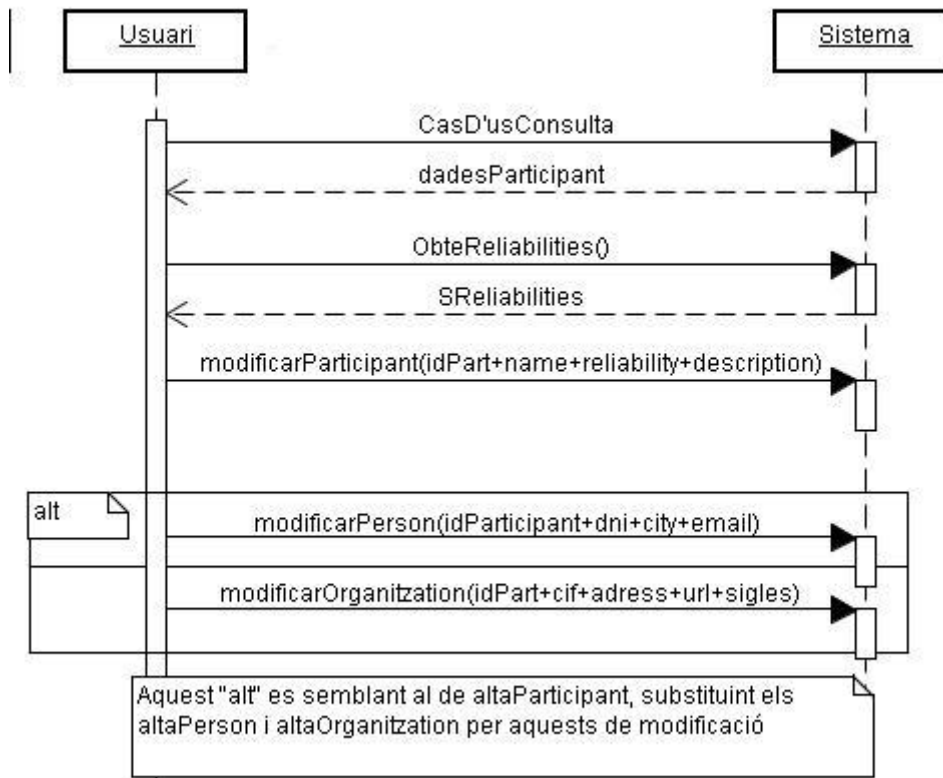


Fig. 18: Diagrama de seqüència de modificar Participant.

Cas d'ús : Llistar Participant

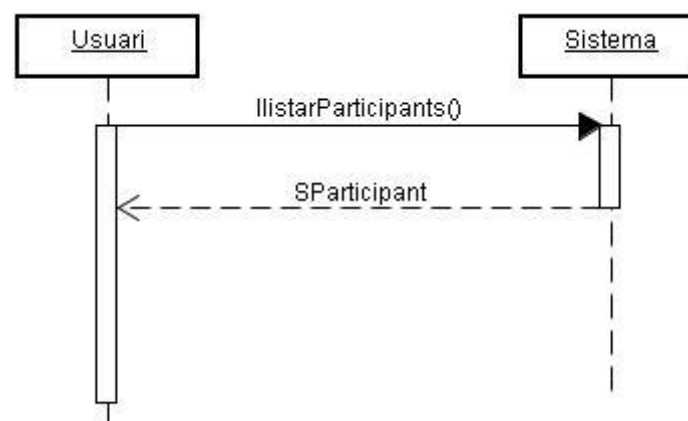


Fig. 19: Diagrama de seqüència de llistar Participants.

Paquet 2: Fonts d'informació

Cas d'ús : Alta Font d'Informació

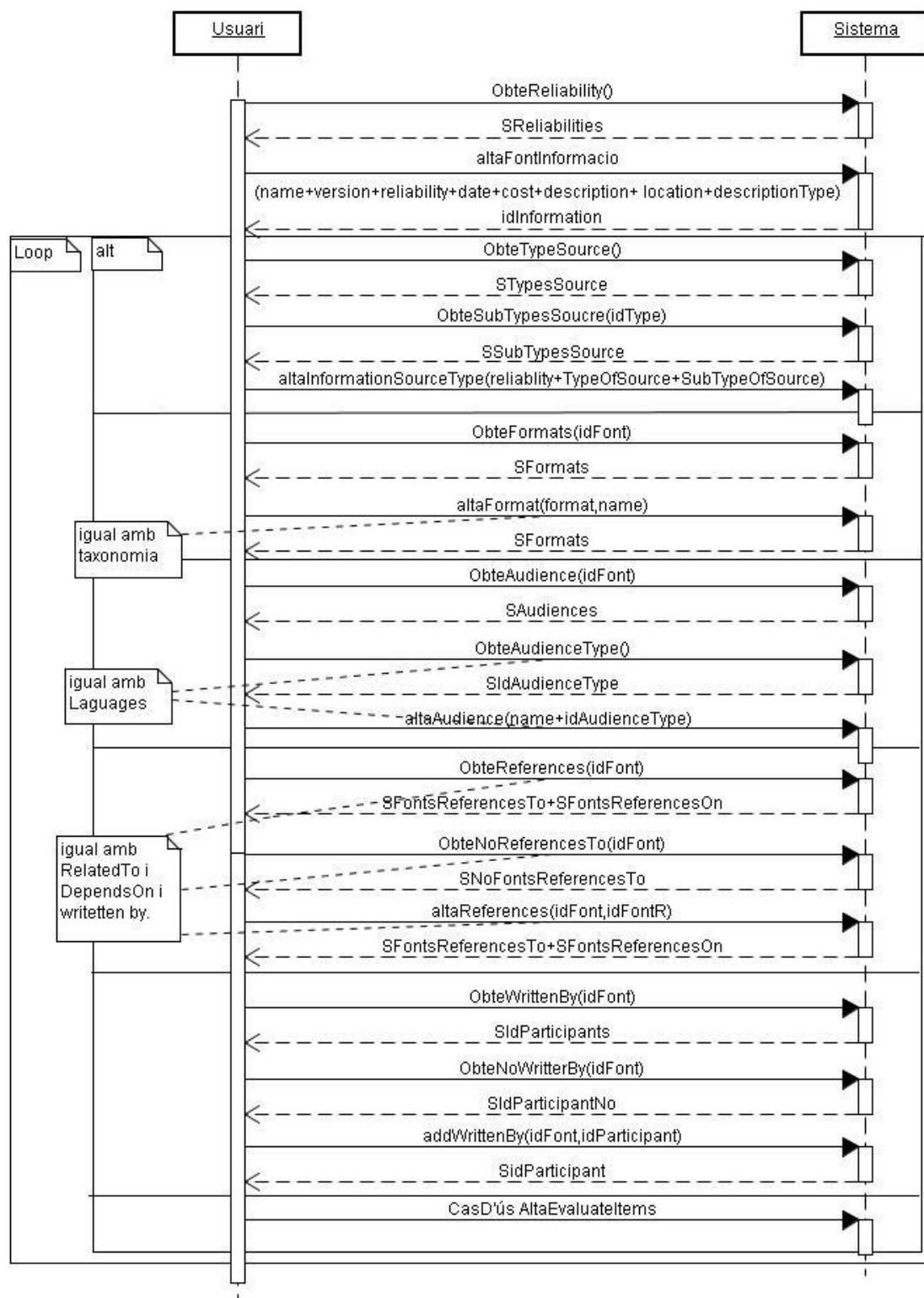


Fig. 20: Diagrama de seqüència d'alta Font d'Informació.

Cas d'ús : Baixa Font d'Informació

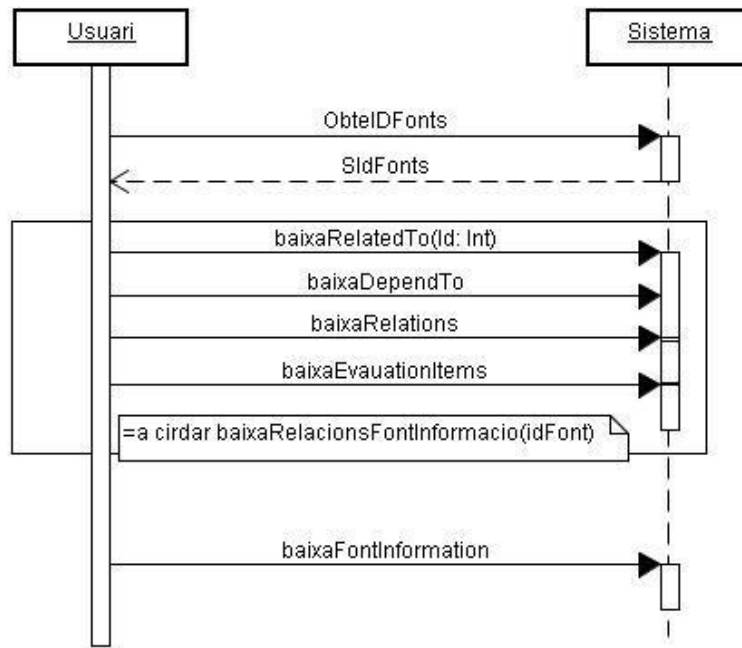


Fig. 21: Diagrama de seqüència de baixa Font d'Informació.

Cas d'ús : Consulta Font d'Informació

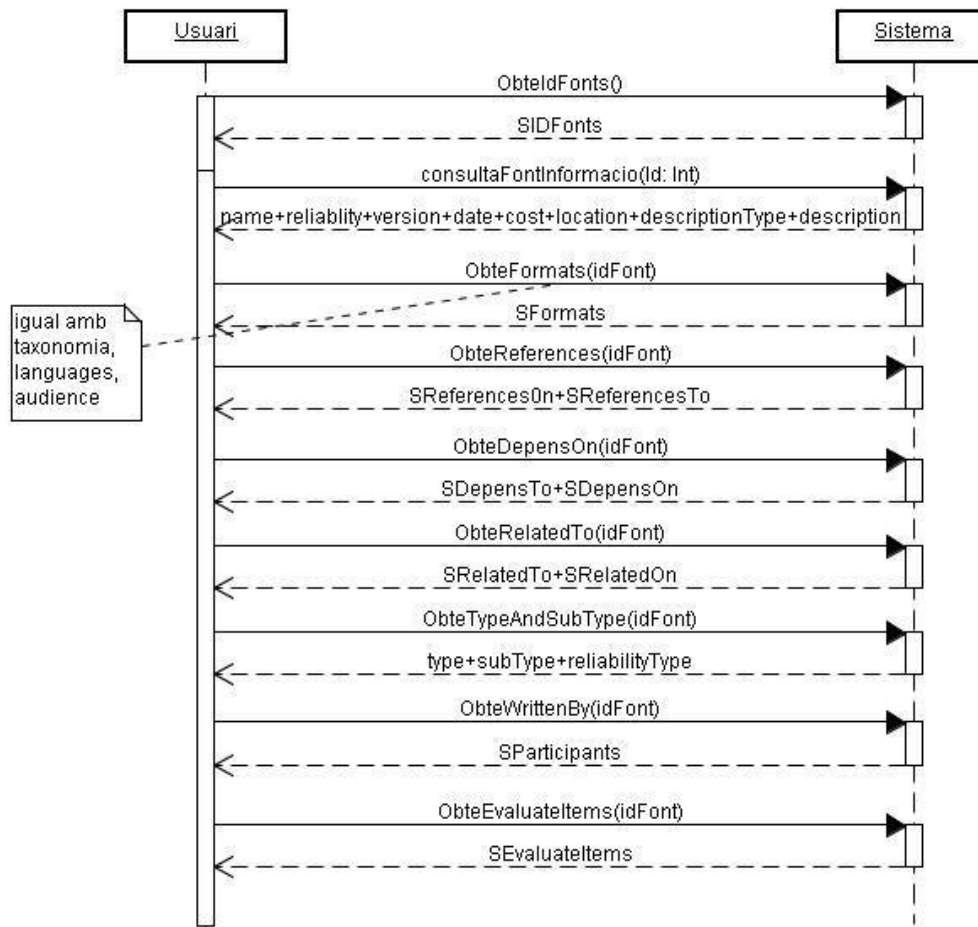


Fig. 22: Diagrama de seqüència de consulta Font d'Informació.

Cas d'ús : Modificar Font d'Informació

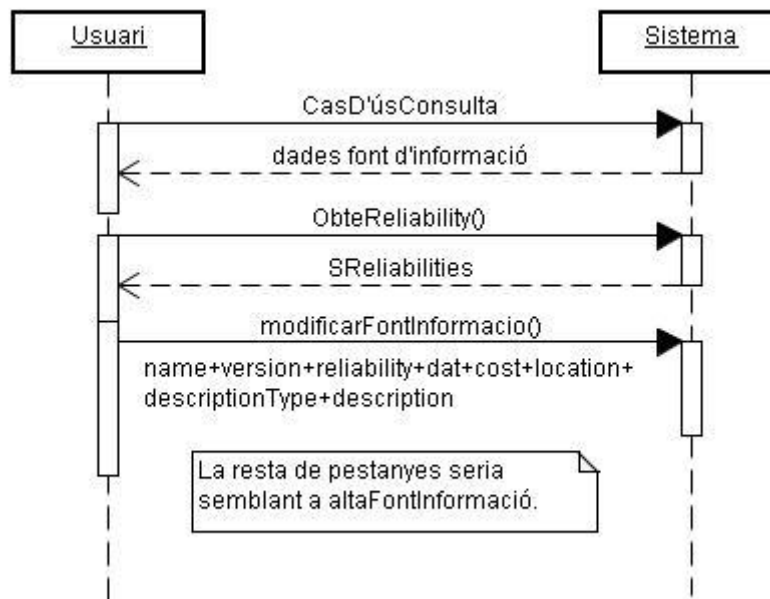


Fig. 23: Diagrama de seqüència de modificar Font d'Informació.

Cas d'ús : Llistar Fonts d'informació



Fig. 24: Diagrama de seqüència de llistar Fonts d'Informació.

Paquet 3: Formats

Cas d'ús : Alta Format

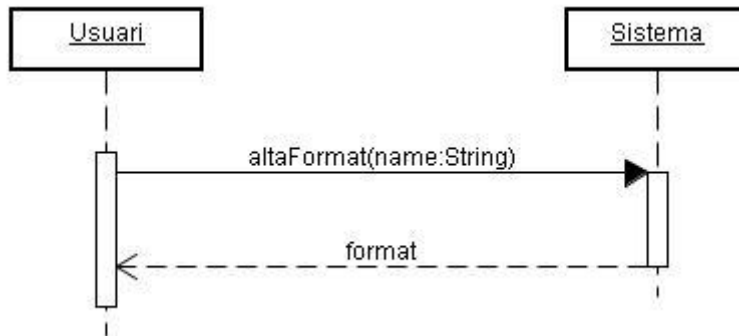


Fig. 25: Diagrama de seqüència d'alta Format.

Cas d'ús : Baixa Format

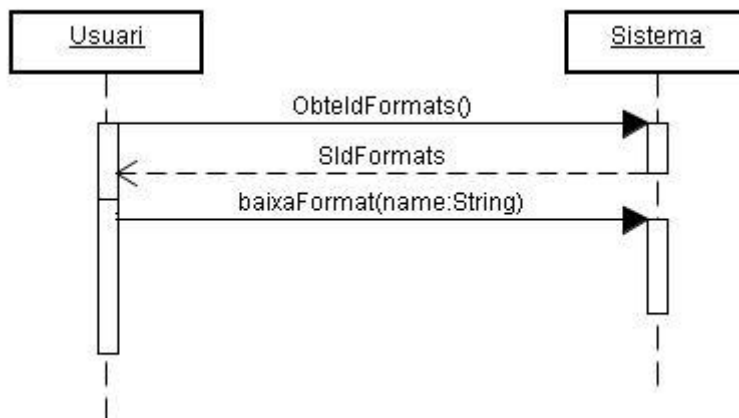


Fig. 26: Diagrama de seqüència de baixa Format.

Cas d'ús : Llistar Formats

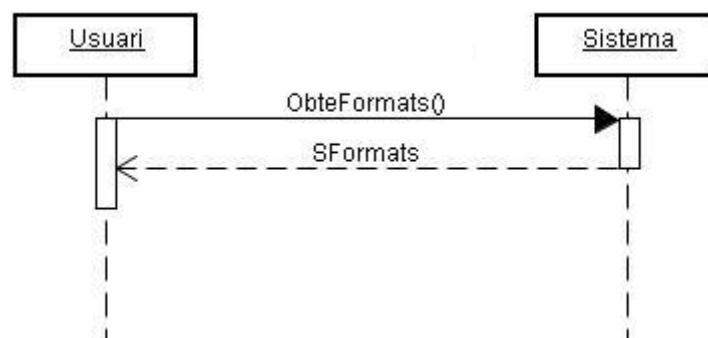


Fig. 27: Diagrama de seqüència de llistar Formats.

Paquet 4: Audiències

Cas d'ús : Alta Audience

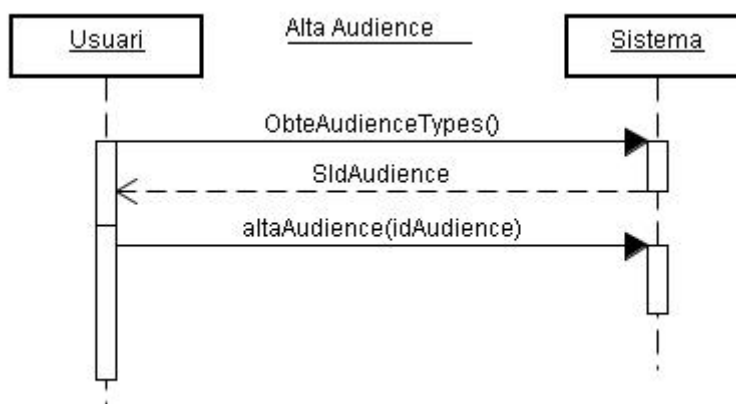


Fig. 28: Diagrama de seqüència d'alta Audience.

Cas d'ús : Baixa Audience

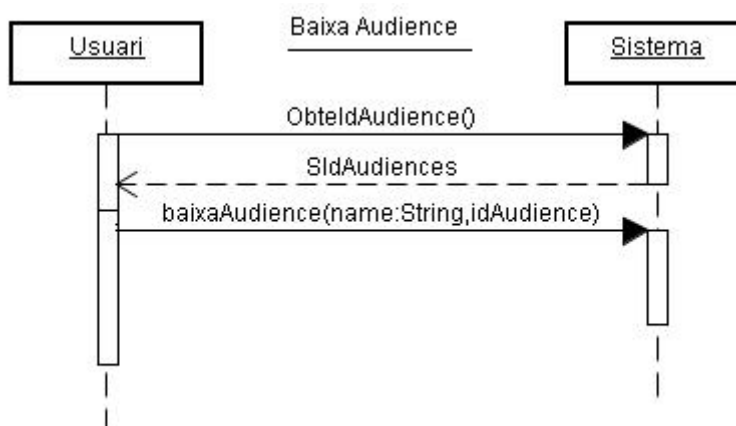


Fig. 29: Diagrama de seqüència de baixa Audience.

Cas d'ús : Llistar Audiències

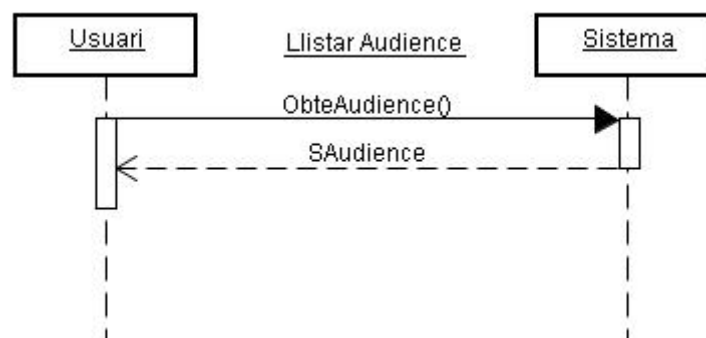


Fig. 30: Diagrama de seqüència de llistar Audiències.

Paquet 5: Valoracions

Cas d'ús : Alta Evaluate-Item

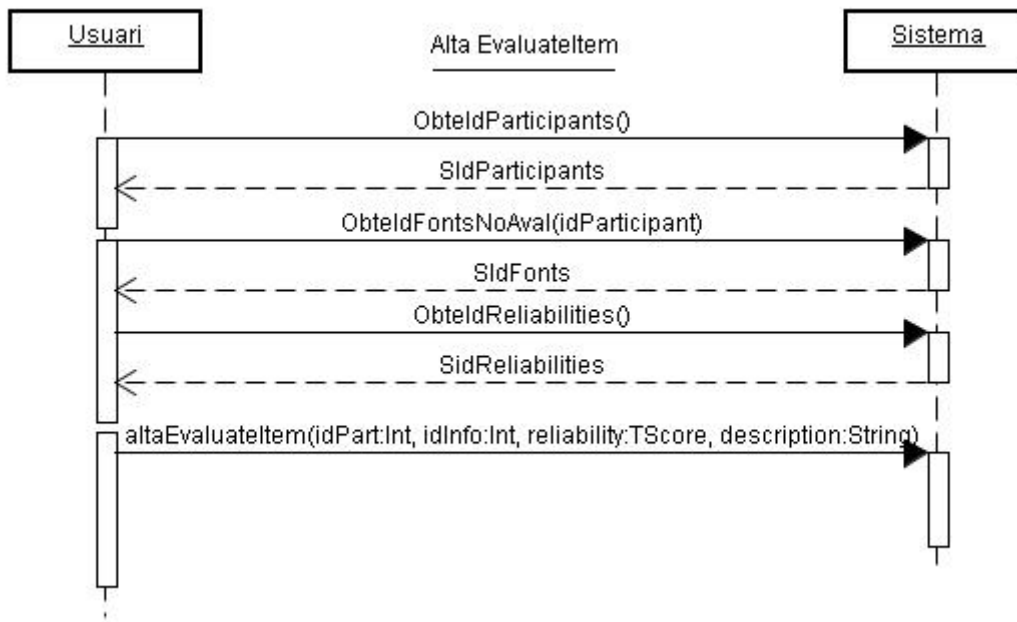


Fig. 31: Diagrama de seqüència d'alta Evaluate-Item.

Cas d'ús : Baixa Evaluate-Item

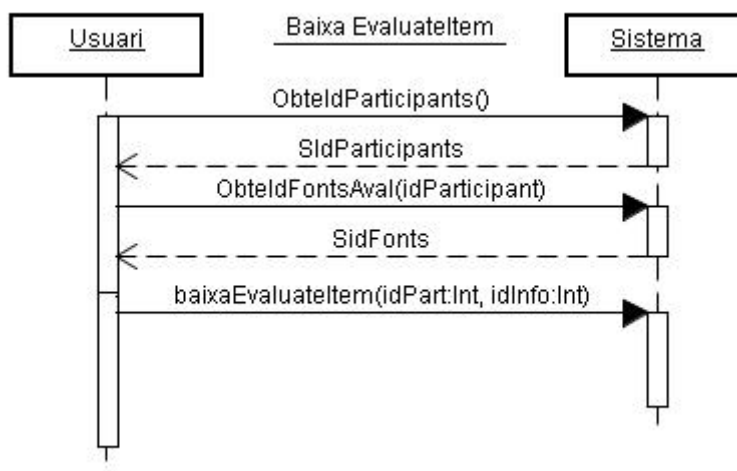


Fig. 32: Diagrama de seqüència de baixa Evaluate-Item.

Cas d'ús : Alta Evaluate-Participant

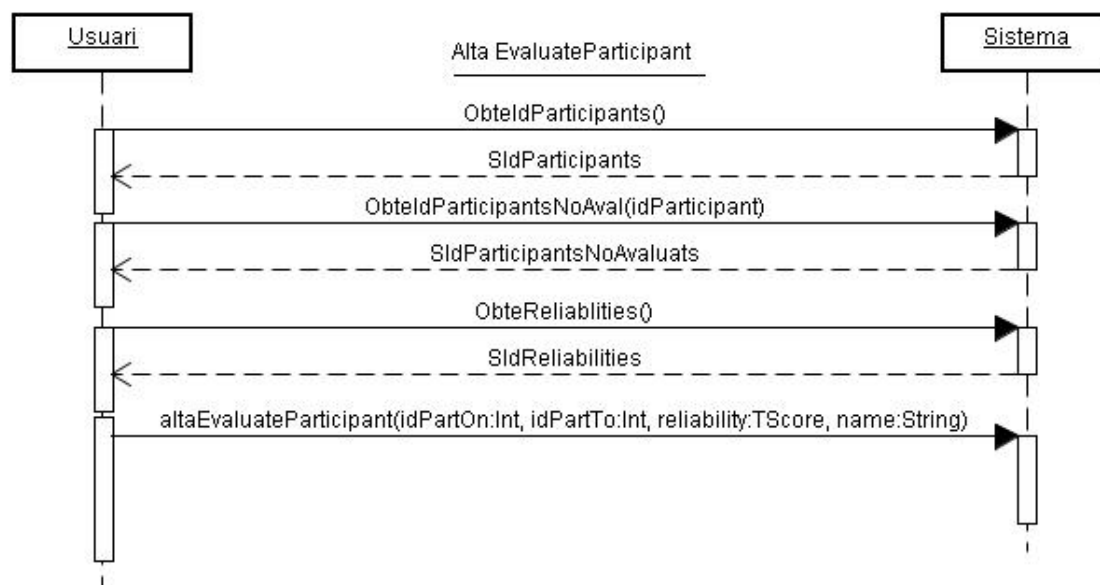


Fig. 33: Diagrama de seqüència d'alta Evaluate-Participant

Cas d'ús : Baixa Evaluate-Participant

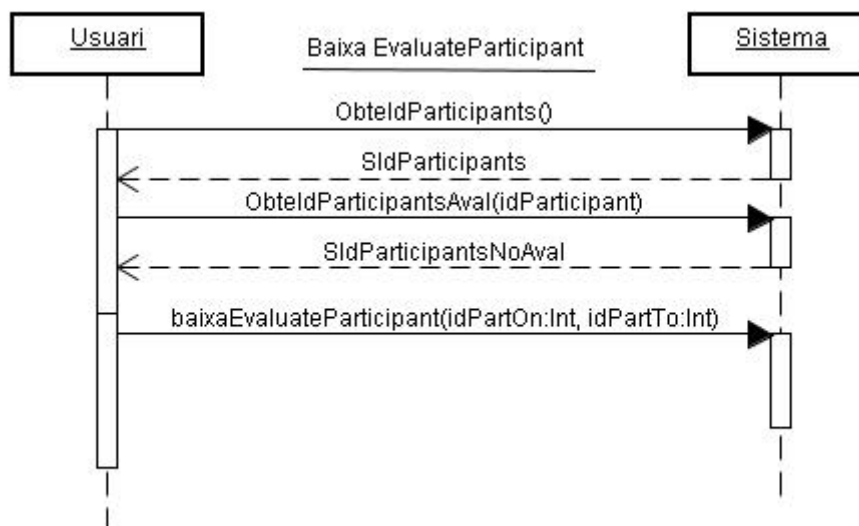


Fig. 34: Diagrama de seqüència de baixa Evaluate-Participant

4.3.2 Contractes de les operacions.

En aquest subapartat es mostren els contractes de les operacions indicades en els diagrames de seqüència anteriors. Cal indicar en aquest punt que els contractes són redactats en llenguatge natural, en lloc d'utilitzar el llenguatge d'especificació OCL, amb l'objectiu de fer més fàcil la comprensió per a lectors no experts amb el llenguatge.

Paquet 1: Participants

Cas d'ús: Alta Participant

Operació obteReliabilities(OUT: SReliabilities)

Semàntica Obtenir el llistat de reiliabilities.

Pre

Post 2.1-Es busca el conjunt de reliabilities.

Return Bool.

Operació altaParticipant (IN:name+reliability+description, OUT: idPersona)

Semàntica donar d'alta un Participant.

Pre 1.1-No E el participant

Post 2.1-Es crea una instància de Participant.

Return Bool.

Operació altaPerson(IN:idParticipant+dni+city+email)

Semàntica donar d'alta l'especialització d'un participant com a person.

Pre 1.1-No E l'especialització del participant

Post 2.1-S'especialitza el participant com a Person

Return Bool.

Operació obteTypesPerson(IN: idParticipant, OUT: STypesPerson+STypesPersonNo)

Semàntica Obtenir el llistat dels tipus Person del participant i els que no.

Pre:

Post: 2.1-Es busca el conjunt de tipus del participant i es posen en STypesPerson, en STypesPersonNo es col·loquen els que no sol del participant.

Return: Bool

Operació addTypePerson (IN: idParticipant+ TypeSeleccionat,

OUT: STypesPerson+STypesPersonNo)

Semàntica Actualitza el llistat dels tipus Person del participant i els que no.

Pre: 1.1-No E la relació.

Post: 2.1-Es crea la relació entre Person i PersonType. Actualitzant els llistats.

Return: Bool

Operació ObteRols(IN: idParticipant, OUT: SRols+SOrganitzacions)

Semàntica Obtenir el llistat dels rols del participant amb organitzacions. Així com el conjunt de organitzacions no relacionades.

Pre:

Post: 2.1-Es busca el conjunt dels rols del participant i les corresponents organitzacions.

Return: Bool.

Operació addRol(IN: idParticipant + rol+Organization, OUT: SRols+SOrganitzacions)

Semàntica Actualitzar el llistat dels rols del participant amb organitzacions.

Pre: 1.1-No E Organization amb el nom del rol indicat.

Post:2.1- Es crea la relació Rol entre el Person i la Organització donant a l'instància associativa el rol, 'actual·litzant el llistat dels rols del participant amb organitzacions.

Return: Bool.

Cas d'ús: Baixa Participant

Operació obteldParticipants(OUT: SIdParticipants)

Semàntica Obtenir el llistat de IdParticipants.

Post: 2.1-Es busca el conjunt de IdParticipants.

Return: Bool.

Operació baixaEvaluateParticipants(IN: idPart Int)

Semàntica donar de baixa tots els Evaluate-Participant relacionats amb el participant.

Pre: 1.1-E el participant

Post: 2.1-Es destrueix la relació Evaluate-Participant dels quals el IdPart es el qui fa la valoració com dels qui la rep, així com les corresponents instàncies de les associacions.

Return: Bool.

Operació baixaEvaluateItems(IN: idPart Int)

Semàntica donar de baixa tots els Evaluate-Item relacionats amb el participant.

Pre: 1.1-E el participant.

Post: 2.1-Es destrueix la relació Evaluate-Item dels quals el IdPart es el qui fa, així com les corresponents instàncies de les associacions.

Return: Bool

Operació baixaFontsInformacioParticipant (IN: idPart: Int)

Semàntica donar de baixa les Fonts d'informació d'un participant.

Pre: 1.1-E el participant.

Post: 2.1- Es destrueix les relacions amb Format, TaxonomyNode, Language, InformationSource, writes i audience.

2.2- Es destrueixen les relacions Related-to, Depends-on, References, així com les Evaluate-Items.

2.3- Es destrueix la relació Writes amb els participants corresponents.

2.4- Es destrueix la instància de InformationSource.

Return: Bool

Operació baixaParticipant(IN:idPart: Int)

Semàntica donar de baixa un participant

Pre: 1.1-E el participant.

Post: 2.1-Es destrueix la instància de participant, així com de l'especialització (i de la corresponent relació amb el Type).

Return: Bool

Cas d'ús: Consulta Participant

Operació obteldParticipants , ja descrit en el cas d'ús Baixa Participant.

Operació consultaParticipant(IN: id: Int, OUT: Participant + ({ Person+PersonType} or {Organization+OrganizationType}))

Semàntica mostrar les dades del participant sol·licitat.

Pre: 1.1-E el participant.

Post: 2.1-Es busquen els valors del participant

Return Bool

Operació llistaFontsInformacio(IN: IdParticipant, OUT: {idInformaitonSource})

Semàntica donar una llista de totes les fonts d'informació d'un Participant.

Pre: 1.1-E el participant.

Post: 2.-1Es seleccionen totes les fonts que tenen la relació Writes amb el participant indicat.

Return : Bool

Operació llistaEvaluateItems(IN: idPart, OUT: {EvaluateItem})

Semàntica llista totes les valoracions que ha fet un participant sobre fonts d'informació.

Pre: 1.1-E el participant.

Post: 2.1-Es llista totes les valoracions que ha fet un participant sobre les fonts d'informació.

Return: Bool.

Operació llistaEvaluateParticipantOn(IN: idPart , OUT: {EvaluateParticipant})

Semàntica llista totes les valoracions que ha rebut un participant.

Pre: 1.1-E el participant.

Post: 2.1-Es llista totes les valoracions que rep un participant.

Return: Bool.

Operació llistaEvaluateParticipantTo(IN: idPart, OUT: {EvaluateParticipant})

Semàntica llista totes les valoracions que ha fet un participant.

Pre: 1.1-E el participant.

Post:2.1-Es llista totes les valoracions que ha fet un participant.

Return: Bool

Cas d'ús: Modificar Participant

Operació Cas d'ús Consulta Participant.

Operació obteReliabilites, ja descrit en el cas d'ús Alta Participant.

Operació modificarParticipant(IN: id+dni+name+reliability+description)

Semàntica modificar les dades del participant sol·licitat

Pre: 1.1-E el participant.

Post: 2.1-Es modifiquen els atributs del participant amb les noves dades.

Return: Bool

Operació modificarPerson(IN: idParticipant+dni+city+email)

Semàntica modificar les dades de l'especialització d'un participant com a Person.

Pre: 1.1-E el participant.

Post: 2.1-Es modifiquen els atributs del Person.

Return: Bool

Cas d'ús: Llistar Participants

Operació llistaParticipants(OUT: SParticipant)

Semàntica donar una llista de tots els participants

Pre:

Post: 2.1-Es donen el conjunt de participants

Return: Bool

Paquet 2: Fonts d'Informació

Cas d'ús: Alta Font d'Informació

Operació altaFontInformacio(IN: name+version+reliability+dat+cost+description+location+descriptionType, OUT:idInfo)

Semàntica donar d'alta una font d'informació

Pre: 1.1-No E la font

Post: 2.1-Es crea una instància de InformationSource.

Return: Bool

Operació obteTypesSource(OUT: STypesSource)

Semàntica Obtenir el llistat dels TypeOfSource.

Pre:

Post: 2.1-Es busca el conjunt i es posen en STypesSource

Return: Bool

Operació obteSubTypesSource(IN:idTypeSource, OUT: SSubTypesSource)

Semàntica Obtenir el llistat dels SubTypeOfSource.

Pre: 1.1-E el TypeSource

Post: 2.1-Es busca el conjunt i es posen en STypesSource

Return: Bool

Operació altaInformationSourceType(IN:reliability+typeOfSource+subTypeSource)

Semàntica donar d'alta una InformationSourceType.

Pre: 1.1-E el typeOfSource i el subTypeOfSource

Post: 2.1-Es crea una instància de InformationSourceType si no E.

2.2-Es crea la associació de la InformationSourceType amb la Font.

Return: Bool

Operació obteFormats(IN: idFont, OUT: SFormats)

Semàntica Obtenir el llistat dels formats.

Pre: 1.1-E la font

Post: 2.1-Es busca i retorna el conjunt de Formats de la font.

Return: Bool

Operació altaFormat (IN:idFont+ idFormat, OUT: SForamts)

Semàntica Actualitza el llistat dels formats de la font d'informació.

Pre: 1.1-E la font

Post: 2.1-Es crea el format si no existeix

2.2-Es crea la relació entre Person i el format. Actualitzant el llistat.

Return: Bool

Operació obteAudience(IN: idFont, OUT: SAudiences)

Semàntica Obtenir el llistat de les Audiences de la font.

Pre: 1.1- E la font

Post: 2.1-Es busca i retorna el conjunt de Audiences relacionades amb la font.

Return: Bool

Operació obteAudienceType(OUT: SIdAudiencesType)

Semàntica Obtenir el llistat dels tipus d'audiencie.

Pre:

Post: 2.1-Es busca el conjunt d'audiencies.

Return: Bool

Operació altaAudience (IN:idFont+name+ idAudienceType, OUT: SAudienceType)

Semàntica Actualitza el llistat de les Audiences de la font d'informació.

Pre: 1.1-E la font i l'audienceType

Post: 2.1--Es crea l'Audience si no existeix.

2.2-Es crea la relació entre la font i aquest . Actualitzant el llistat.

Return: Bool

Operació obteReferences(IN:idFont, OUT: SReferencesOn+SReferencesTo)

Semàntica Obtenir el llistat de fonts referencés de la font tant To com ON.

Pre: 1.1-E la font

Post: 2.1-Es busca i dona els conjunt de referencés.

Return: Bool

*igual amb DependsOn i Related-to

Operació obteNoReferencesTo(IN:idFont, OUT: SNoReferencesTo)

Semàntica Obtenir el llistat de fonts no referencés de la font en To.

Pre: 1.1-E la font

Post: 2.1-Es busca i es dona els conjunt.

Return: Bool

*igual amb DependsOn i Related-to

Operació obteNoReferencesOn(IN:idFont, OUT: SNoReferencesOn)

Semàntica Obtenir el llistat de fonts no referencés de la font en On.

Pre: 1.1-E la font

Post: 2.1-Es busca i es dona els conjunt.

Return: Bool

*igual amb DependsOn i Related-to

Operació altaReferences (IN: idInformationOn:Int,
IdInformationTo:Int,OUT:SReferencesON+SReferncesTo)

Semàntica donar d'alta una relació References.

Pre: 1.1-No E la relació

Post: 2.1-E les fonts.

2.2-Es crea la relació References entre dues fonts d'informació.

Return: Bool

*igual amb DependsOn i Related-to

Operació obteWrittenBy(IN: idFont, OUT: SidParticipants)

Semàntica Obtenir el llistat de idParticipants que han escrit la font.

Pre: 1.1-E la font

Post: 2.1-Es busca i dona el conjunt.

Return: Bool

Operació obteNoWrittenBy(IN: idFont, OUT: SidParticipants)

Semàntica Obtenir el llistat de Idparticipants que no han escrit la font.

Pre: 1.1-E la font

Post: 2.1-Es busca i dona el conjunt.

Return: Bool

Operació addWrittenBy (IN:idFont+ idParticipant, OUT: SIdParticipant)

Semàntica Actualitza el llistat dels participants en la font d'informació.

Pre: 1.1-No E la relació

Post: 2.1- E la font i el Participant

2.2- Es crea la relació entre la font i el participant. Actualitzant el llistat.

Return: Bool

Cas d'ús: Baixa Font d'Informació

Operació obteldFonts(OUT: SIdFonts)

Semàntica Obtenir el llistat de IdFonts.

Pre:

Post: 2.1-Es busca el conjunt de IdFons.

Return: Bool

Operació baixaRelacionsFontInformacio(IN: idInformation:Int)

Semàntica donar de baixa les relacions d'una font d'informació.

Pre: 1.1-E la font

Post: 2.1-Es destrueix la relació Related-To, Depend-On, References en les quals
esta la font indicada.

2.2-Es destrueixen també les relacions Evaluate-Item amb les seves classes.

Return: Bool

Operació baixaFontInformacio (IN: idFont: Int)

Semàntica donar de baixa una Font d'informació.

Pre: 1.1E la font

Post:2.1-Es destrueix les relacions amb Format, TaxonomyNode, Language i
InformationSourceType.

2.2- Es destrueix la relació Writes amb els participants corresponents.

2.3- Es destrueix la instància de InformationSource.

Return: Bool

Cas d'ús: Consulta Font d'Informació

Operació obteldFonts, ja descrit en el cas d'ús Baixa Font d'Informació.

Operació consultaFontInformacio (IN: id: Int, OUT: InformationSource).

Semàntica mostra les dades d'una informació.

Pre: 1.1-E la font

Post: 2.1-buscar les dades de la font i mostrar-les.

Return: Bool

Operació obtFormatst, obtReferences i obtWrittenBy ja descrits en el cas d'ús Alta Font d'Informació.

Operació ObteTypeAndSubType(IN: IdFont, OUT: SInformationSourceType)

Semàntica Obtenir el llistat InfomationSourceType

Pre: 1.1-E la font

Post: 2.1-Es busca el conjunt i es mostra.

Return: Bool

Operació ObteEvaluateltem(IN: idInformation, OUT: {Evaluateltem})

Semàntica llista totes les valoracions que ha rebut una font d'informació.

Pre: 1.1-E la font

Post: 2.1-Es llista totes les valoracions que té una font d'informació.

Return: Bool

Cas d'ús: Modificar Font d'Informació

Operació obteldFonts, obtReliabilities ja descrits anteriorment.

Operació modificarFontInformacio (IN: id+name+version+reliability+date+cost+description+location+descriptionType)

Semàntica modificar les dades d'una informació.

Pre: 1.1-E la font

Post: 2.1-Es modifiquen els atributs de la informació amb les noves dades.

Return: Bool

Operació altaFormat (IN: idFont+ idFormat, OUT: SForamts)

Semàntica Actualitza el llistat dels formats de la font d'informació.

Pre: 1.1-E la font

Post: 2.1-Es crea el format si no existeix

2.2-Es crea la relació entre Person i el format. Actualitzant el llistat.

Return: Bool

Cas d'ús: Llistar Fonts d'Informació

Operació llistaFontsInformacio(OUT: {InformationSource})

Semàntica donar una llista de totes les fonts d'informació.

Pre:

Post: 2.1- Es dona la llista de fonts

Return: Bool

Paquet 3: Formats

Cas d'ús: Alta Format

Operació altaFormat (IN: name: String , OUT: idFormat)

Semàntica donar d'alta un nou tipus de format.

Pre: 1.1-No E el format

Post: 2.1-Es crea una instancia de format amb name = name.

Return: Bool

Cas d'ús: Baixa Format

Operació ObteFormats(IN: idFont, OUT: SFormats)

Semàntica Obtenir el llistat dels formats.

Pre:

Post: 2.1-Es busca i retorna el conjunt de Formats de la font.

Return: Bool

Operació baixaFormat (IN: idFormat: String)

Semàntica donar de baixa un tipus de format.

Pre: 1.1-E el format

Post: 2.1-Es destrueix la instancia de format amb name = name.

Return: Bool

Cas d'ús: Llista Formats

Operació llistaFormats(OUT: {Format})

Semàntica donar una llista de tots els formats.

Pre:

Post: 2.1-Busca el conjunt de formats

Return: Bool

Paquet 4: Audiencies

Cas d'ús: Alta Audience

Operació ObteAudienceType(OUT: SIdAudienciesType)

Semàntica Obtenir el llistat dels tipus d'audiencie.

Pre:

Post: 2.1-Es busca el conjunt d'audiencies.

Return: Bool

Operació altaAudience (IN:i idAudienceType+name,OUT:Audience)

Semàntica donar d'alta un nou Audience

Pre: 1.1-No E l'Audience.

Post: 2.1-Es crea l'Audience.

Return: Bool

Cas d'ús: Baixa Audience

Operació ObteldAudience(OUT: SIdAudiences)

Semàntica Obtenir el llistat del Audiences.

Pre:

Post: 2.1-Es busca i retorna el conjunt deAudiences.

Return: Bool

Operació baixaAudience (IN: idAudience)

Semàntica donar de baixa una audience

Pre: 1.1-E l'audience

Post: 2.1-Es destrueix la instancia d' Audience.

Return: Bool

Cas d'ús: Llista Audiences

Operació llistaAudiences(OUT: SAudiences)

Semàntica donar una llista de tots els Audiences.

Pre:

Post: 2.1-Busca el conjunt i el mostra.

Return: Bool

Paquet 5: Valoracions

Cas d'ús: Alta Evaluate-Item

Operació obteldParticipants ,ja detallat anteriorment.

Operació ObteldFontsNoAval(IN:IdParticipant,OUT: SIdFonts)

Semàntica Obtenir el llistat de IdFonts no avaluades pel participant.

Pre: 1.1-E el participant.

Post: 2.1-Es busca el conjunt de IdFons no avaluades pel participant.

Return: Bool

Operació obteliabilities, ja detallat anteriorment.

Operació altaEvaluateltem (IN:idPart:Int, IdInfo:Int, reliability: TScore,
description:String)

Semàntica donar d'alta un Evaluate-Item.

Pre: 1.1- No E el Evaluateltem

1.2-La font no es writtenBy pel participant..

Post: 2.1-Es crea la relació Evaluate-Item entre el participant i la informació.

2.2- Donant a l'opinió els valors a reliability i description.

Return: Bool

Cas d'ús: Baixa Evaluate-Item

Operació obteldParticipants ,ja detallat anteriorment.

Operació ObteFontsAval(IN:IdParticipant,OUT: SIdFonts)

Semàntica Obtenir el llistat de IdFonts avaluades pel participant.

Pre: 1.1-E el participant.

Post: 2.1-Es busca el conjunt de IdFons avaluades pel participant.

Return: Bool

Operació baixaEvaluateItem (IN: idPart:Int, IdInfo:Int)

Semàntica donar de baixa un Evaluate-Item.

Pre: 1.1- E el evaluateItem

1.2- E el Participant i la font.

Post: 2.1-Es destrueix la relació Evaluate-Item entre el participant i la informació.
Així com la instància de la associació.

Return: Bool

Cas d'ús: Alta Evaluate-Participant

Operació obteldParticipants ,ja detallat anteriorment.

Operació ObteldParticipantsNoAval(IN:IdParticipant,OUT: SIdParticipants)

Semàntica Obtenir el llistat de IdParticipants no avaluats pel participant.

Pre: 1.1-E el participant.

Post: 2.1-Es busca el conjunt de Participants no avaluats pel participant, excepte ell mateix.

Return: Bool

Operació obteReliabilities, ja detallat anteriorment.

Operació altaEvaluateParticipant (IN: idPartOn:Int, IdPartTo:Int, reliability: TScore, description:String)

Semàntica donar d'alta un Evaluate-Participant

Pre: 1.1- E els dos participants

1.2.- son diferents.

Post: 2.1-Es crea la relació Evaluate-Participant entre els participants.
Donant a l'opinió els valors a reliability i description.

Return: Bool

Cas d'ús: Alta Evaluate-Participant

Operació obteldParticipants ,ja detallat anteriorment.

Operació obteldParticipantsAval(IN:IdParticipant,OUT: SIdParticipants)

Semàntica Obtenir el llistat de IdParticipants avaluats pel participant.

Pre: 1.1-E el participant.

Post: 2.1-Es busca el conjunt de Participants avaluats pel participant.

Return: Bool

Operació baixaEvaluateParticipant(IN: idPartOn:Int, IdPartTo:Int)

Semàntica donar de baixa un Evaluate-Participant.

Pre: 1.1-E els dos participants.

1.2-E la relació.

Post: 2.1-Es destrueix la relació Evaluate-Participant entre els participants.
Així com la instància de la associació.

Return: Bool

5. Disseny

En aquest capítol es defineix el disseny de la solució proposada al problema plantejat i especificat en els capítols anteriors. Aquesta etapa del desenvolupament del software mira de respondre a *Com ho fa el sistema ?* i es, a diferència de les anteriors, depenent de la tecnologia.

Així, en primer lloc, es presenta l'arquitectura software del sistema, mirant de facilitar a l'usuari la seva comprensió, així com explicant les decisions preses, com són el nombre de capes, la tecnologia emprada i l'estratègia de persistència.

A continuació, es troben les etapes inicials de la fase de disseny segons l'enginyeria del Software: model conceptual i contractes normalitzats, assignació a capes, especificació de cada capa, disseny extern i intern de la interfície, acabant amb el disseny de la base de dades.

Cal indicar que la primera etapa inicial es la construcció dels casos d'ús concrets, recollint tots els esdeveniments de la presentació i analitzar el impacte del disseny extern en els casos d'ús, determinant en quin moment les

dades es fan persistents, així com establir clarament la relació entre el cas d'ús i els elements del disseny extern. Però per evitar repeticions de casos d'ús, s'ha considerat adient indicar, en aquesta memòria, solament un sol cop els casos d'ús ja concrets, els quals es troben detallats en l'apartat 4.1 de l'especificació (pàg 29).

5.1. Arquitectura del sistema

5.1.1 Arquitectura de l'aplicació

Amb l'especificació i els requeriment no funcionals definits, cal determinar el patró arquitectònic més adequat i realitzar la corresponent adaptació.

Per equilibrar les forces entre els factors de qualitat: mantenibilitat, reusabilitat, portabilitat, provable... (veure apartat requeriments no funcionals, pàg21) i sabent que no es persegueix l'eficiència, sent-hi aquest un factor secundari. Es determina com la solució més apropiada, partint del repositori de patrons, l'aplicació de l'Arquitectura en capes, concretament el format per 3 capes; Patró estudiat en les assignatures de l'enginyeria del software a la carrera d'Informàtica i el propi al desenvolupar qualsevol interfície d'una aplicació informàtica.

Per una banda, l'estratègia de comunicació utilitzada, entre les capes adjacents, és el Model "empenta", on la informació es comunica en la petició del servei, és a dir, la capa que fa la petició ja envia la informació necessària per a que la capa inferior.

Per un altra, el que seria el tractament d'errors, es pujaran els errors per les capes fins arribar a la capa més superior, per així comunicar-ho al usuari mitjançant la interfície gràfica.

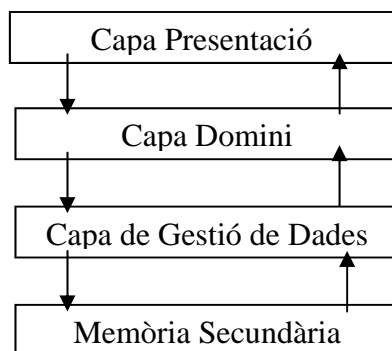


Fig. 35: Arquitectura 3 capes.

Capa Presentació

Aquesta capa és la responsable de la interacció amb l'usuari, és a dir, els serveis que presenta aquesta capa proporcionen la interfície necessària per presentar la informació i reunir dades. Generalment aquests serveis són identificats com la interfície d'usuari. Aquesta capa coneix com presentar les dades a l'usuari, però ignora quines transformacions cal fer per donar resposta a les peticions de l'usuari. Assegura els serveis de domini per oferir les capacitats de transaccions requerides e integrar a l'usuari amb l'aplicació per executar un procés de domini. S'assabenta de les peticions dels usuaris, ordena l'execució d'accions, rep i comunica els resultats, per altra banda, també realitza tractament de finestres, diàlegs, menús, botons, llistats...

Capa Domini

Aquesta capa es la responsable de la implementació de les funcionalitats del sistema (del que s'ha dit a l'especificació). Els serveis de domini són el "pont" entre un usuari i els serveis de dades. Ja que és qui rep l'entrada del nivell de presentació i, a la vegada, interactua amb els serveis de dades per executar les operacions de domini necessàries. Aquesta capa coneix com satisfer les peticions de l'usuari, però ignora on es guarden les dades i com es presenten a l'usuari. S'assabenta d'esdeveniments i consultes, controla la

validesa, canvia l'estat del domini, executa les accions encomendades, obté resultats i els comunica.

Capa Dades

Aquesta capa es troba sota de la capa de domini i és la responsable de la interacció amb el SGBD. Coneix on i com estan emmagatzemades les dades, però desconeix com tractar-les. S'ocupa de gestionar la persistència de les dades, les funcions concrets d'aquesta capa depenen de l'estratègia concreta de realitzar la persistència.

SGBD

Aquesta té la responsabilitat de gestionar les dades persistents. La qual s'encarrega de mantenir una representació persistent i concreta de l'estat del domini.

5.1.2 Estratègia de Persistència

Dintre de les estratègies de persistència es presenta l'opció d'elegir l'estratègia Disseny Directe o Generació automàtica.

S'ha elegit l'estratègia de Generació automàtica, en la qual la gestió de la persistència es transparent al dissenyador, ja que proporciona una traducció automàtica per emmagatzemar les dades en memòria externa; encara que pot no aprofitar completament les funcionalitats del SGBD. Transforma les actualitzacions fetes a memòria principal en actualitzacions en memòria externa. Requereix especificar la correspondència entre el diagrama de classes i l'esquema de la base de dades.

El motiu principal que ha portat a l'elecció d'aquesta estratègia és personal, ja que s'ha realitzat projectes aplicant l'estratègia de Disseny Directe, però en canvi, no s'ha desenvolupat cap projecte utilitzant la Generació

automàtica de la persistència. Per tant, és un fort al·licient el aprofitar aquest projecte per dependre aquesta estratègia, a més de que aquestes estratègies cada cop són més utilitzades en els desenvolupaments de sistemes software.

5.1.3 Tecnologia emprada

Amb la utilització d'un llenguatge de programació Orientat a objectes. S'aplica el patró "Orientació a Objectes" a cada capa, amb l'ajuda de UML (igual que s'ha fet en l'especificació).

De entre els principis de disseny d'una arquitectura orientada a objectes, ens centrem en:

- el principi de l'acoblament baix, el qual mesura el grau de connexió, coneixement i dependència d'aquesta classe respecte d'altres classes.
- el principi de la cohesió alta, el qual mesura el grau de relació i de concentració de les diverses responsabilitats (atributs, associacions i operacions), es a dir, mesura la relació que hi ha entre les coses d'una classe. Permet la fàcil comprensió, reutilització i manteniment.

La posterior aplicació dels patrons de disseny es fa tenint en compte aquest dos principis.

Per altra banda, el llenguatge de programació que es farà servir per al desenvolupament de l'eina serà el anomenat JAVA, el qual és orientat a objectes. Pel que fa a la generació automàtica, es realitzarà mitjançant una eina anomenada Hibernate, i el Sistema Gestor de bases de dades, amb el qual els objectes es faran persistents, podria ser mitjançant el MySQL (per bases de dades relacionals).

5.2. Model Conceptual Normalitzat

En aquest subapartat es representa el que seria la segona etapa del disseny, on es mostra el model conceptual del sistema normalitzat. Degut a la limitació de la tecnologia orientada a objectes: no permet implementar directament tots els conceptes que hem usat en l'especificació; s'han de realitzar un conjunt de tractaments per l'adaptació del model conceptual a les restriccions actuals, és a dir, es tracta d'obtenir un model equivalent e implementable en classes d'objectes, jerarquies, associacions... Aquest procés de normalització també s'anomena binarització, el qual impacta tant en el diagrama de classes com als contractes. Els tractaments que es requereixen són:

- Transformació d'associacions n-àries ($n > 2$) i classes associatives en associacions binàries.
- Transformació de la informació derivada en materialitzada o calculada.
- Afegir la comprovació de les restriccions d'integritat en els contractes.

També es contempla en el nou model la navegabilitat que hi ha entre les diferents classes, lo qual indica si és possible o no travessar una associació d'una classe a un altra (cal indicar que les navegabilitats en un únic sentit es present amb fletxes, però en canvi, les navegabilitats en ambdós sentits es presenta com una línia).

Per finalitzar aquest subapartat, es mostra per fragments les corresponents a la separació del model conceptual. De cada fragment es comentaran els canvis realitzats a destacar.

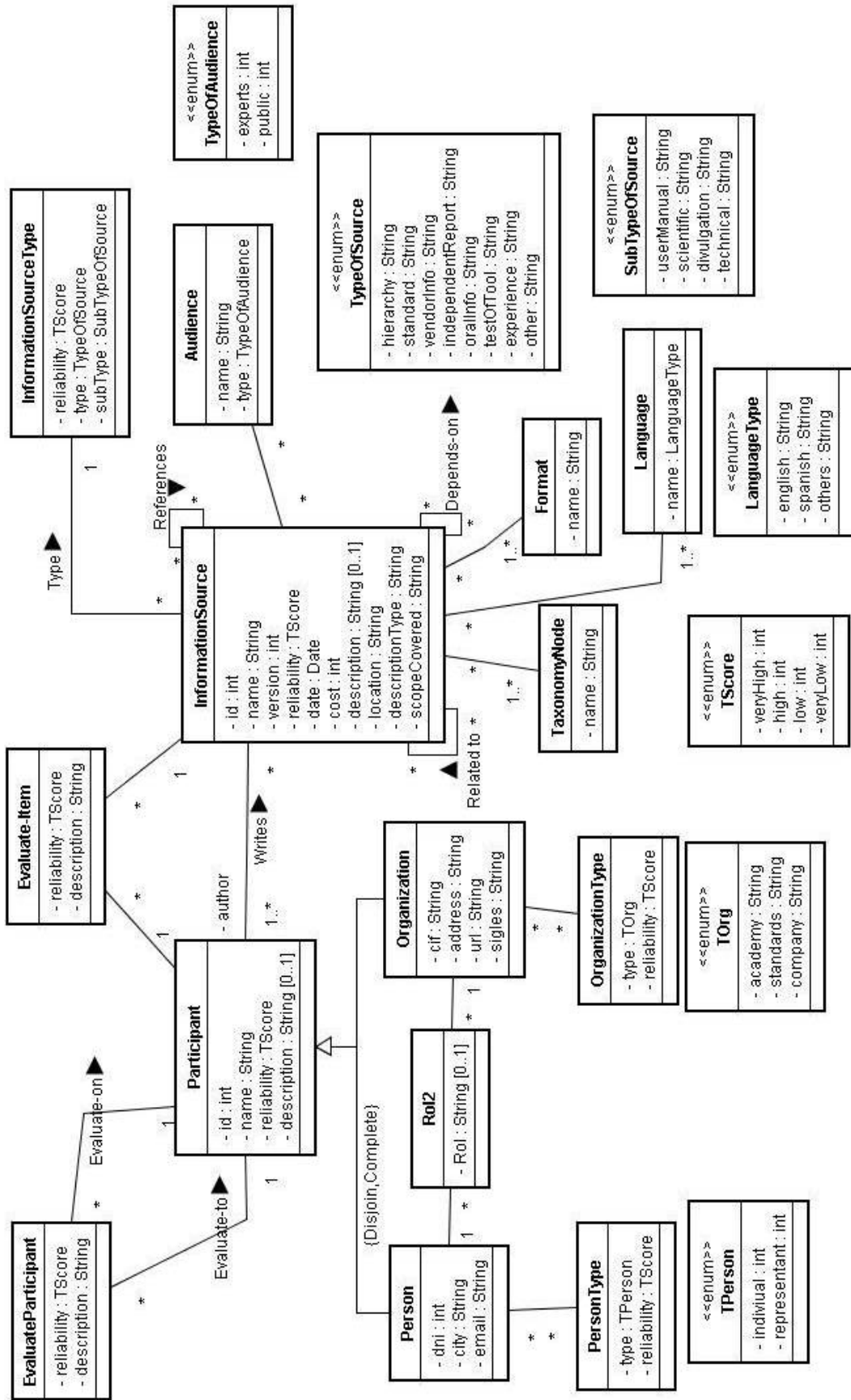


Fig. 36: Diagrama Model Conceptual Normalitzat.

5.2.1 Fragments del Model Conceptual Normalitzat

5.2.1.1 Fragment 1: Participants

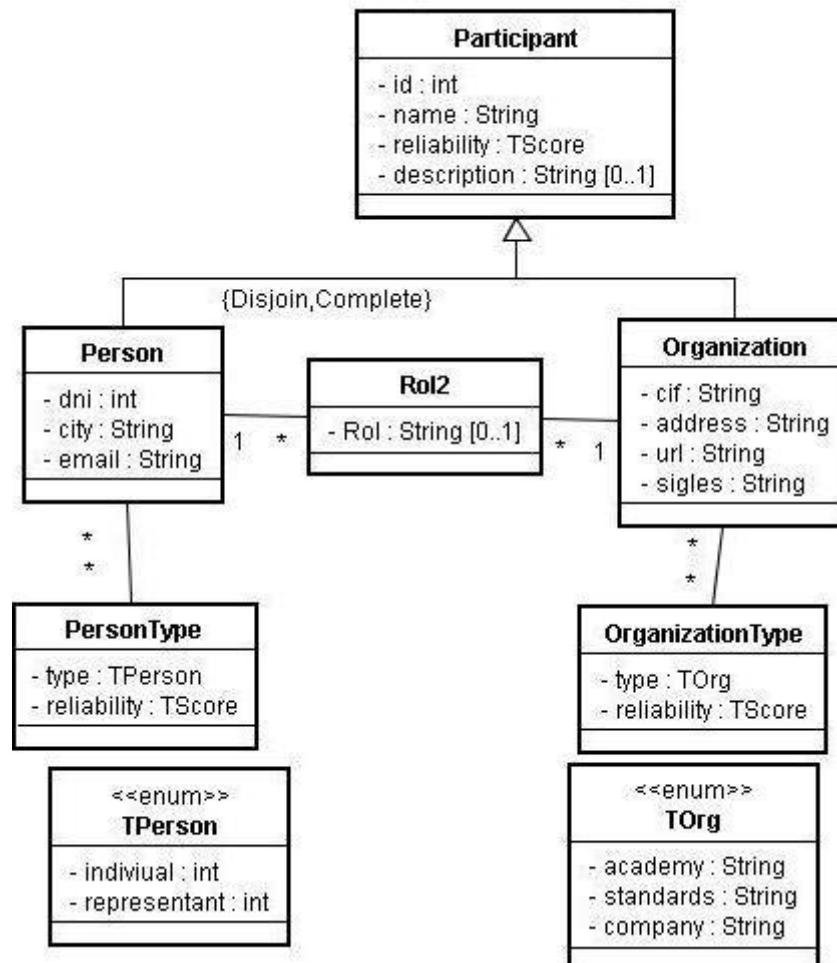


Fig. 37: Fragment del Model normalitzat relacionat amb els Participants.

En aquest fragment es presenten els canvis relacionats amb la part dels Participants.

Un canvi important és que la associació Rol, passa a ser una associació binària, on la classe Rol es relaciona amb la classe Person i per altra banda, amb la classe Organization.

Pel que fa a la navegabilitat, un Person accedeix al tipus (classe PersonType), però no es necessari a la inversa. Semblant passa amb la classe Organization i el seu tipus (classe OrganizationType).

Les restriccions d'integritat textuals continuen sen les anteriors.

5.2.1.2 Fragment 2: Fonts d'Informació.

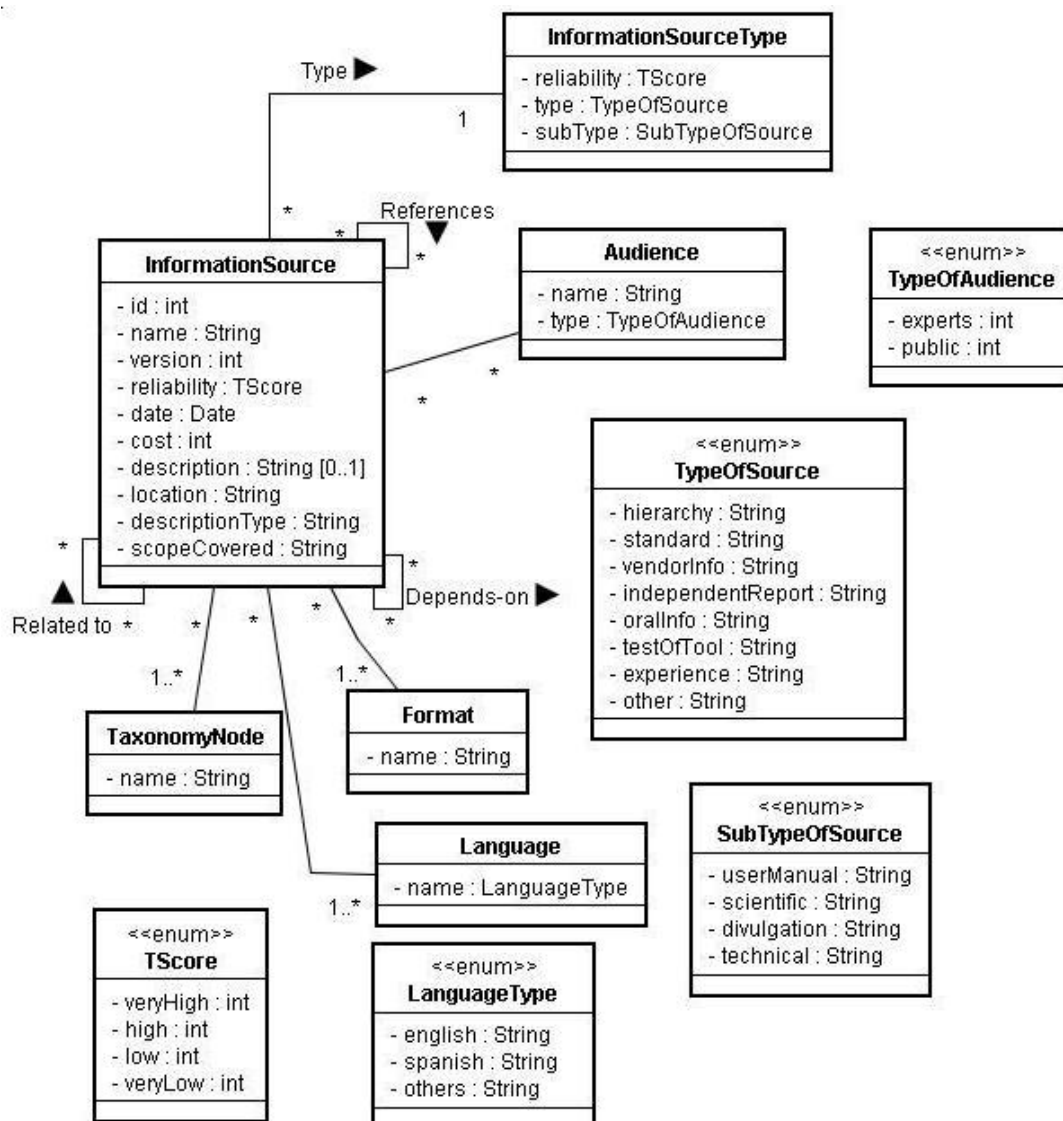


Fig. 38: Fragment del Model normalitzat relacionat amb les Fonts d'Informació.

En aquest fragment es presenten els canvis relacionats amb la part de les Fonts d'Informació.

Pel que fa a la navegabilitat, un **InformationSource** és la que accedeix a al tipus, als format, als llenguatge, així com a les taxonomies. En canvi, les relacions entre **InformationSources** es mantenen bidireccionals, tant la relació **DependsOn**, **RelatedTo** com **References**.

En aquest fragment les restriccions d'integritat textuals continuen sen les anteriors.

5.2.1.3 Fragment 3: Valoracions.

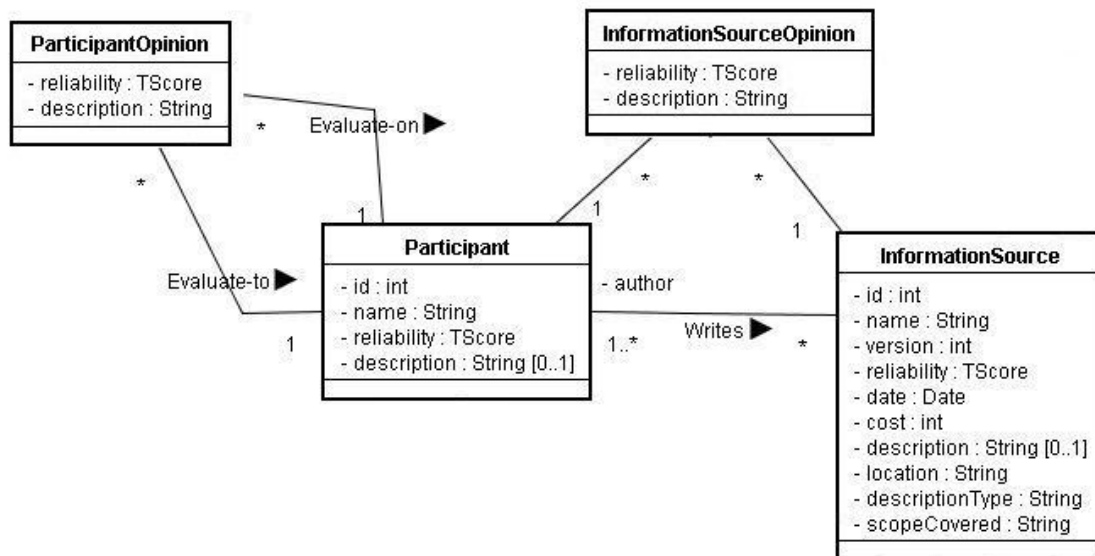


Fig. 39: Fragment del Model normalitzat relacionat amb les valoracions.

En aquest fragment es presenten els canvis relacionats amb la part de les Valoracions.

Un canvi, són que la associacions EvaluateParticipant i EvaluateItem, que passen a ser associacions binaries. La associació EvaluateParticipant passa a convertir-se en dos relacions: Evaluate-on i Evaluate-To.

Pel que fa a la navegabilitat, totes les relacions són bidireccionals. Les relacions Evaluate-on i Evaluate-To són bidireccionals, ja que és necessari saber, a partir d'un participant, les valoracions que realitza com les que rep per part d'altres participants. Semblant passa amb les relacions que té la classe InformationSourceOpinion, ja que es important per un InformationSource saber les valoracions que rep i qui la fa, com per cada participant saber les valoracions que realitza. Finalment, la relació Writes també es manté bidireccional, ja que es necessari saber per a un participant els InformationSource que escriu, com per cada InformationSource saber quins participants l'escriuen.

Les Restriccions d'integritat textuals no varien respecte a les anteriors.

5.3. **Assignació de Responsabilitats a capes**

En aquest apartat es realitza l'assignació de responsabilitats a cada una de les capes

Per poder assignar les responsabilitats a les capes, cal una preparació, aquesta tasca és anomenada "*Modificació dels contractes*". Aquesta tasca consisteix en:

- convertir les precondicions dels contractes en postcondicions, sempre amb l'excepció de que les precondicions dels casos d'ús continuen sent-ho, ja que les considerem prou fortes i no cal tocar-les.
- Les operacions que poden provocar errors retornaran un booleà per poder detectar-lo, i si abans retornaven un resultat aquest té que passar a ser un paràmetre més per la sortida.
- L'enumeració de totes les parts de la postcondició també es una taca, els controls d'errors s'enumeren com 1.x , els canvis d'estat o càlculs quan tot va bé com 2.x.

Pel que fa a la realització d'aquest pas, per evitar la repetició dels contractes, cal dir que en l'apartat 4.1 de l'especificació (pàg 29) es presenten els contractes amb la numeració que correspon a les modificacions que es realitzen en aquest punt. En el que es troba en l'apartat 4.1 sols ens faltaria moure els 1.x al post.

Arribats a aquest punt, cal assignar cada postcondició a una o més capes. Cal recordar que s'ha elegit una estratègia de Gestió automàtica per al tractament de la persistència, el que provoca una preponderància de la capa de domini. No es pot oblidar que les dades persistents acaben emmagatzemades a la base de dades.

A continuació, es presenten un parell d'exemples del que seria l'assignació de responsabilitats a les diferents capes de l'arquitectura.

Exemple 1: Cas d'ús Alta Participant

Responsabilitat		Assignació	
Operació	Post	Capa	Justificació
ObteReliabilities	2.1	Presentació	Inicialitza i mante el conjunt SReliabilities.
		Domini	Buscar les dades.
AltaParticipant	1.1	Domini	Comprova que no E el participant .
	2.1	Domini	Mante les instancies de Participant i relacions.
Enregistra les dades.			
AltaPerson	1.1	Domini	Comprova que no E l'especialització i enregistra les dades
	2.2		
ObteTypesPerson	2.1	Domini	Crea la instancies de l'especialització Person i busca les dades.
		Presentació	Inicialitza i mante els conjº StypePerson/No.
ObteRols	2.1	Presentació	Inicialitza i mante els conjº SRols i SOrganitzations.
		Domini	Buscar les dades.
AddTypesPerson	1.1	Presentació	Nomès permet seleccionar els del conjunt.
	2.1	Domini	Crea les relacions entre typesPerson i Person
Enregistra les dades.			
AddRol	1.1	Presentació	Només permet seleccionar els no assignats.
	2.1	Domini	Crea les instancies i relacions.
Enregistra les dades			
Altes Operacions: Cas D'ús AltaFonts, AltaEvaluateItems, AltaEvaluateParticipants.			

Exemple 2: Cas d'ús Alta Font d'Informació

Responsabilitat		Assignació	
Operació	Post	Capa	Justificació
ObteReliabilities	2.1	Presentació	Inicialitza i mante el conjunt SReliabilities.
		Domini	Buscar les dades.
AltaFontInformacio	1.1	Domini	Comprova que no E la font.
	2.1	Domini	Mante les instancies de la font d'informació i
Enregistra les dades.			
ObteTypesSource	2.1	Presentació	Inicialitza i mante el conj° STypeSource.
		Domini	Buscar les dades.
ObteSubTypesSource	1.1	Presentació	Comprova que existeix el TypeSource
	2.1	Presentació	Inicialitza i mante el conj° SSubTypeSource.
		Domini	Buscar les dades.
AltaInformationSource Type	1.1	Presentació	Comprova que E la els Type i SubType Només permet seleccionar els del conjunt.
	2.1	Domini	Crea les instancies i les relacions.
			Comprova l'existència i enregistrament de dades
2.2			
ObteFormats	1.1	Presentació	Comprova que E la Font.
	2.1	Presentació	Inicialitza i mante els conj° SFormats.
		Domini	Buscar les dades.
AltaFormat	1.1	Presentació	Comprova que E la Font.
	2.1	Domini	Crea les instancies i la associació amb la Font
			Comprovació i enregistra les dades
2.2			
ObteAudiencies	1.1	Presentació	Comprova que E la Font.
	2.1	Presentació	Inicialitza i mante el conj° SAudiencies.
		Domini	Buscar les dades.
ObteAudienciesTypes	2.1	Presentació	Inicialitza i mante el conj° SAudienciesType.
		Domini	Buscar les dades.
AltaAudience	1.1	Presentació	Comprova que E la font i l'audienceType. Només permet seleccionar els del conjunt
	2.1	Domini	Crea les instancies i les relacions.
			Comprova l'existència i enregistrament de dades
2.2			
ObteReferences	1.1	Presentació	Comprova que E la font.
	2.1	Presentació	Inicialitza i mante el conjunt SreferencesOn i

		Domini	Buscar les dades.
ObteNoReferencesTo	1.1	Presentació	Comprova que E la font.
	2.1	Presentació	Inicialitza i mante el conjunt SNoReferencesTo
		Domini	Buscar les dades.
ObteWrittenBy	1.1	Presentació	Comprova que E la font.
	2.1	Presentació	Inicialitza i mante el conjunt SIdParticipants
		Domini	Buscar les dades.
ObteNoWrittenBy	1.1	Presentació	Comprova que E la font
	2.1	Presentació	Inicialitza i mante el conjunt SIdParticipants
		Domini	Buscar les dades.
AltaReferences	1.1	Presentació	Només permet seleccionar els del conjunt.
	2.1	Presentació	Comprova q E les fonts
	2.2	Domini	Crea les instancies i les relacions.
			Comprova l'existència i enregistrament de dades
AltaWrittenBy	1.1	Presentació	
	2.1	Presentació	Comprova que E la font i el participant.
	2.2	Domini	Crea les instancies i les relacions.
			Comprova l'existència i enregistrament de dades
Cas D'ús Alta Evaluate-Item			

5.4. Especificació de cada capa

En aquest apartat es mira de presentar l'etapa de disseny que presenta l'especificació de cada capa, segons l'Enginyeria del Software. Realitzant aquesta etapa s'identifiquen les operacions de cada capa, on es veu la interacció entre les capes durant la realització d'un cas d'ús. En aquesta tasca es considera cada capa com una caixa negra, a la vegada que es presenta l'especificació en funció de cada una de les operacions.

Igual que abans, es presenten un parell d'exemples del que es l'especificació de cada capa. Els contractes com em dit abans serien els que es troben en l'apartat de l'especificació.

Exemple 1: Cas d'ús: Alta Participant

Capa Presentació	Capa Domini
Sreliabilities:Set(String), STypePerson:Set(String), STypePersonNo(String), SRols(String+String), SOrganitzacions(String)	IdParticipant:Int.
Constructor()	obteReliabilities(OUT:Set(STScore)) (2.1)
PremaSeguent()	altaParticipant(IN: name,reliability,descripció): Bool (1.1)(2.1)
PremaSeguent()	altaPerson(dni, city, email): Bool (2.1)(2.2) obteTypesPerson(OUT:StypePerson, STypePersonNo) (2.1) obteRols(OUT:SRols+SOrganitzacions) (2.1)
PremaAddType()	addType(typePerson, OUT:STypePerson, StypePersonNo): Bool (2.1) és controlat per la capa presentació. (2.2)
PremaAddRol()	addRol(rol, organization, OUT: SRols+SOrganitzacions): Bool (2.1) (2.2)

*ara correspondria la alta font, alta EvaluateItem,.. que són casos d'ús.

Exemple 2: Cas d'ús: Alta Font d'Informació

Capa Presentació	Capa Domin
SReliabilities:Set(String). STypeSource:Set(String), SSubTypeSource:Set(String), SFormats:Set(String), SAudiencesType:Set(String), SReferencesOn i To:Set(String) SnoReferencesOn i To:Set(String) SIdParticipants:Set(String) EtypeSource Efont, EAudienceType,	IdInformationSource, typeSource:TypeSource, subTypeSource:SubTypeSource.
Constructor()	obteReliabilities(OUT:Set(STScore)) (2.1)
PremaSeguent()	altaFontInformacio(IN: name,version,reliability, data,cost,description,location,descriptionType) (1.1)(2.1) obteTypesSource(OUT:STypesSource) (2.1)
SeleccioTypeSource()	obteSubTypesource(OUT:SSubTypeSource) (2.1)
PremaSeguent()	altaInformationSource(reliabiliy,typeSource,subTypeSource) (2.1)(2.2) controlat per la capa presentació. (2.2)(2.3) obteFormats(OUT:SFormats) (2.1) controlat per la capa presentació (2.2) obteAudiciencies(OUT:SAudiciencies) (2.1) controlat per la capa presentació (2.2) obteAudicienciesType(OUT:SAudicienciesType) (2.1) . obteReferences(OUT:SReferencesOn, SreferencesTo) obteNoReferencesOn i To(OUT:SReferencesOn i To) obteWrittenBy(OUT:SIdParticipants) obteNoWrittenBy(OUT:SIdParticipants) (2.1) controlat per la capa presentació (2.2)
PremaAddFormat()	addFormat(idFormat, OUT:SFormats) (2.1) controlat per la capa presentació (2.2)(2.3)
PremaAddAudience()	addAudience(idAudienceType,name) (2.1) controlat per la capa presentació (2.2)(2.3)
PremaAddReference()	AltaReferences(OUT:SReferencesOn,SReferencesTo) (1.1)(2.1) controlat per la capa presentació (2.2)
PremaAddWrittenBy()	AltaWrittenBy(OUT:SIdParticipant) (1.1)(2.1) controlat per la capa presentació (2.2)

*ara correspondria la alta font, alta EvaluateItem,.. que són casos d'ús.

5.5. Diagrames de seqüència

Arribat el moment en que es disposa de l'especificació de cada capa, es dissenya cada una de les capes independentment: de dalt a baix per si detectem errors o mancances, sempre seguim els principis del disseny per contracte, aplicant patrons de disseny a cada capa. Tenint com a objectiu definir la interacció entre les classes d'objectes per respondre als esdeveniments.

A continuació es mostren un parell d'exemples de diagrames de seqüència de la capa de domini, que mostren la realització d'un esdeveniment, formada per les operacions ja definides. Pel que fa a la capa de presentació es troba a l'apartat següent.

En aquests diagrames es pot observar l'aplicació d'alguns dels patrons de disseny. Cal esmentar que sorgeixen noves classes degut a l'aplicació de patrons, els quals s'han donat a l'assignatura d'Enginyeria al Software de la carrera d'Informàtica. Alguns dels exemples de l'aplicació dels patrons es troben en :

Patró Controlador, per cas d'ús.

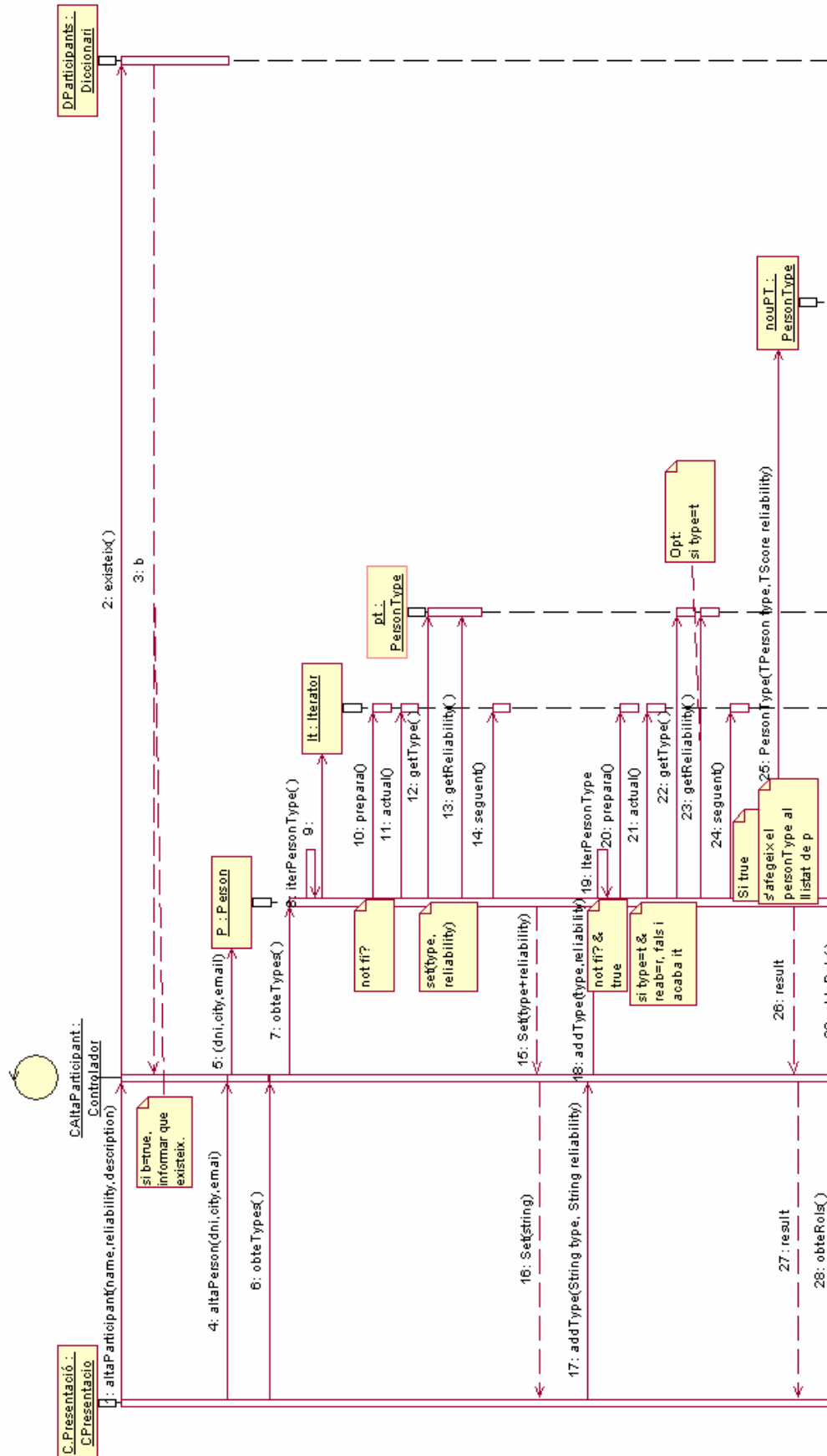
Patró Iterador en:

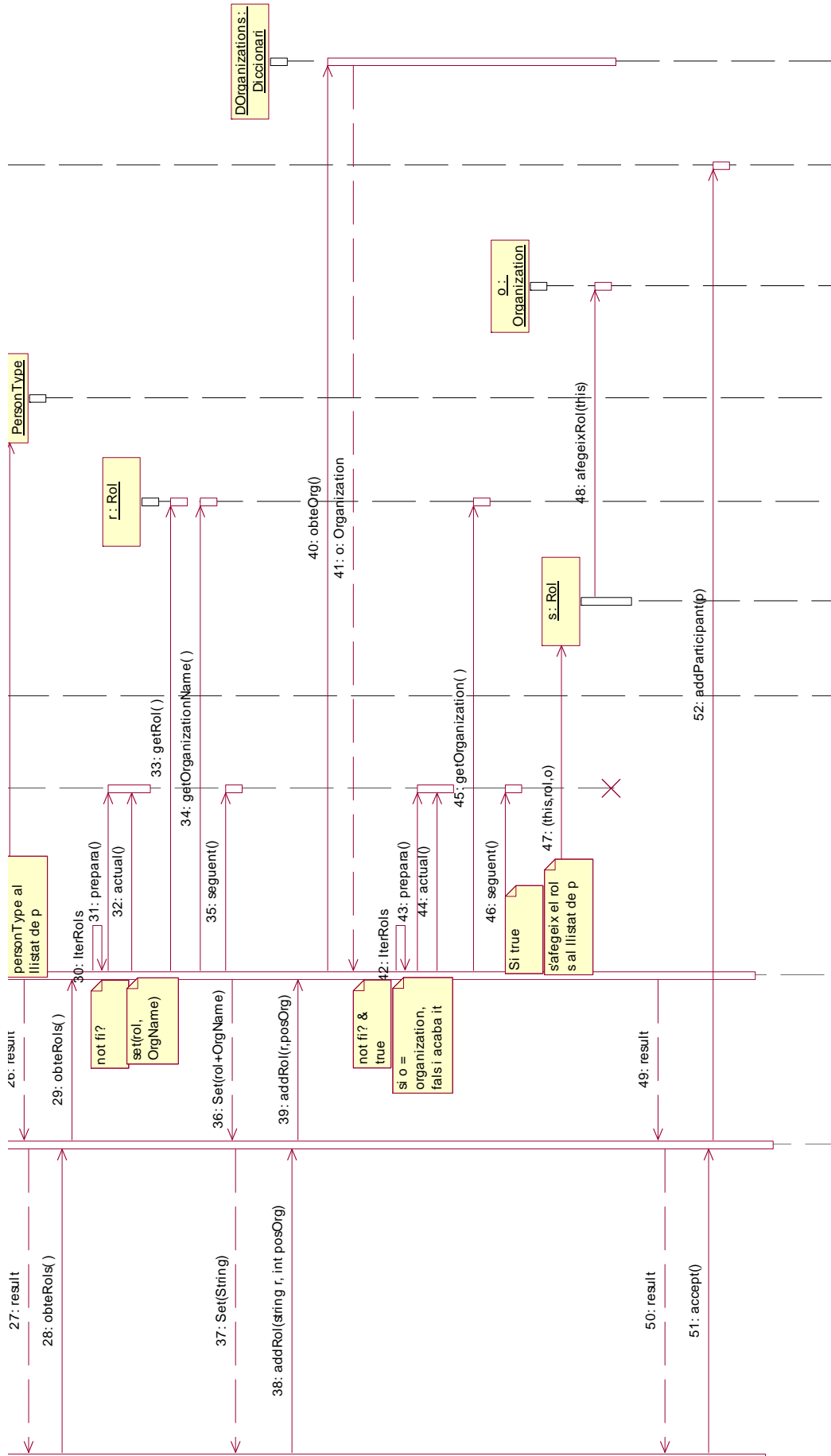
- de Person sobre PersonType, Rols.
- De Participant sobre InformationSourceOpinion i ParticipantOpinion.
- De Participant sobre informationSource.
- de InformationSource sobre TaxonoyNode, Format, Language, Audience, participants "writes", EvaluatItem, Related-to, Depends On, References.

Patró Diccionari per Participants i un altre per InformationSource.

Patrons d'assignació de responsabilitats, com patró Creador.

Exemple 1: Cas d'ús: Alta Participant





5.6. Disseny de la interfície

En aquest apartat, es presenta el disseny de la capa de presentació, concretament, es comenten el disseny Intern i Extern de la capa, que es realitzen paral·lelament. Finalment, es mostra els Mapes Navegacionals.

Cal esmentar que s'apliquen també patrons de disseny, per al disseny intern són molt semblants a la metodologia seguida en el disseny de la capa de domini, en canvi, per al disseny extern, s'apliquen altres com són la navegació per pestanyes, patrons de disposició, de selecció i guia (protecció, avís, desfer).

5.6.1 Disseny Extern

En el disseny extern ens interessa més saber com interactua el Usuari amb l'aplicació, és a dir, mira de dissenyar els elements, diguem-ne tangibles, que l'usuari veu, sent i toca a l'interaccionar amb el sistema. Donant com a resultat el disseny d'una interfície gràfica d'usuari (GUI).

Es defineixen mecanismes d'interacció, mitjançant el ratolí, el teclat,.. i mecanismes de presentació de informació, com es la visualització de les dades mitjançant les finestres.

S'han aplicat tres regles, considerades d'or, pel disseny de la interfície d'usuari:

- L'usuari té el control.
- Minimitzar memorització.
- Mantenir la interfície consistent.

Tenint sempre en consideració factors humans, socials o culturals.

Es podria presentar les pantalles dissenyades en el disseny extern, però per no estendre excessivament el projecte, es presenten més endavant el que serien els Mapes Navegacionals, en l'apartat Story Board.

5.6.2 Disseny Intern

En el disseny Intern, es dissenya la capa de presentació internament, dissenyant: com rep els esdeveniments de presentació, la gestió de la interacció amb l'usuari, de la comunicació amb la capa de domini, i presentació de les dades (rebudes des de la capa domini) en el format establert en el disseny extern.

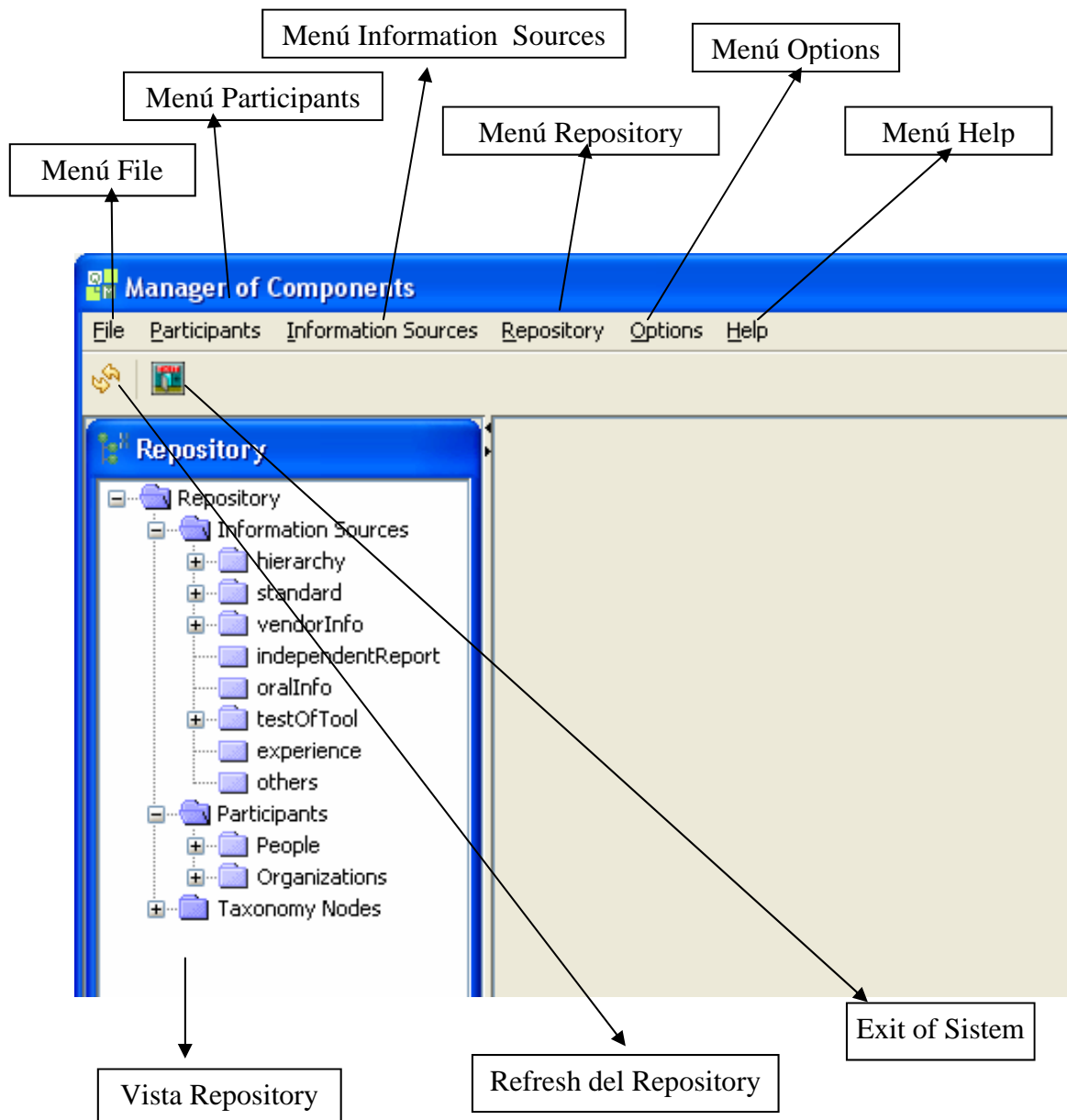
En el disseny intern es mira elements com a Objectes, el disseny de la interfície fent servir una biblioteca, concretament com fem servir Java, es treballa amb la biblioteca JFC/Swing.

5.7. Story Board

Arribats a aquest punt, es dona a conèixer les principals navegacions entre les principals pantalles del sistema dissenyat.

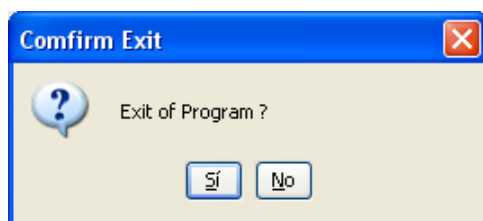
5.7.1 Mapes Navegacionals

5.7.1.1 Pantalla Principal

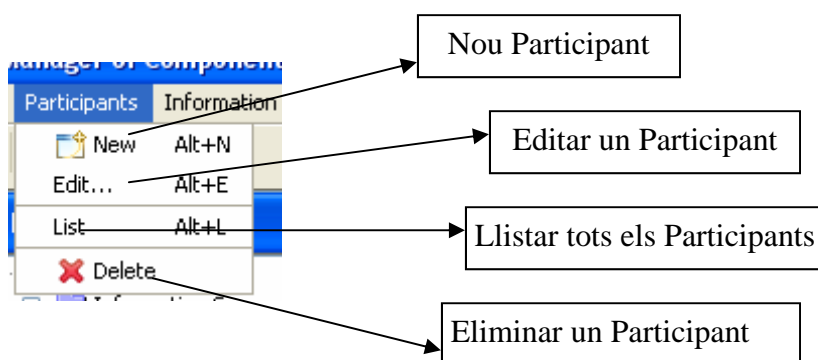


Menú File

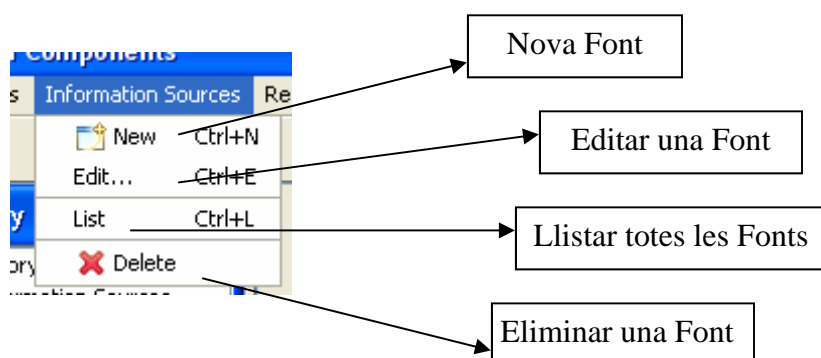
Des d'aquest menú o amb l'accés directe de la barra "ToolBar", igual que amb una combinació de tecles, és pot surti de l'aplicació. Sempre amb una confirmació.



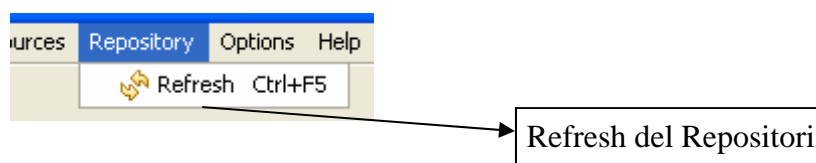
Menú Participants



Menú Information Sources

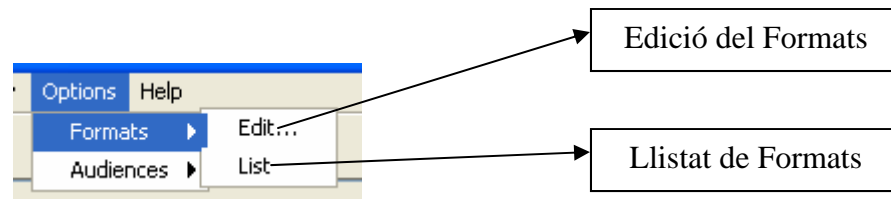


Menú Repository



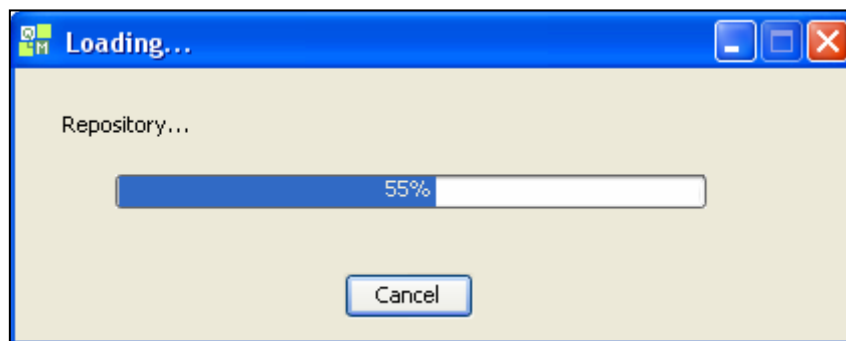
Aquesta acció també es pot realitzar amb el accés directe de la barra "ToolBar" o amb una combinació de tecles.

Menú Options



El que passa amb audiències es semblant al de formats.

5.7.1.2 Barra de Progrés



Al arrancar l'aplicació, es mostra una barra de progrés, per poder observar com es va carregant les diferents dades a memòria principal.

5.7.1.3 Participants

La vista de un nou participant es molt semblant a la de edició, per no repetir sols es presenten les navegacions d'edició. Es poden veure les vistes de les pestanyes més importants.

Edit Participant ISO

Evaluate-Items | Evaluate-Participants | Evaluate Participant To

Basic Info | Organization | Writes

Name: ISO

Reliability: veryHigh

Description: rganization for Standarization

Type: Person Organization

Accept Cancel See altogether

Edit Participant ISO

Evaluate-Items | Evaluate-Participants | Evaluate Participant To

Basic Info | Organization | Writes

C.I.F.:

Adress:

URL:

Sigles:

List of Types: standars - veryHigh

Add Delete

List of Rols:

Add Delete

Accept Cancel See altogether

Add Type

List of Types: academy, standars, company

Reliability: veryHigh, veryHigh, high, low, veryLow

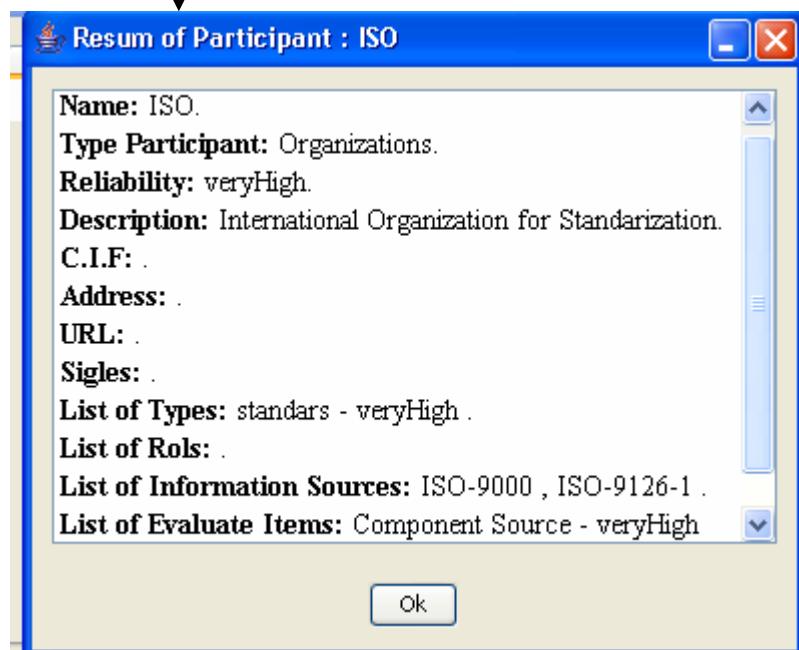
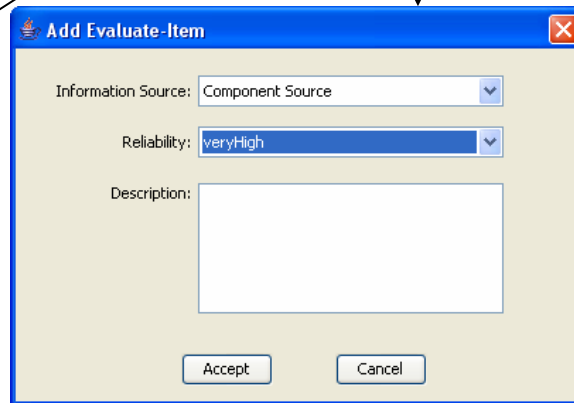
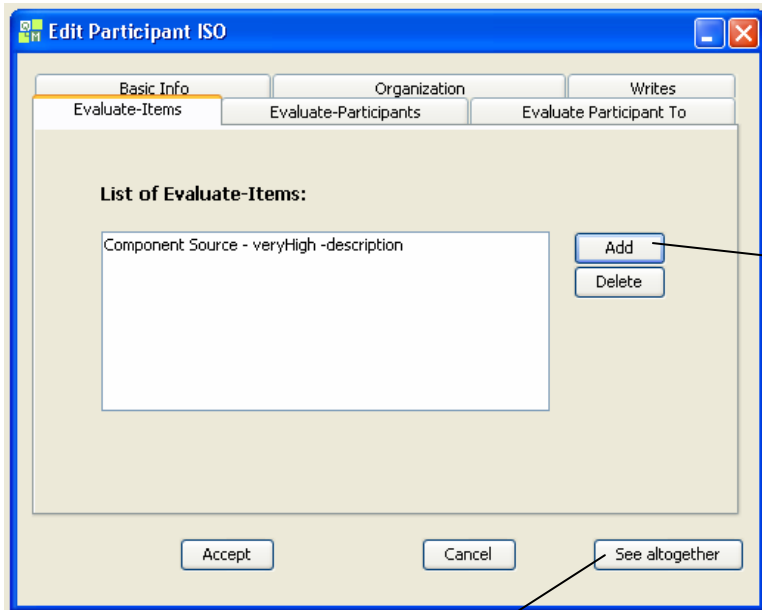
Accept Cancel

Add Rol

Person: Fernando Meseguer

Rol:

Accept Cancel



5.7.1.4 Fonts d'Informació

The screenshot shows the 'Edit Information Source ISO-9126-1' dialog box with the 'Basic Info' tab selected. The fields are as follows:

Name:	ISO-9126-1	Location:	www.iso.org
Version:	Ed.1	Description Type:	Natural Language
Reliability:	veryHigh	Scope Covered:	structured set of factors
Date:	01/01/2001		
Cost:	68,85		
Description:	cost for document. Standard about evaluation of Part 1: Quality Model		

Buttons at the bottom: Accept, Cancel, See altogether.

The screenshot shows the 'Edit Information Source ISO-9126-1' dialog box with the 'Format' tab selected. The 'List of Related To:' section is visible, showing a list of related sources and a list of related sources.

List of Related To:

- Component S...
- Gestio Software
- GNRT (guide to ne
- ISO-9000
- MSN Messenger
- OptiBase
- SIP
- Skype
- VoIP

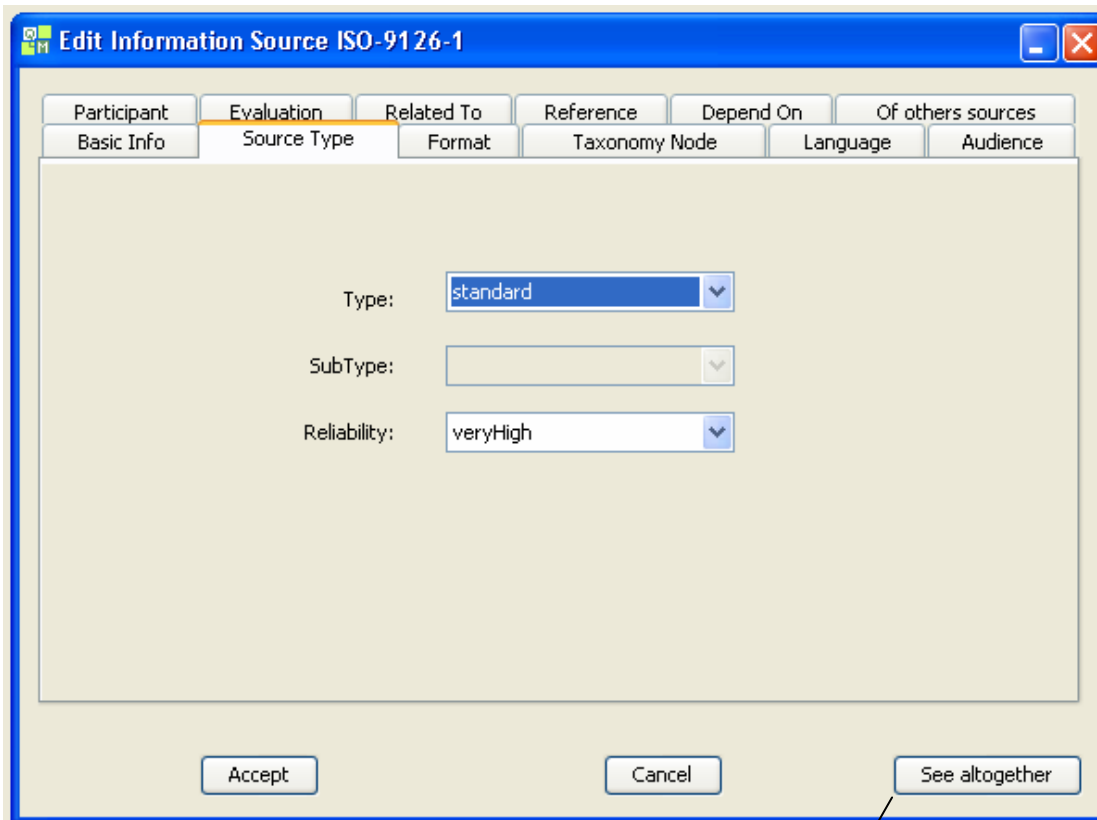
Buttons: Add, Delete

List of Related To:

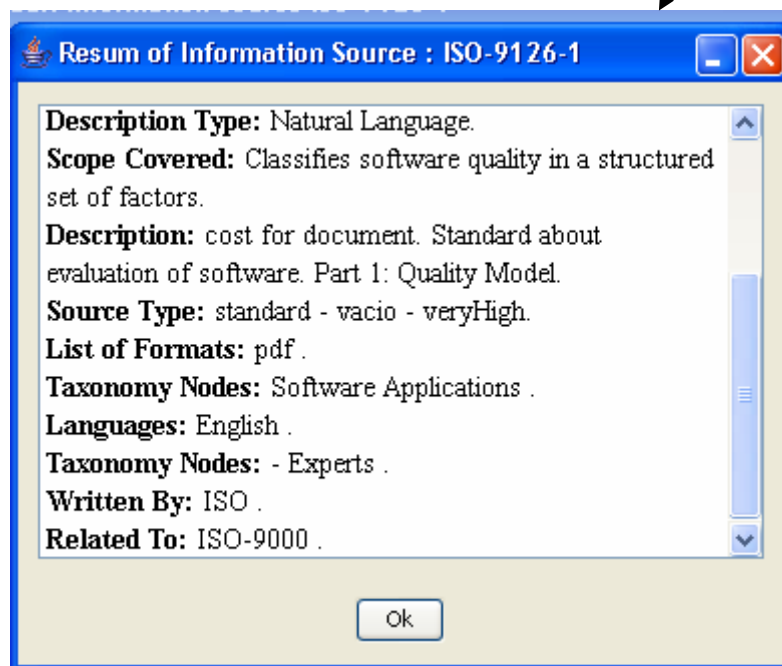
- ISO-9000

Buttons at the bottom: Accept, Cancel, See altogether.

La resta de pestanyes són semblants.

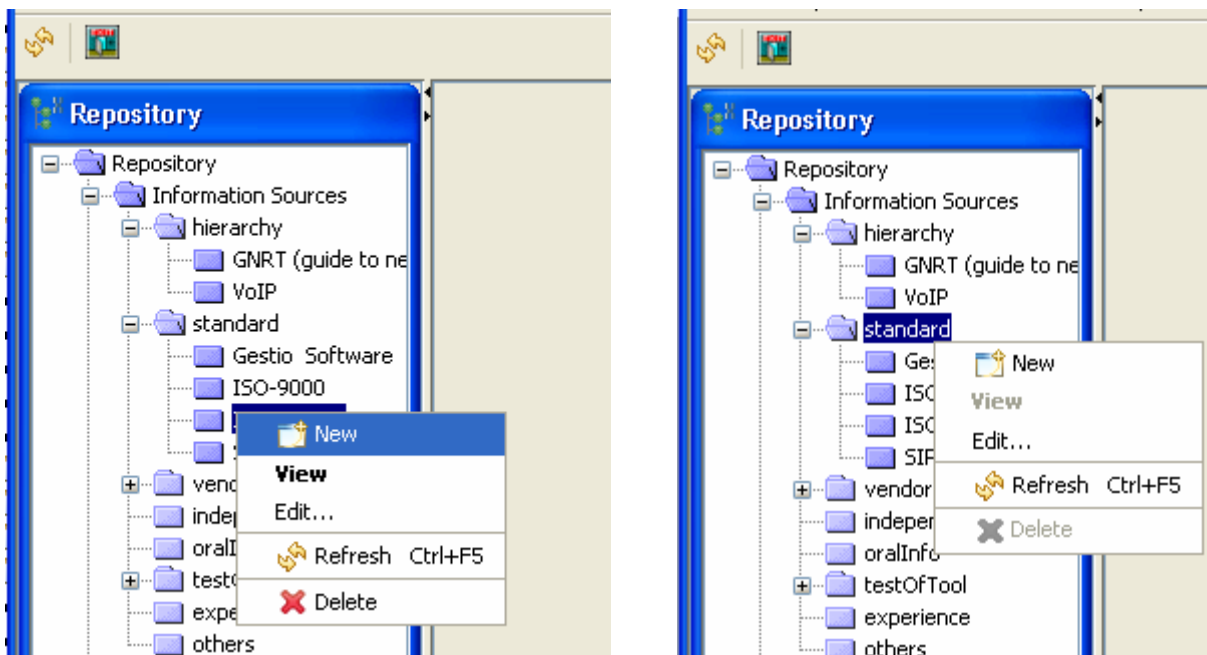


The screenshot shows a dialog box titled "Edit Information Source ISO-9126-1". It has a blue title bar with standard window controls. Below the title bar is a tabbed interface with the following tabs: "Participant", "Evaluation" (selected), "Related To", "Reference", "Depend On", and "Of others sources". Under the "Evaluation" tab, there are sub-tabs: "Basic Info", "Source Type", "Format", "Taxonomy Node", "Language", and "Audience". The main area contains three dropdown menus: "Type" set to "standard", "SubType" (empty), and "Reliability" set to "veryHigh". At the bottom, there are three buttons: "Accept", "Cancel", and "See altogether".



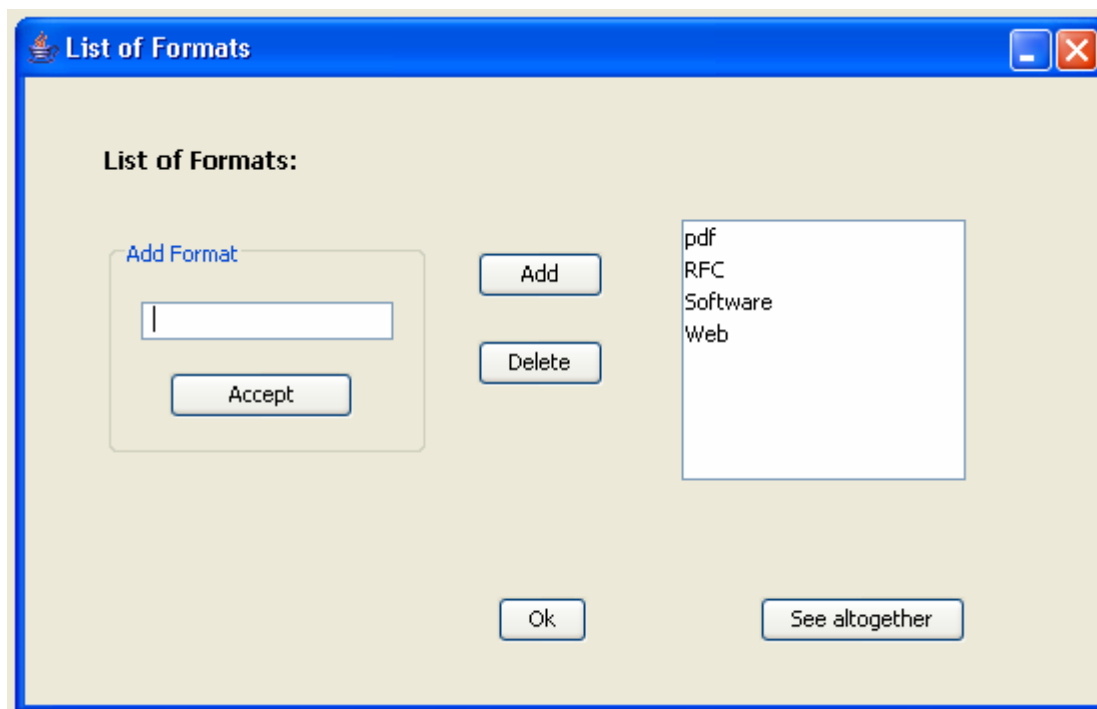
The screenshot shows a dialog box titled "Resum of Information Source : ISO-9126-1". It has a blue title bar with standard window controls. The main area contains a list of key-value pairs: "Description Type: Natural Language.", "Scope Covered: Classifies software quality in a structured set of factors.", "Description: cost for document. Standard about evaluation of software. Part 1: Quality Model.", "Source Type: standard - vacio - veryHigh.", "List of Formats: pdf .", "Taxonomy Nodes: Software Applications .", "Languages: English .", "Taxonomy Nodes: - Experts .", "Written By: ISO .", and "Related To: ISO-9000 .". At the bottom, there is an "Ok" button.

5.7.1.5 Repositori



Des de el repositori podem accedir a un menú “pop-up”, i poder editar, veure un resum, fer un refresh del repositori, o inclús eliminar.

5.7.1.6 Formats



La vista Audiences és semblant.

5.8. Disseny de la BD

Arribats a aquest punt, toca parlar de la persistència de les dades. Com ja s'ha comentat en l'apartat de la tecnologia emprada, el Sistema Gestor de bases de dades és un sistema relacional, concretament el MySQL,

Per a la traducció de l'esquema conceptual orientat a objectes que hi ha fins al moment, cal realitzar un conjunt de tasques: Disseny Lògic de la base de dades, tractament de les restriccions d'integritat, tractament de la informació derivada, finalment el disseny de les operacions. Però algunes d'aquestes tasques són dependents de l'estratègia de gestió de la persistència seleccionada, però d'altres no, com la normalització (tasca ja realitzada en anterioritat sobre el model conceptual) i el disseny lògic, tasca que es mostra una mica més endavant.

Per aquest disseny de la base de dades, es pot analitzar, a més de les ja tractades en la normalització:

- les jerarquies d'especialització, on es manté totes les classes de la herència,
- aparició d'identificadors interns, al diagrama es pot observar amb PK els *Primary Key*, amb PK2 els que són Primary Key i a la vegada Foreign Key, i amb FK els que únicament són Foreign Key.

Com ja s'havia comentat en l'apartat de estratègia de la persistència, es fa servir, la generació automàtica, la qual proporciona una persistència transparent al dissenyador, és a dir, una traducció automàtica per emmagatzemar les dades a memòria secundària. Per altra banda, aquesta tècnica requereix l'especificació de la correspondència entre el diagrama de classes i l'esquema de la BD; en l'apartat següent de la implementació es mostren alguns exemples d'aquesta especificació.

5.7.1 Traductor de Dades

Aquest punt esta dissenyat mitjançant el patró “*Traductor de Dades*”. El traductor de dades es troba en la capa de dades, entre el SGBD i la capa de domini, on es troben els objectes. Així el traductor s’assabenta automàticament de les modificacions fetes als objectes, en memòria principal, i les trasllada a les taules, en memòria secundària.

L’eina utilitzada és el *Hibernate*, amb el qual es treballa mitjançant sessions. La sessió manté la col·lecció d’objectes carregats a memòria, es podria veure com un diccionari genèric de les dades persistents.

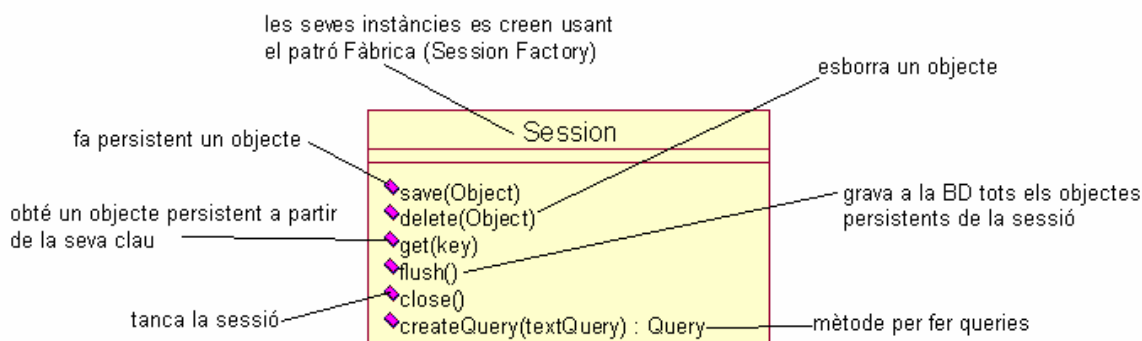


Fig. 40: Classe Session de Hibernate.

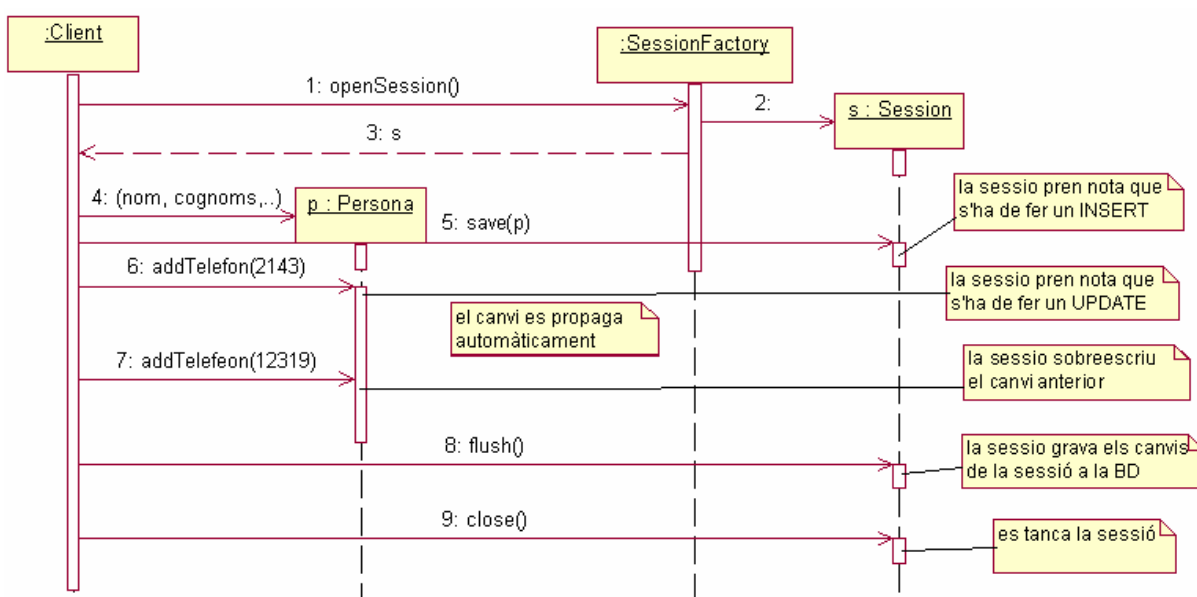


Fig. 41: Exemple de Generació automàtica amb Hibernate.

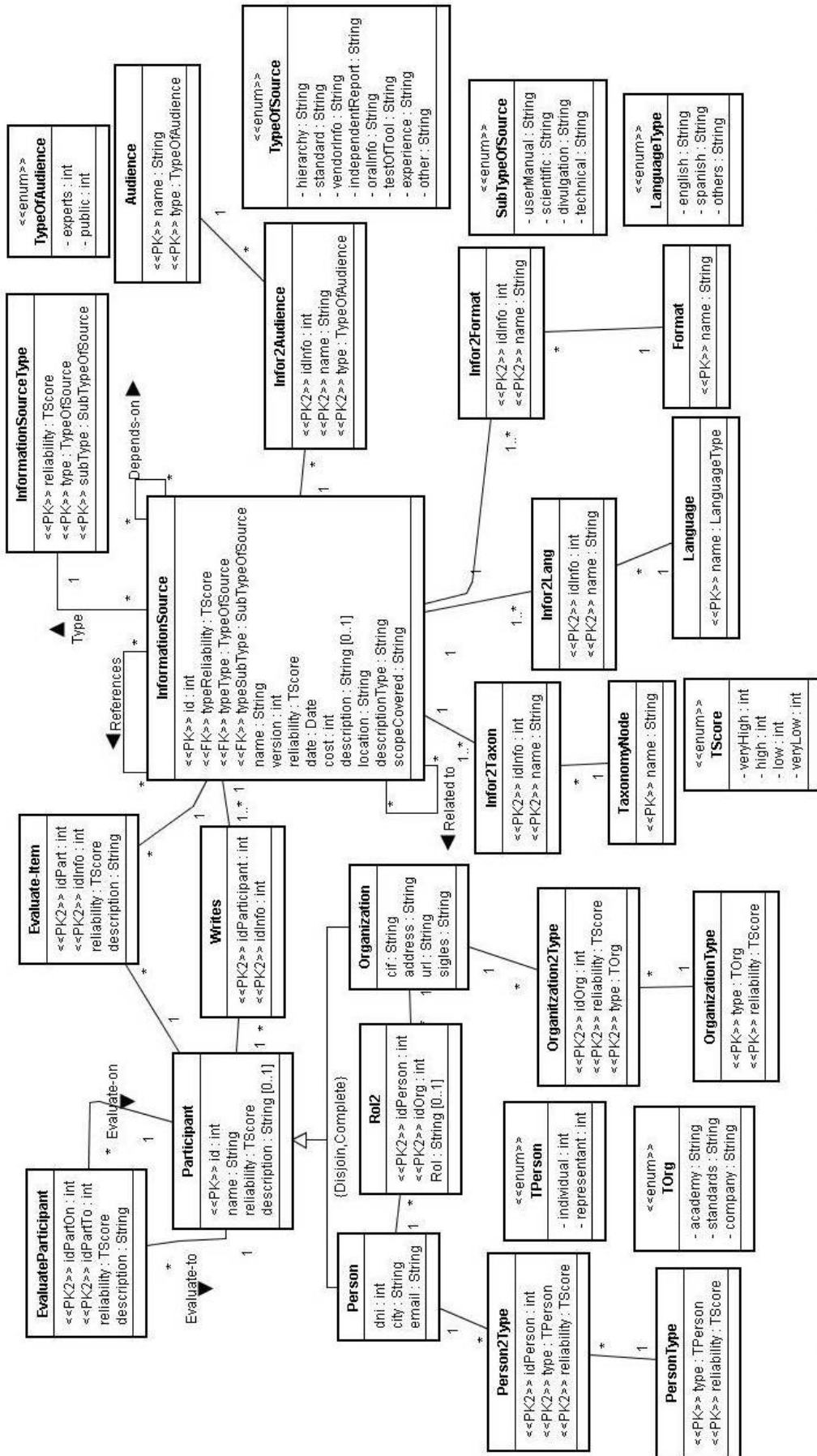


Fig. 42: Diagrama del Model de dades.

6. Implementació i Proves

Un cop es té tota l'eina software dissenyada i aprovada pel client, es passa al que es la implementació de la mateixa.

Per lo que en aquest capítol es presenten els aspectes més generals del que ha segut la implementació, descrivint les eines, tant hardware com software, utilitzades per dur a cap el projecte. A més, també es mostren alguns aspectes específics a remarcar del desenvolupament d'aquest, .

6.1. Aspectes generals de la implementació

6.1.1 Eines i Hardware utilitzats

Una de les qüestions que condiciona la veracitat de les simulacions, sobre tot en els càlculs que es fan de temps, és amb quin hardware i software s'ha implementat i es fan les proves. Es per això que en aquest apartat es descriu el hardware i el software utilitzat.

Concretament, aquest projecte s'ha dut a terme en un ordinador amb les següents característiques:

- Processador: PC, Intel Pentium IV a 2,4 Ghz.
- Memòria: 256 Mbytes de RAM.
- Disc: 40 Gb de disc dur.

El software utilitzat en aquest treball ha estat:

- Sistema Operatiu: Microsoft Windows XP Professional.
- Entorn : Windows.
- Compilador:
 - Maquina Virtual Java.
 - Llenguatge de programació Java, versió JRE 1.5.0_09
- Eines utilitzades per al desenvolupament:
 - Eclipse 3.2.
 - Hibernate v. 3.2.1.
 - MySQL v. 4.1.9.

- Llibreries:
 - JDOM, per al tractament de fitxers XML.
 - mysql-connector-java-3.1.14, comunicador entre Java i MySQL, requerit per Hibernate.
 - antlr-2.7.6, cglib-2.1.3, commons-collections-2.1.1, commons-logging-1.0.4, jdbc2_0-stdext, jta, log4j-1.2.11, asm, ant-1.6.5, org-jdesktop-layout, httpunit, necessaris per la compilació i funcionament del Hibernate.
 - swing-lawout-1.0, per al disseny de les pantalles.

- Altres eines:
 - Microsoft Word, Microsoft Visio, Rational Rose Ent. Ed i Rational Unified Proces, i Jude Community.

6.1.2 Aspectes específics de la implementació ha destacar

Com aspectes específics de la implementació, es mira de destacar l'eina Hibernate, tant en els fitxers de configuració com en els necessaris per a la correspondència de les dades.

A continuació es troba el fitxer de la configuració de Hlberate del producte software creat. On les propietats són: 1.- direcció d'on es troba la BD amb el port i el seu nom; 2.- Username del usuari que es connecta (i si fora necessari password); 3.- Driver i Dialecte emprat (depèn de la BD); 5.- Classe de transacció; 6.- Treball per threads; 7.- si es desitja veure al log les consultes en SQL (comentat). Després es troba el mapejar dels fitxers que defineixen la correspondència de les classes a dades.

```

<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE hibernate-configuration PUBLIC
    "-//Hibernate/Hibernate Configuration DTD 3.0//EN"
    "http://hibernate.sourceforge.net/hibernate-configuration-
3.0.dtd">

<hibernate-configuration>
  <session-factory>
    <!--connexió BD en local, port 3306 i nomBD realtime -->
    <property name="hibernate.connection.url">
      jdbc:mysql://localhost:3306/realtime
    </property>
    <property name="hibernate.connection.username"> root </property>
    <property name="hibernate.connection.driver_class">
      org.gjt.mm.mysql.Driver
    </property>
    <property name="hibernate.dialect">
      org.hibernate.dialect.MySQLInnoDBDialect
    </property>
    <property name="transaction.factory_class">
      org.hibernate.transaction.JDBCTransactionFactory
    </property>
    <property name="current_session_context_class">thread</property>
    <!-- this will show us all sql statements-->
    <!--<property name="hibernate.show_sql">true</property-->

    <mapping
      resource="src/datos/dominio/participant/Participant.hbm.xml"/>
    <mapping
      resource="src/datos/dominio/participant/PersonType.hbm.xml"/>
    <mapping
      resource="src/datos/dominio/participant/OrganizationType.hbm.
xml"/>
    <mapping resource="src/datos/dominio/participant/Rol.hbm.xml"/>
    <mapping
      resource="src/datos/dominio/information/InformationSource.hbm
.xml"/>
    <mapping
      resource="src/datos/dominio/information/InformationSourceType
.hbm.xml"/>
    <mapping
      resource="src/datos/dominio/information/Format.hbm.xml"/>
    <mapping
      resource="src/datos/dominio/information/TaxonomyNode.hbm.xml"
/>
    <mapping
      resource="src/datos/dominio/information/Language.hbm.xml"/>
    <mapping
      resource="src/datos/dominio/information/Audience.hbm.xml"/>
    <mapping resource="src/datos/dominio/EvaluateItem.hbm.xml"/>
    <mapping
      resource="src/datos/dominio/participant/EvaluateParticipant.h
bm.xml"/>
  </session-factory>
</hibernate-configuration>

```

Fig. 43: Arxiu de configuració de Hibernate.

```

<?xml version="1.0"?>
<!DOCTYPE hibernate-mapping PUBLIC
    "-//Hibernate/Hibernate Mapping DTD 3.0//EN"
    "http://hibernate.sourceforge.net/hibernate-mapping-3.0.dtd">
<hibernate-mapping package="src.dominio.information">
  <class name="Language" table="language" >
    <id name="name"
        type="src.datos.dominio.enums.LanguageTypeUserType"
        length="20">
      <generator class="assigned"></generator>
    </id>
  </class>
</hibernate-mapping>

```

Fig. 44: Arxiu de correspondència de les dades amb la classe Language.

L'Hibernate al ser una eina que no s'havia fet servir mai, a donat peu a entrebancs, com a segut el reconèixer les classes *Enumeration* de Java. Però l'experiència a estat satisfactòria. És necessari comentar que com a títol personal l'eina Hibernate m'ha sorprès, al descobrir el seu potencial.

La implementació de l'eina segueix la següent distribució en paquets:

❖ **bbdd**

es troben els CREATES de la BD i taules, així com INSERTS per als jocs de prova.

❖ **images**

es troben les imatges necessàries per al producte software.

❖ **lib**

conté les llibreries que necessita l'aplicació.

❖ **src**

es troba els fitxers de configuració del hibernate, a més de la classe principal (Main).

• **src.interfaz**

Controlador de C. de Presentació més les vistes més generals.

○ **src.interfaz.vistes**

es troben les vistes dels formats i audiències.

○ **src.interfaz.vistes.information**

conté les vistes de la interfície relacionades amb les fonts.

- **src.interfaz.vistes.participant**
es troben les vistes de la interfície relacionades amb el paquet Participants.
- **src.dominio**
aquí es troben les classes de la C. de Domini (Controladors del Domini, i el Repositori, així com el comunicador al servidor de Taxonomies)
 - **src.dominio.enums**
es troben totes les classes que són enumeracions.
 - **src.dominio.information**
conté les classes de domini que representen informació del Paquet Fonts d'Informació.
 - **src.dominio.participant**
conté les classes de domini que representen informació del Paquet Participants.
- **src.datos**
aquí es troben els fitxers de correspondència amb les dades. I el xml de les Taxonomies obtingut del servidor.
 - **src.datos.dominio.enums**
fitxers .java que fan de comunicadors per als Enumeration.
 - **src.datos.dominio.information**
fitxers xml que fan la correspondència de les dades relacionades amb el Paquet Fonts d'Informació.
 - **src.datos.dominio.participant**
fitxers xml que mantenen la correspondència de les dades del Paquet Participants.
- **src.exeptions**
guarda les classes que hereten de la classe Exception.
- **src.common**
conté un fitxer amb variables globals, classes genèriques per al repositori.
 - **src.common.progresBar**
conté les classes per a la barra de progrés.

6.2. Proves

S'han dissenyat un conjunt de jocs de prova per poder comprovar que l'eina desenvolupada funciona com es d'esperar. Durant el procés d'implementació de cada una de les funcionalitats, s'han realitzat un joc de proves per assegurar que treballa correctament, acabant amb un joc de proves real, és a dir, amb un format per les dades obtingudes en el procés de recerca de fonts d'informació.

6.2.1 Interacció amb pantalles

- S'ha comprovat que el botó OK realitza l'operació correcta de cada pantalla.
- S'ha comprovat que el botó Cancel cancel·la la operació i tanca la pantalla.
- En l'alta d'un participant o InformationSource, s'ha comprovat que es els botons Back i Next controlin les responsabilitats assignades (com són les claus).
- En les edicions dels elements s'ha comprovat que es controla les restriccions.

6.2.2 Visualització de pantalles

- S'ha comprovat que les pantalles mantenen un ordre correcte, i que no pots passar a l'anterior sense respondre a la confirmació o diàleg superior.

6.2.3 Entrada de dades

- S'ha comprovat que en llocs clau i no s'entra text o espais indiqui error.
- S'ha comprovat que en llocs clau, si el text introduït existeix doni error, i si no es clau ho comunica per informar per evitar possibles repeticions.
- S'ha comprovat que en camps numèrics, únicament es poden introduir números.

6.2.4 Proves del sistema

- S'han realitzat un gran nombre de combinacions d'altres, baixes, modificacions, crear relacions, desfer relacions, eliminacions en cascada, refrescades de repositori... comprovant que es manté la informació coherent.
- Es comprova que es carrega i guarda les dades correctament.
- Finalment, s'ha comprovat que l'aplicació suporta bé les dades reals, cercades en una etapa anterior.

7. Planificació del Projecte

Per poder preveure un marge de reacció per als possibles entrebancs és essencial tenir una bona planificació del treball a realitzar en un projecte, poden així tenir un bon desenvolupament d'un projecte, mirant d'assegurar objectius claus com són: cost, temps, qualitat; a la vegada que es mira de facilitar l'estalvi de recursos. Es per tot això, que en aquest capítol es mira de mostrar el planning temporal que s'ha seguit en aquest projecte, així com possibles desviaments al llarg del treball realitzat, per poder garantir els resultats en el temps.

Primer de tot es fa menció de les etapes de desenvolupament segons el mètode *Rational Unified Process* (RUP). Després es comenta la planificació de treball per al projecte que ens ocupa mitjançant un desglossament d'aquesta en cada una de les fases i les modificacions que s'han degut de realitzar i quins problemes han ocasionat aquestes canvis.

7.1. Mètode RUP

A continuació es pot observar les etapes del desenvolupament del projecte segons el mètode RUP. Aquest mètode divideix el procés de desenvolupament d'un producte software en 9 etapes. En aquesta gràfica es relaciona aquestes etapes amb quatre fases en que es troba un projecte: concepció, elaboració, construcció i transició; encara que aquestes es subdivideixen en més fases (veure par inferior de la gràfica). Cadascuna de les fases mostra la proporció de treball que s'ha de dur a terme de cada una de les diferents etapes.

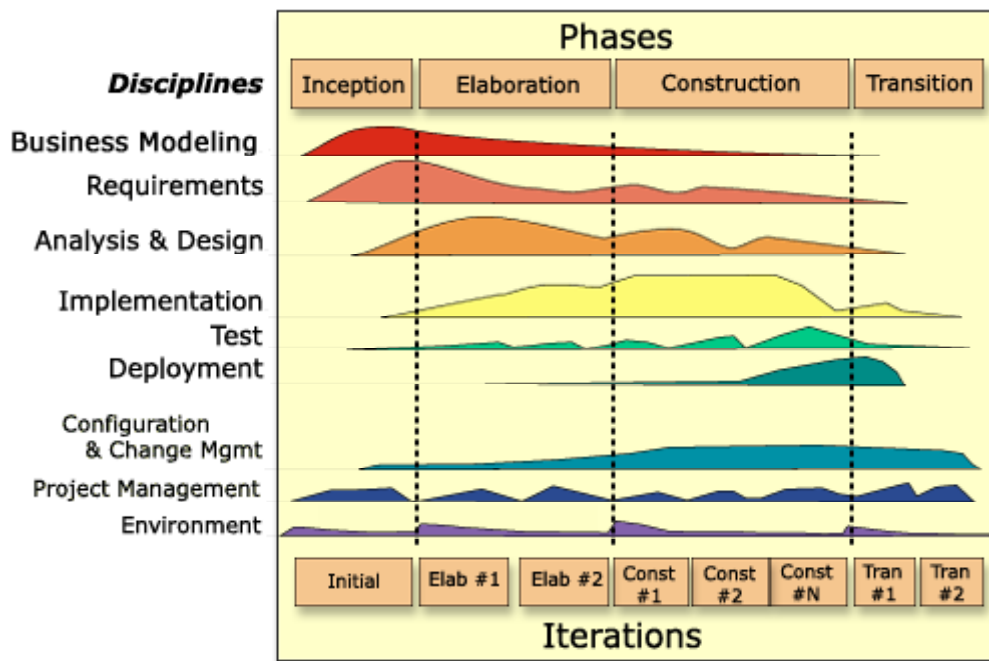


Fig. 44: Etapes del RUP.

7.2. Planificació de treball

Com qualsevol empresa humana, els projectes es troben sota un conjunt de restriccions: temps, cost i abast; també conegut com el triangle de la gestió de projectes, on cada vèrtex representa una de les restriccions. Cal tenir en conté que un vèrtex no es pot modificar sense impactar en els altres.

Temps

El temps emprat en dur a terme un projecte es finita, i la d'aquest es de 4 mesos, més concretament del La durada d'un projecte tots sabem que es finita, des de el 12 de Setembre fins al 29 de Gener.

Cost

El cost ve donat per múltiples variables, incloent costs de mà d'obra, de materials, infraestructures i utilitats. En el cas d'aquest projecte, com a mà d'obra sols es disposa de l'estudiant que el realitza i pel que fa als materials son eines lliures o proporcionades per la Universitat.

Abast

Els objectius que es persegueixen en aquest projecte ja han estat comentat en anterioritat, tant en la introducció els objectius i en l'anàlisi els requeriments específics.

Amb la finalitat d'acabar el treball dins d'aquest termini, és necessari tenir ben planificat el temps per cada una de les tasques a realitzar, mantenir un treball constant, a la vegada que es fa un control de manera periòdica, podria ser setmanalment, del treball realitzat en comparança amb el planificat.

7.2.1 Planificació de cada fase RUP

En aquest punt, es troba la planificació del treball pensada al inici del projecte per al seu desenvolupament, es presenta segons les fases de del mètode RUP, per cada una d'aquestes fases es pot observar l'etapa del projecte planificada amb el seu nombre d'hores pensades per realitzar-ho, i si ha sofert una desviació es troba les hores reals entre parèntesis. Encara, que primer de tot es mostra en un petit resum, fent una comparativa per cada una de les etapes del mètode RUP, de la planificació feta abans del projecte i de la planificació real.

Etapas del Mètode RUP		Hores de cada Etapa	
		Plan	Real
Concepció		134	136
Elaboració	Elabor. 1	151	152
	Elabor. 2	225	266
Construcció		255	286
Total d'hores		865	940

7.2.1.1 Fase de concepció

DISCIPLINA	Fase Concepció	Hores
Anàlisi del Problema i dels Requeriments.	Anàlisi del Problema	35 (45)
	Anàlisi de Requeriments	30 (40)
Especificació	Model Casos d'ús	25
	Model Conceptual	15
Disseny		
Implementació i Proves		
Documentació	Redacció	24 (6)
Gestions	Reunions amb el Client/Director	4
	Inscripció del Projecte	1

Total hores: 136

El que és la primera fase, de concepció, a segut realitzada amb èxit, exceptuant que el temps establert per l'anàlisi del problema i de requeriments ha sofert retards, el que ha impedit el inici esperat en la redacció de la memòria.

7.2.1.2 Fase 1 d'elaboració

DISCIPLINA	Fase Concepció	Hores
Anàlisi del Problema i dels Requeriments.		
Especificació	Model Casos d'ús	8
	Model Conceptual	8
	Model Comportament	25
Disseny	D. Arquitectura	10
	D. Interfície	15
	Normalització	15
	Assignació Responsabilitats	15
	D. Base de Dades	8
Implementació i Proves	Aprenentatge Hibernate	24 (35)
	Codificació	80 (70)
	Proves	10
Documentació	Redacció	24
Gestions	Reunions amb el Client/Director	4
	Realitzar i entregar Informe	5

Total hores: 252

En aquesta segona fase, s'havia previst un temps per a poder dependre l'eina Hibernate, però al ser una eina nova per al projectista (no vista mai), cosa que a provocat un nou retard en la planificació, ha hagut un temps de bloqueig amb el tractament amb Hibernate, al treballar-ho en la implementació, cosa que ha impedit codificar les funcionalitats que s'esperaven en aquesta primera part de la fase d'elaboració. A la vegada, es sumava les desviacions provocades en la fase anterior.

7.2.1.3 Fase 2 d'elaboració

DISCIPLINA	Fase Concepció	Hores
Anàlisi del Problema i dels Requeriments.		
Especificació		
Disseny	D. Interfície	8
	Normalització	21
	Assignació Responsabilitats	8 (2)
	Especificació de cada capa	20
	D. Base de Dades	20
	Story Board	(10) 15
Implementació i Proves	Codificació	88 (130)
	Proves	10
Documentació	Redacció	32
Gestions	Reunions amb el Client/Director	4

Total hores: 266

Un cop resolt els problemes trobats amb l'Hibernate, s'ha pogut pujar el rendiment en la codificació, cosa que es tenia parada en gran part, per causa del bloqueig. Les altres desviacions en aquesta segona part de la fase d'elaboració no provoquen cap tipus de retard, ja que, dient-ho d'alguna manera, s'anul·len d'una a l'altra.

7.2.1.4 Fase de Construcció

DISCIPLINA	Fase Concepció	Hores
Anàlisi del Problema i dels Requeriments.		
Especificació		
Disseny		
Implementació i Proves	Codificació	150 (175)
	Proves	8
Documentació	Redacció	80 (86)
Gestions	Reunions amb el Client/Director	3
	Preparació del CD	8
	Enquadernació i entrega de Memòria	2
	Preparació+Assaig+Presentació	4

Total hores: 286

En l'última fase, la fase d'acabament de la memòria, s'ha incrementat ja que era important poder tenir constància de tot, a la mateixa vegada, la implementació encara es veia retardada pel bloqueig amb l'hibernate, per lo que s'ha tingut que dedicar una mica més de temps per poder implementar totes les funcionalitats.

8. Estudi Econòmic

Encara que es tracti d'un treball final de carrera, es sempre força interessant mostrar una valoració econòmica del projecte, per tal de poder tindre una visió diferent del treball realitzat o com es la seva situació en el mercat actual.

En aquest capítol es mira de calcular el preu del producte final, amb el seu desglossament de les hores dedicades a cada una de les tasques a realitzar. Cal dir que el preu, en aquest projecte, és bàsicament les hores invertides en el seu desenvolupament, més el cost del hardware aportat, a la vegada que hi ha altres variables com són: manteniment, assistència, garantia, etc.

Primer es presenten els costos dels recursos humans utilitzats en les diferents parts de la realització del projecte. Després, es donen a conèixer altres costos. Per acabar amb el cost final del treball.

8.1 Càlcul de costos

8.1.1 Recursos humans

Fase	Rol	Preu	Hores	Total
Requeriments	<i>Analista</i>	43	85	3.655
Anàlisi	<i>Analista</i>	43	219	9.417
Disseny	<i>Dissenyador</i>			
Implementació	<i>Programador</i>	29	453	13.137
Proves	<i>Programador</i>			
Documentació	<i>Enginyer</i>	43	183	7.869

Total Cost: 34.078 euros

Cal dir que els preus han estat aproximats als indicats pel Institut d'Estadística, de la Consergeria d'Economia e Innovació Tecnològica de la Comunitat de Madrid

8.1.2 Altres

Una de les coses a tenir en consideració en els costos, es el n^o d'hores d'electricitat, la variable quantitat depèn dels materials necessitats, en aquest cas un ordinador personal, on s'ha desenvolupat tot el treball. També es té en consideració el % per al impost especial sobre la electricitat.

$$\text{Cost: } 0,200 \text{ (kW)} * 940 \text{ (h)} * 1,46 \text{ 1129 €/KW} * 1,05113 \text{ (IEE)} * 4,864\% * 1,16 \text{ (IVA)} = 16,29 \text{ € de consum}$$

El càlcul es troba realitzat en tarifa domèstica (2.0).

8.1.3 Cost final

El cost final per una empresa seria de : 34.094,29 €

Si el preu fora per venda distribuïda, per exemple, per a una demanda de 100 usuaris , seria de 340€.

9. Conclusions

S'ha realitzat la recerca i estudi d'informació rellevant d'un conjunt de components software del mercat de components. Determinant la informació "valuable" de les fonts d'Informació; definint els atributs que podran permetre la estructuració; estudiant i determinat l'estructura per poder classificar les diferents font; classificant els diferents proveïdors, a la vegada que, la relació entre les fonts, com de les respectives valoracions.

S'han estudia els problemes que sorgeixen de la classificació, per poder crear una eina que doni suport a l'anàlisi de dominis, A partir dels quals s'ha realitzat un anàlisi i s'han especificat els requeriments tant generals com específics, tant funcionals com no funcionals, que ha de satisfer el sistema.

Un cop realitzat l'anàlisi, s'ha respost al *Que ha de fer , el què farà, i com ho farà*, és a dir, s'ha desenvolupat l'eina segons les etapes i tècniques apreses en les diferents assignatures realitzades en els cinc anys de carrera: especificant els requeriments assequibles, mitjançant tècniques apreses com el llenguatge UML, i artefactes com els models de casos d'ús, model conceptual i

de comportament, diagrames de seqüència i contractes de les operacions; dissenyant el software a desenvolupar, a través de les etapes del disseny (arquitectura, normalització del esquema conceptual i contractes, assignació a capes, especificació de cada capa, disseny intern i extern de la interfície, i definit les dades que són necessàries guardar en un suport no volàtil de forma eficient (disseny de la base de dades); a la vegada, que s'ha implementat mitjançant la tecnologia establerta en el projecte, el software, i finalment, s'han realitzat la posterior comprovació de l'eina.

Amb aquest projecte s'ha intentant posar el granet d'arena al projecte UPIC, en l'objectiu concloent, d'obtenir una estratègia eficaç per la recerca d'informació de software en un anàlisi sòlid del domini per així poder establir una infraestructura reutilitzable de software.

Per finalitzar, esmentar que el participar en un projecte de final de carrera en el camp de la recerca i desenvolupament de software, partint de zero, a nivell personal a estat molt gratificant, ideal per poder aplicar tots els coneixem apresos durant els anys a la Universitat, a la vegada que s'aprenen altres, tant a nivell acadèmic com professional, en un projecte real, mantenint relació amb altra gent, perseguint uns objectius, i el fet d'estar sota la pressió del temps, a la vegada que es té una responsabilitat. Aconseguint que un tingué una nova visió.

Línies de continuació del Projecte

Aquest projecte, ha aportat noves contribucions, però a la vegada, al ser un projecte d'un cert nombre d'hores limitades pels crèdits matriculats, ha deixat noves línies per poder continuar en projectes futurs. En aquest apartat es destaquen les línies de continuació següents:

- Eina distribuïda: S'ha desenvolupat l'eina per treballar en un equip, ja que el fet de fer-ho distribuït implicava més temps i no es considerà oportú. Pel que, una línia seria el implementar l'eina de manera distribuïda i així permetre el treball simultani i des de diferents llocs.
- Accés restringit: La utilització de l'eina, en principi, s'ha dissenyat per ser utilitzada per un usuari expert, i per tant, el tindrà instal·lat a un lloc segur. Però no deixa de ser una bona millora, el tenir un control en l'accés a l'aplicació, és a dir, permetre un accés sols a personal autoritzat, mitjançant un sistema d'identificació.
- Repositori força versàtil: De moment, s'ha treballat el repositori per poder tindre un accés més ràpid a la informació que hi ha introduïda, amb un criteri de classificació establert. Encara que seria bo, poder donar-li més responsabilitat al repositori, permeten, per exemple, que es pugui mantenir uns criteris de classificació variables, com podria ser, en funció del valor d'algun atribut concret.

10. Bibliografia

Referències Bibliogràfiques i documentació en línea

Per dur a terme aquest projecte s'han consultat diferents referències bibliogràfiques, així com els següents documents i/o llocs en línea.

10.1 Llibres

- Bruce Eckel, *"Pienso en java"*, 2^a ed. Prentice Hall. (2002)
- Agustín, Froute, *"Java 2: Manual de Usuario y tutorial"*, 2^a ed. Ra-ma. (2000)
- Dolors Costal, Xavier Franch, M.Ribera Sancho, Ernest Teniente, *"Enginyeria del Software Especificació"*, 3^a ed. Edicions UPC. (2005).
- Erik T. Ray. Learning XML. O'reilly. (2001).
- Charles F. Goldfarb i Paul Prescod, *"XML Handbook"*. 4^a ed. Prentice Hall Ptr. (2002).
- Sebastián Ávila, *"Suport a la reenginyeria de processos basat en actors"*. (2006)

10.2 Material Docent

- Apunts de l'assignatura d'Enginyeria al Software I i II (ES1 i ES2).
- Apunts de l'assignatura de Disseny i Administració de Bases de Dades (DABD).
- Apunts de l'assignatura Projecte d'Enginyeria al Software i Bases de Dades (PSBD).
- Conceptes apresos de programació.

10.3 Articles

- [Ayala-Franch2006] Ayala, C., Franch, X., “*A Goal-Oriented Strategy for Supporting Commercial Off-The-Shelf Components Selection*” *9th International Conference on Software Reuse (ICSR) 2006*.
- [Ayala-Franch2007] Ayala, C., Franch, X. “*Acquisition and Analysis of Data and Information for Software Package Classification and Selection*”. *To be submitted*.
- [Carvalho-et-al2004] Carvalho, J.P., Franch, X., Quer, C., Torchiano, M. “*Characterization of a Taxonomy for Business Applications and the Relationship Among Them*”. *Proceedings of 3rd International Conference on COTS-Based Software Systems (ICCBSS), LNCS 2959, 2004*.
- [Ayala-et-al2005] Ayala, C., Botella, P., Franch, X., “*On Goal-Oriented COTS Taxonomies Construction*”. *4th ICCBSS. LNCS 3412, Bilbao, Spain; February 2005*.
- [Cechich-et-al2006] Cechich, A., Réquile-Romanczuk, A., Aguirre, J., Luzuriaga, J.M. “*Trends on COTS Component Identification and Retrieval*”. *In Proceedings of 5th International Conference on COTS-Based Software Systems (ICCBSS), IEEE Computer Society, 2006, pp 90-99*.

-
- [Carney-Long2000] Carney D., Long F.: “*What Do You Mean by COTS? Finally a Useful Answer*”. *IEEE Software*, 17 (2), March/April 2000.
 - [Meyers-Oberndorf2002] Craig Meyers, B., Oberndorf, P. “*Managing Software Acquisition*”. SEI Series in Software Engineering, 2002

10.4 Pàgines Web

- Java Sun. <http://java.sun.com>.
JFC/Swing <http://java.sun.com/docs/books/tutorial/uiswing/index.html>
- The XML standard. <http://www.w3.org>.
- Patrons. http://www.uie.com/articles/elements_of_a_design_pattern.
- MySQL. <http://www.mysql.com>
- Apache. <http://www.apache.org>
- Eclipse. <http://www.eclipse.org>.
- Hibernate <http://www.hibernate.org>.
 - Problems <http://www.hibernate.org/116.html>,
<http://www.hibernate.org/117.html>.
- Ant. <http://www.chuidiang.com/java/herramientas/ant.php>.
- Rational Rose, RUP
- Preus dels recursos Humans:
<http://www.madrid.org/iestadis/fijas/otros/servprec.htm>
- Comissió Nacional d'Energia <http://www.cne.es/cne/contenido.jsp>