

# PRÀCTICA

Entrega final

**Enginyeria del Coneixement i Sistemes Distribuïts  
Intel·ligents**

Quadrimestre de Primavera 2023/2024

03/06/2024

**Grup**

Adrià Barragán Redondo

Nil del Pozo Comas

Lluc Paret Aguer



UNIVERSITAT POLITÈCNICA  
DE CATALUNYA  
BARCELONATECH



# ÍNDEX

<b>1. Disseny detallat.....</b>	<b>3</b>
1.1. Servei comandes.....	3
1.2. Servei clients.....	3
1.3. Servei assignador transportista.....	3
1.4. Servei assignador de lots.....	4
1.5. Servei buscador.....	4
1.6. Servei retornador.....	4
1.7. Servei catàleg.....	4
1.8. Servei entrega.....	4
<b>2. Ontologia.....</b>	<b>6</b>
2.1. Acció.....	6
2.1.1. BuscarProductes.....	6
2.1.2. ComprarProductes.....	7
2.1.3. CobrarProductesVenedorExtern.....	7
2.1.4. ProcessarEnviament.....	7
2.1.5. RecomanaProducte.....	8
2.1.6. ValorarProducte.....	8
2.1.7. RetornarProducte.....	9
2.1.8. EditarProductesCatàleg.....	9
2.2. CentreLogístic.....	9
2.3. Client.....	10
2.4. Comanda.....	10
2.5. Lot.....	12
2.6. Producte.....	14
2.7. RespostaTemptativa.....	16
2.8. Restriccions.....	17
2.9. Transportista.....	17
2.10. VenedorExtern.....	18
<b>3. Descripció de la Implementació.....</b>	<b>19</b>
3.1. Eines.....	19
3.2. Agents externs.....	20
3.3. Accions.....	20
3.3.1. Buscar productes.....	20
3.3.2. Comprar productes.....	21
3.3.3. CobrarProductesVenedorExtern.....	21
3.3.4. ProcessarEnviament.....	21
3.3.5. Valorar productes.....	22
3.3.6. Retornar productes.....	23
3.3.7. Recomana productes.....	24
3.3.8. Afegir producte extern.....	24
<b>4. Tasques nivell avançat.....</b>	<b>26</b>

5. Tasques nota extra.....	27
6. Jocs de prova.....	28
7. Planificació.....	29
8. Evaluació de resultats.....	30
9. Consideracions finals.....	31

# 1. Disseny detallat

## 1.1. Servei comandes

Agent que s'encarrega d'acceptar les comandes encarregades per l'usuari i realitzar la localització dels productes de les comandes als centres logístics del sistema.

Aquest agent és centralitzat i es posa en marxa per servir compres dels usuaris, per tant, agrupa rols reactius.

Té totes les accions per rebre la petició d'una comanda amb els productes demanats i dur a terme la comunicació amb l'agent assignador lot per informar la nova comanda amb els productes demanats a tots els centres logístics. Finalment, cal informar la resposta temptativa a l'usuari.

Els productes que siguin de venedors externs seran passats al servei entrega perquè reclama el pagament i es notifica als venedors externs de la compra.

## 1.2. Servei clients

Agent que s'encarrega de la interacció directa amb el client mitjançant percepcions proactives. Agrupa els rols "Recomanador productes" i "Analista valoració".

Agrupa les accions amb les quals el sistema envia productes recomanats a l'usuari basat en l'historial de compres i les valoracions.

També realitza les accions necessàries per afegir valoracions als productes que han estat comprats.

## 1.3. Servei Centre Logístic.

L'Agent Servei Centre Logístic es troba distribuït per tots els centres logístics del sistema i aquest s'encarregarà de tota la part de gestió de lots i d'interacció amb els transportistes.

Aquest servei es posa en marxa quan rep una comanda del Servei Comandes, agafa els productes de la comanda i es ajunta en lots.

Llavors, cada centre logístic negociaria amb els transportistes per l'enviament d'un lot. Realitza la comunicació amb el servei entrega perquè informi els usuaris dels productes dins del lot la data d'entrega i amb quin transportista serà enviat i perquè comenci el procés de pagament.

## 1.4. Servei buscador

Agent que s'encarrega d'atendre a les búsquedes de productes que realitza l'usuari i retornar la informació necessària. Agrupa únicament un rol "Buscador", ja que es posa marxa moltes més vegades que altres rols com l'"Acceptador de comandes".

Agrupa les accions per fer la búsqueda en el catàleg sencer productes propis + externs processa els filtres que ha proporcionat l'usuari.

## 1.5. Servei retornador

Agent que s'encarrega d'atendre a les devolucions de comandes de l'usuari. Ha de processar la petició de devolució per part de l'usuari, calcular la resolució i informar l'usuari la forma de retornament d'aquesta. També ha de comunicar a l'agent "Servei entrega" que ha de realitzar una transacció a l'usuari per retornar-li els diners (aquest agent agrupa el rol Tresorer).

## 1.6. Servei catàleg

L'Agent Servei catàleg dona suport als venedors externs que volen fer actualitzacions als productes que tenen a la venda al nostre sistema. Aquest agent processa la petició de voler modificar, afegir o eliminar un producte al catàleg extern i modifica la base de dades de catàleg.

## 1.7. Servei entrega

L'Agent Servei entrega s'encarrega de totes les comunicacions que s'han de realitzar per tal de completar la compra d'un producte.

Agrupa les accions que processen el missatge del servei assignador transportista amb la data d'entrega i el transportista escollit i informará el client.

També aquest agent agrupa totes les accions del tresorer del sistema.

Primer es rep un missatge amb els productes de venedors externs i el client que els ha comprat perquè reclami els diners a l'usuari i després pagui al venedor extern.

Si el missatge que rep és procedent de l'assignador transportista aquest agent informará els usuaris que tinguin productes al lot que s'envia amb la data d'entrega i el transportista assignat. Posteriorment, notifica al tresorer perquè reclami el cobrament a l'usuari.

Finalment, si s'accepta el retorn d'un producte arriba un missatge amb l'usuari i la quantitat a retornar, es comunica amb el servei de pagament perquè es retornin els diners a l'usuari.



## 2. Ontologia

En aquest apartat definirem cada una de les classes de la ontologia. Aquest, és una versió més extensa que el pdf generat a partir de la ontologia i ens serveix per justificar l'ús de cada classe així com els atributs i relacions d'aquestes.

### 2.1. Acció

La classe Acció representa el conjunt d'accions que hauran de realitzar els diferents agents del sistema per poder arribar a complir amb tots els escenaris. A continuació s'explica en detall cada una de les subclasses que conformen el conjunt d'accions de l'aplicació.

#### 2.1.1. BuscarProductes

Aquesta acció és realitzada per l'agent Servei Buscador per donar resposta a la búsqueda d'un client. S'activarà sempre que el sistema rebí un conjunt de restriccions (veure més endavant 2.8) per part de l'**Agent Assistent**. Quan hagi rebut aquestes restriccions de compra, l'**Agent Servei Buscador** farà una búsqueda entre els productes (veure més endavant 2.6) oferts al catàleg que les compleixin i enviarà la informació dels productes resultants al mateix **Agent Assistent**.

#### 2.1.2. ComprarProductes

Aquesta acció correspon a la primera de l'escenari de Compra Producte i abarca, primerament, la creació de la comanda dels productes que ha enviat l'**Agent Assistent** i que ha obtingut de la búsqueda. L'acció **ComprarProductes** conté una subclasse anomenada **EnviarRespostaTemptativa** que representa l'acció d'enviar a l'**Agent Assistent** la factura de compra dels productes. Aquesta mena de factura preliminar s'anomena RespostaTemptativa (veure més endavant 2.7). Aquesta acció és executada pel **Servei Comandes**.

Finalment, aquest agent es posa en contacte amb el **Servei Centre Logístic** per enviar-li la informació de les comandes que s'han d'enviar. Aquest es troba distribuït als centres logístics i se li envia una comanda/subcomanda (veure més endavant 2.5). Distribuirà la comanda rebuda en lots depenent del pes i el nombre de productes. L'objectiu és tenir lots que puguin ser enviats per un transportista i continguin productes que van a la mateixa ciutat.

### 2.1.3. CobrarProductesVenedorExtern

Després de realitzar la compra de productes, si aquesta conté productes d'un venedor extern, l'**Agent Servei Comandes** es posa en contacte amb l'**Agent Servei Entrega** per gestionar el cobrament d'aquests productes. L'agent s'encarregarà de cobrar els productes a l'usuari amb el mitjà de pagament que ha especificat prèviament. A continuació s'executa l'acció subclasse **PagarVenedorExtern** que consisteix en contactar amb el servei de pagament per fer les ingressos al compte bancari dels venedors externs.

### 2.1.4. ProcessarEnviament

Aquesta acció representa tot el procés d'enviament dels lots a les destinacions dels productes que conté. S'inicia pel **Servei Centre Logístic** al moment de rebre la petició de distribució de comanda amb l'objectiu de trobat una empresa de transport que envii un lot.

El primer pas d'aquesta acció conforma una subclasse de **ProcessarEnviament** i s'anomena **EnviarCondicionsEnviament**. Consisteix en contactar amb el conjunt de transportistes (veure més endavant 2.9) informant de les condicions d'enviament del lot sol·licitat. S'espera una resposta de cada transportista amb un preu amb el que estan disposats a realitzar la entrega o una declinació.

A continuació trobem la subclasse **EnviarContraoferta** on s'inicia un procés de negociació amb cada transportista. L'agent **Servei Centre Logístic** enviarà una contraoferta amb un preu lleugerament inferior a la oferta mínima rebuda pels transportistes. Els transportistes s'espera que contestin acceptant la contraoferta o rebutjant-la i enviant un nou preu que estan disposats a cobrar per l'encàrrec. Assumirem que cap transportista amb el que negociem sortirà de la negociació si no ho ha fet al **EnviarCondicionsEnviament**.

A continuació, quan es decideix que una contraoferta d'un transportista és la correcta o acceptin la nostra es realitza una assignació. Aquesta acció representa una subclasse de la ontologia i s'anomena **AssignarTransportista**. També inclou el procés de relacionar el transportista amb el lot (assignar-lo) i informar al transportista que se li ha atorgat la entrega.

Finalment, trobem la subclasse **InformarEnviament**. En aquesta l'agent **Servei Centre Logístic** envia la informació d'assignació i data d'entrega al **Servei Entrega** que s'encarrega d'informar al client. Aquesta acció també conté una subclasse **CobrarProductes** que consisteix en cobrar a cada usuari que tenia productes en el lot distribuït el preu d'aquests. S'encarrega del cobrament l'agent centralitzat **Servei Entrega** i realitza la execució quan rep la informació anterior (que conté informació del lot també).



### 2.1.5. RecomanarProducte

Aquesta acció correspon a l'escenari **Recomanar** i consisteix en recomanar, per a tots els clients, un conjunt de productes que li podrien interessar. Aquesta acció l'executa l'**Agent Servei Client** de forma periòdica. Per calcular els productes del catàleg que li podrien interessar es basa en el seu historial de compra. Si no en té, se li recomanen productes amb valoracions altes.

Aquests seran enviats a l'**Agent Assistent** que sera responsable d'interactuar directament amb els usuari encarregant se de mostrar les recomanacions a través d'una interfície.

### 2.1.6. ValorarProducte

Aquesta acció correspon a l'escenari **Donar Feedback** i consisteix en sol·licitar, als clients que han realitzat compres de productes, la valoració d'algun d'aquests. L'a executa l'**Agent Servei Client** de forma periòdica. Demana les valoracions a l'**Agent Assistent** i, si aquest decideix valorar el producte, el sistema afegirà la valoració.

### 2.1.7. RetornarProducte

Quan l'**Agent Assistent** indiqui que desitja realitzar la devolució d'un producte, li enviarà a la botiga una sol·licitud de devolució. Això iniciarà l'acció de la nostra ontologia **RetornarProducte**.

Aquesta acció és executada per l'**Agent Servei Retornador** quan arribi la petició i es disposi a processar-la. El primer pas consisteix en fer la resolució de la petició que es calcularà en base als dies que fa que s'ha realitzat la entrega del producte. Si la resolució és correcta, l'agent es comunica amb el **Servei Entrega** per realitzar el pagament a l'usuari de l'import del producte a retornar. Aquesta acció de pagar al client representa una subclasse de la **RetornarProducte** anomenada **PagarUsuari**. El l'**Agent Servei Entrega** contacta amb el **Servei Pagador** per realitzar el pagament.

Per finalitzar la acció principal l'**Agent Servei Retornador** retorna a l'**Agent Assistent** la resolució del retornament. Si es positiva s'assigna un transportista que s'encarrega de la devolució i es retorna també la data de recolliment del paquet.

### 2.1.8. AfegirProducteExtern

Aquesta acció succeeix quan un venedor extern de la botiga virtual vol oferir un nou producte a través de la botiga virtual. L'**Agent Servei Catàleg** és qui rep aquesta sol·licitud del venedor extern i s'encarrega d'actualitzar el catàleg

### 2.1.9. EliminarProducteExtern

Aquesta acció succeeix quan un venedor extern de la botiga virtual vol eliminar un producte que ofereix a través de la botiga virtual. L'**Agent Servei Catàleg** és qui rep aquesta sol·licitud del venedor extern i s'encarrega d'actualitzar el catàleg

## 2.2. CentreLogístic

Aquesta classe representa els centres logístics que emmagatzemen els productes oferts per la botiga. El productes de venedors externs son distribuïts no estan distribuïts a aquests centres logístics. És la classe encarregada de realitzar totes les accions delegades a cada centre logístic. Per poder representar-la conté el següents atributs:

- Id: String que identifica cada un dels centres
- Ciutat: String que conté la ciutat on es troba ubicat el centre logístic

Amb l'atribut Ciutat podem calcular la geolocalització (latitud, longitud) de cada centre logístic i estipular la proximitat que té el centre a les destinacions de les comandes on s'han d'enviar.

El centre logístic conté una sola relació:

ProductesCentreLogistic		
Indica quins productes es troben al centre logístic		
Domini	CentreLogistic	1:N
Rang	Producte	1:N

## 2.3. Client

Aquesta classe representa un usuari del sistema. El client és qui inicia la majoria de les accions descrites a la ontologia. Conté els següents atributs:

- DNI: String que identifica cada un dels clients.

Les relacions de la classe Client són:

ClientComanda		
Indica de quin client pertany la comanda		
Domini	Comanda	1:1
Rang	Client	1:N

## 2.4. Comanda

La classe Comanda representa la compra de un nombre de productes que fa l'**Agent Assistent** a la botiga. Aquesta classe comanda també serveix per representar les subcomandes que es creen quan una comanda es divideix en parts per ser distribuïdes a centres logístics diferents. Això pot succeir quan una mateixa comanda té productes que es troben en centres logístics diferents.

Té els següents atributs:

- ID: String que identifica cada una de les comandes.
- Ciutat: String que conté la ciutat on s'han d'entregar els productes de la comanda.
- Data: DateTime que registra la data esperada d'entrega dels productes de la comanda.
- PreuTotal: Float que indica el preu total de la comanda.
- Prioritat: Int que indica el nivell de prioritat de la entrega. Conté valors de 1 a 3 que corresponen a: un dia, de 3 a 5 dies, quan sigui.
- TargetaCredit: String que representa la informació d'una targeta de crèdit. És nullable.
- CompteBancari: String que representa el codi IBAN d'un compte bancari. És nullable.

Ha de contenir o una TargetaCredit o un CompteBancari.

Les relacions de la classe Comanda són:

ClientComanda		
Indica de quin client pertany la comanda		
Domini	Comanda	1:1

Rang	Client	1:N
------	--------	-----

<b>Processa</b>		
Indica quina de quina comanda s'està processant l'enviament		
Domini	ProcessarEnviament	1:1
Rang	Comanda	1:1

<b>ProductesComanda</b>		
Indica quins productes conté la comanda		
Domini	Comanda	1:N
Rang	Producte	1:N

<b>RetornaProducteDeComanda</b>		
Indica de quina comanda és el producte que s'està retornant		
Domini	RetornarProducte	1:1
Rang	Comanda	1:1

<b>ComandaRespostaTemptativa</b>		
Indica de quina comanda és la factura que s'envia al client		
Domini	RespostaTemptativa	1:1
Rang	Comanda	1:1

## 2.5. Lot

Aquesta classe representa una agrupació de productes de comandes que ha rebut el centre logístic. S'utilitza per distribuir els productes en una unitat que sigui entregable pels transportistes ja que les comandes poden contenir qualsevol nombre

de productes. Per exemple podem tenir una comanda amb un sol producte que esdevindria en una carrega molt petita per un transportista o una comanda amb 40 productes que seria massa. En un mateix lot s'agrupen productes que tinguin una mateixa ciutat d'entrega.

Té els següents atributs:

- ID: String que identifica cada un dels lots.
- Ciutat: String que conté la ciutat on s'ha d'enviar.
- Pes: Float que representa el pes total de la suma de pesos dels productes que conté el lot.
- Prioritat: Int que indica el nivell de prioritat de la entrega (mateix que la de Comanda). Es registra la prioritat mitjana del Lot per a calcular la data d'entrega del lot.

Les relacions de la classe Lot són:

<b>AssignaLot</b>		
Indica quin lot s'assigna per entregar al transportista que ha guanyat la subasta		
Domini	AssignarTransportista	1:1
Rang	Lot	1:1

<b>EnviaCondicions</b>		
Indica de quin lot s'envien les condicions per realitzar l'entrega		
Domini	EnviarCondicionsEnviament	1:1
Rang	Lot	1:1

<b>EnviaContraoferta</b>		
Indica de quin lot s'envia la contraoferta per realitzar l'entrega		
Domini	EnviarContraoferta	1:1
Rang	Lot	1:1

<b>InformaLot</b>		
Indica de quin lot s'envia la contraoferta per realitzar l'entrega		
Domini	InformarEnviament	1:1
Rang	Lot	1:1

<b>ProductesLot</b>		
Indica de quins productes conté el lot		
Domini	Lot	1:N
Rang	Producte	1:N

<b>TransportistaLot</b>		
Indica de quin transportista s'ha assignat al lot		
Domini	Lot	1:1
Rang	Transportista	1:N

## 2.6. Producte

La classe producte és de les més importants de la nostra ontologia. Defineix les característiques essencials de cada producte del catàleg, tant si són propis de la botiga o si són de venedors externs.

Té els següents atributs:

- ID: String que identifica cada un dels productes.
- Nom: String que indica el nom del producte.
- Marca: String que indica la marca del producte.
- Pes: Float que indica el pes del producte.
- Preu: Float que indica el preu del producte.
- Categoria: String que s'utilitza per realitzar recomanacions i analitzar patrons d'interés dels clients.
- Valoració: Float que indica la valoració mitjana que té un producte. Es calcula a partir del feedback que aporta el client després de comprar-lo.

Les relacions de la classe Producte són:

<b>Afegeix</b>		
Indica quins productes s'estan editant del catàleg de productes de la botiga		
Domini	EditarProductesCatalog	1:N
Rang	Producte	1:N

<b>Compra</b>		
Indica quins productes s'estan comprant		
Domini	ComprarProductes	1:N
Rang	Producte	1:N

<b>Recomana</b>		
Indica quins productes s'estan recomanant al client		
Domini	RecomanarProducte	1:N
Rang	Producte	1:N

<b>RetornaProducte</b>		
Indica quin producte s'està retornant		
Domini	RetornarProducte	1:1
Rang	Producte	1:N

<b>Valora</b>		
Indica de quins productes s'està demanant feedback al client		
Domini	ValorarProducte	1:N
Rang	Producte	1:N

<b>ProductesComanda</b>		
Indica quins productes conté la comanda		
Domini	Comanda	1:N
Rang	Producte	1:N

<b>ProductesLot</b>		
Indica de quins productes conté el lot		
Domini	Lot	1:N
Rang	Producte	1:N

<b>ProductesVenedorExtern</b>		
Indica quin catàleg de productes ofereix el venedor extern a la botiga		
Domini	VenedorExtern	1:N
Rang	Producte	1:1

<b>ProductesCentreLogistic</b>		
Indica quins productes es troben al centre logístic		
Domini	CentreLogistic	1:N
Rang	Producte	1:N

## 2.7. RespostaTemptativa

Aquesta classe representa la primera factura que s'entrega al client en el moment de realitzar la compra. Es retorna informació bàsica sobre la comanda i una aproximació del temps d'entrega depenent de la prioritat d'entrega que s'ha especificat.

Té els atributs:

- ID: String que identifica cada una de les respostes temptatives.



- Ciutat: String que conté la ciutat on s'han d'entregar els productes de la comanda.
- Data: DateTime que registra la data esperada d'entrega dels productes de la comanda.
- PreuTotal: Float que indica el preu total de la comanda.

Les relacions de la classe Producte són:

<b>EnviaRespostaTemptativa</b>		
Indica la resposta temptativa que s'ha d'enviar. Aquesta té una relació amb la comanda per poder retornar la llista de productes comprats		
Domini	EnviarRespostaTemptativa	1:1
Rang	RespostaTemptativa	1:1

<b>ComandaRespostaTemptativa</b>		
Indica de quina comanda és la factura que s'envia al client		
Domini	RespostaTemptativa	1:1
Rang	Comanda	1:1

## 2.8. Restriccions

Aquesta classe identifica les restriccions que l'Agent Assistent envia per realitzar la búsqueda dels productes. Conté una serie de filtres per tal de concretar la búsqueda dels productes del catàleg. Aquesta classe té 4 subclasses que especifiquen el conjunt de restriccions que es pot instanciar:

- RestriccioMarca: Té un atribut String que defineix la marca del producte.
- RestriccioNom: Té un atribut String que defineix el nom del producte o part d'ell.
- RestriccioPreu: Té dos atributs PreuMax i PreuMin que defineixen un rang de preu dels productes a buscar.
- RestriccioValoracio: Té un atribut Valoracio que indica la valoració mínima que ha de tenir els productes de la búsqueda. Si s'especifica valoració = 4 vol dir que només es retornaràn productes amb valoració de 4 a 5.

- RestriccioCategoria: Té un atribut Categoria que indica la categoria a la que pertanyen els productes.

Les relacions són:

<b>Busca</b>		
Indica quines restriccions conté la búsqueda de productes que s'ha realitzat		
Domini	BuscarProductes	1:N
Rang	Restriccions	1:N

## 2.9. Transportista

Aquesta classe representa les empreses de transportistes que s'encarreguen d'entregar els lots que se li assignen. Assumim que no es troben localitzades a cap ciutat en concret i que totes tenen viabilitat per realitzar una entrega des de qualsevol centre logístic. Quan s'assigna un lot a un transportista aquest s'encarrega d'anar al centre logístic a buscar-lo en la data indicada i repartir-lo. Representa un agent extern al sistema.

Els atributs que s'enregistren de cada transportista son:

- ID: String que identifica cada un dels transportistes.
- Nom: String.
- PreuKg: Float que indica el preu per kg que estan disposats a pagar d'entrada per enviament.
- PreuKm: Float que indica el preu per km que estan disposats a pagar d'entrada per enviament.
- Marge: Float que indica el percentatge de Preu que estan disposats a acceptar per realitzar un encàrrec.

Com a relació del transportista trobem, únicament, la TransportistaLot que ja s'ha esmentat anteriorment.

## 2.10. VenedorExtern

Aquesta classe representa un venedor extern a la empresa de la botiga. Aquests utilitzen la botiga virtual per vendre-hi els seus productes i s'encarreguen de la

distribució dels seus productes venuts. Per tant només utilitzen l'app pel procés de venda.

Els atributs que s'enregistren son:

- Nom: String que representa el nom de la empresa i que s'utilitza per identificar cada un dels venedors externs (Assumim que no hi ha 2 empreses amb el mateix nom).
- CompteBancari: String que representa l'IBAN del compte bancari al que s'han de realitzar els pagaments.

Com a relació del VenedorExtern trobem, únicament, la ProductesVenedorExtern que ja s'ha esmentat anteriorment.

### 3. Descripció de la Implementació

El nostre sistema permet a l'usuari connectar-se a la aplicació mitjançant un Agent Assistant. Al executar aquest se li demana el DNI per tal de poder registrar l'usuari dins del sistema tal i com s'ha comentat a la explicació d'ontologia. A continuació, l'usuari té la opció d'accedir a realitzar una búsqueda de productes, a veure l'historial de comandes realitzades i a sota, si hi ha recomenacions, se li mostra un llistat de productes que el sistema li ha recomenat.

El nostre sistema també permet a empreses externes afegir productes externs per ser venuts mitjançant la nostra plataforma. Per fer-ho tenim un Agent Venedor Extern que, al executar-lo, pot registrar nous productes o visualitzar els que ja ha afegit anteriorment. Queda pendent poder editar aquests productes ja afegits.

A continuació veurem tot el conjunt de la implementació del sistema separat en apartats. Per començar una breu explicació de les eines de programació que hem emprat, seguidament una descripció dels Agents externs que hem implementat i finalment una explicació de les accions definides a la ontologia.

#### 3.1. Eines

En el desenvolupament del nostre sistema multiagent, hem adoptat una sèrie d'eines i tecnologies que han resultat essencials per a l'operativitat i la interacció efectiva entre els diversos components del sistema. Hem utilitzat Flask, un microframework de Python, que ha facilitat la creació d'interfícies web per als agents, permetent-nos definir rutes i gestionar peticions HTTP de manera eficaç. Això ha estat vital per al maneig de les comunicacions en el nostre entorn multiagent.

Per a la gestió de dades semàntiques, RDFlib ha estat una eina clau, permetent-nos treballar amb RDF i realitzar operacions sobre aquest format estructurat. Juntament amb SPARQLWrapper, que facilita la realització de consultes SPARQL, hem pogut manipular i consultar dades de manera precisa, cosa que és fonamental per a la interoperabilitat entre agents.

La creació i el maneig de grafos mitjançant els namespaces i classes proporcionats per RDFlib han jugat un paper crucial en la modelació de les entitats i accions del nostre sistema. Això ha permès una representació clara i eficient dels conceptes i relacions dins la nostra ontologia. Per a la comunicació entre agents, hem implementat missatges segons l'estàndard FIPA-ACL, utilitzant funcions

específiques per construir i enviar aquests missatges, garantint que siguin comprensibles independentment del desenvolupament dels agents.

Geopy ha estat utilitzada per a calcular distàncies geogràfiques, essencial per a la logística i la planificació de rutes en el nostre sistema distribuït. L'ús de la biblioteca multiprocessing ha millorat significativament l'eficiència del sistema, permetent la execució de tasques en paral·lel i millorant la resposta del sistema.

El servidor Apache Jena Fuseki ha proporcionat una plataforma robusta per a l'emmagatzematge i gestió de les dades RDF, permetent consultes eficients i gestió de dades a gran escala. A més, la integració amb serveis i APIs externs com OpenStreetMap ha enriquit les funcionalitats del sistema, permetent càlculs complexos i millorant la presa de decisions basada en dades reals.

En resum, l'adopció d'aquestes tecnologies ha permès no només cobrir les necessitats tècniques del projecte, sinó també garantir l'escalabilitat, mantenibilitat i extensibilitat del sistema, aspectes fonamentals per al seu èxit continuat en entorns de producció.

## 3.2. Agents externs

Al fer el disseny detallat ens vam adonar que era necessari començar implementant els agents externs ja que en la majoria d'accions hi intervenien o iniciaven la mateixa acció. Aquests agents externs com a tal no formen part dels agents del nostre sistema però hi interactuen enviant informació o revent resultats. Per tant, per poder començar a implementar funcionalitats i comprobar que s'executessin vam haver de començar amb aquests.

Per començar trobem l'agent extern més important que és l'**Agent Assistent**. Això és degut a que la majoria d'accions del sistema estan enfocades a l'usuari i aquest és el que ens simula la interacció amb ell. Per tant aquest agent al final fa d'intermediari entre les decisions de l'usuari i les accions que s'han de dur a terme al sistema. S'encarrega de desplegar el conjunt de vistes html amb la informació necessària rebuda d'altres agents i els botons i àrees de text per enviar consultes.

També hem implementat l'**Agent Venedor Extern** que realitza una funció similar d'intermediar a l'assistent. S'encarrega de fer d'intermediari entre el representant d'una empresa externa i la resta d'agents. També conté unes vistes html amb les que se li permet crear nous productes, llistar els productes que ha afegit anteriorment i esborrar-ne algun.

Agent Servei Pagament

### 3.3. Accions

#### 3.3.1. Buscar productes

Aquesta acció la desencadena l'usuari mitjançant l'**Agent Assistant** quan desitja fer una búsqueda de productes. Per fer-ho, ha d'accedir a buscar productes desde el menú principal. Aquest podrà omplir o no els filtres que desitgi. Els filtres son Nom, Marca, Categoria, Valoració, Preu Mínim i Preu Màxim. L'**Agent Assistant** recollirà les dades i formara un missatge RDF per descriure l'acció de búsqueda amb filtres. Al rebre el missatge l'agent **ServeiBuscador** processa el contingut, verifica l'acció i extreu les restriccions per la consulta.

Realitzara la consulta amb el productes enregistrats al sistema tant Producte com ProducteVenedorExtern. Els productes que compleixen els criteris seran recol·lectats en un graf RDF.

Els resultats es serialitzen en un XML i es retornen a l'**Agent Assistant**.

L'**Agent Assistant** processa el graf per extreure la informació dels productes i la presenta a l'usuari a través d'una interfície gràfica.

L'usuari podrà visualitzar el productes que coincideixen amb els seus criteris. En cas de no trobar-ne cap es mostra un missatge indicant que la busqueda a estat fallida.

#### 3.3.2. Comprar productes

L'usuari al indicar la ciutat de destí, la prioritat i la seva targeta de credit juntament a una llista dels productes a comprar fa que l'**Agent Assistant** creï un graf amb aquesta acció. Li assigna els atributs generals de la compra i després crea una relació "Compra" per l'acció i els productes de la compra i els seus atributs.

Envia aquest graf al **Servei Comandes** que agafa aquest gràfic i començara a formar la resposta temptativa. Aquesta és una comanda amb producte i preu, preu total de la compra, ciutat destí, prioritat i targeta utilitzada.

Aquí just abans de retornar la resposta temptativa i per què la resposta s'envia el més aviat possible començaran dos subprocessos. El primer subproceso notificará al **Servei Entrega** amb l'accio "CobrarProductesVenedorExtern".

El segon subprocés registra la comanda a la base de dades i realitza la búsqueda dels productes al centre logístic. Per cada producte busca a quin centre logístic esta disponible i després es calcula quin és el centre més proper a la ciutat de destinació de l'usuari. Un cop s'ha escollit quin producte s'assigna a cada centre logístic comença l'acció "ProcessarEnviament".

### 3.3.3. CobrarProductesVenedorExtern

La gestió del cobrament de productes de venedors externs dins d'un sistema multiagent es desenvolupa a través de diverses interaccions entre els agents **Servei Comandes**, **Agent Pagament**, **Agent Assistent**, i altres agents implicats en el procés, com **Servei Entrega**.

Quan un usuari completa una compra que inclou productes de venedors externs, l'**Agent Assistent** inicia el procés recopilant tota la informació necessària dels productes adquirits, com són els preus, les descripcions i els identificadors dels venedors externs. Aquesta informació s'estructura en un missatge RDF, que s'envia al **Servei Comandes** per processar la comanda.

El **Servei Comandes**, al rebre aquesta informació, verifica i valida els detalls de la comanda. Això inclou comprovar la disponibilitat dels productes i confirmar els detalls d'enviament. Una vegada aprovat, el **Servei Comandes** prepara un nou missatge per al **Agent Pagament**, que és l'agent encarregat de realitzar les transaccions econòmiques.

L'**AgentPagament**, al rebre el missatge, processa el pagament. Això pot involucrar la interacció amb sistemes de pagament externs o bancs per transferir els fons del compte del client al compte del venedor extern. Un cop el pagament és realitzat amb èxit, l'**Agent Pagament** envia una confirmació de la transacció al **Servei Comandes**.

Després de rebre la confirmació del pagament, el **Servei Comandes** notifica l'èxit de la transacció a l'**Agent Assistent**, que finalment actualitza l'estat de la comanda en el sistema. L'**Agent Assistent** també s'encarrega de notificar al client que la seva compra ha estat processada correctament i que el pagament al venedor extern s'ha realitzat satisfactòriament.

Aquest procés no només assegura que totes les parts de la compra siguin gestionades eficaçment, sinó que també manté un alt nivell de comunicació entre els diferents agents per garantir la satisfacció del client i la correcta distribució dels fons als venedors externs.

### 3.3.4. ProcessarEnviament

Aquesta acció comença a **Servei Comandes** un cop ja es saben quins productes es demana a cada centre logístic. Aquí es construeix el graph de l'acció com a subcomanda dels productes que s'envien a cada centre logístic.

Com el **Servei Centre Logístic** està distribuït el **Servei Comandes** comandes

inicia tants subprocessos com serveis hi hagi actius per que realitzin la feina en paral·lel.

La comunicació arribar a al **Servei Centre Logístic** aquest agent agafa la comanda i els seus productes i la transforma en un lot llest per enviar.

El centre logístic inicia una comunicació amb l'**Agent Transportista** de la seva ciutat per per demanar el preu d'enviament d'un lot a través de l'acció "EnviarCondicionsEnviament". Aquesta acció es un graph amb la ciutat de destí, el pes del lot i la prioritat. L'**Agent Transportista** al rebre aquesta acció calcula per cada un dels transportistes de la ciutat quin seria el cost del l'enviament en funció dels km i el pes del lot. A més aquest preu hi apliquem un random perquè no sempre surti el transportista que ofereix el preu més baix per km i kg. Un cop calculats els preu d'enviament de cada transportista es tornen en forma de graph juntament amb la data d'arribada calculada segons la prioritat al **Servei Centre Logístic**. Llavors, el centre logístic recollirà al transportista que li hagi ofert un preu més barat per iniciar una negociació.

La negociació será indefinida fins que no s'arribi a un acord. El primer pas de la negociació es enviar una contraoferta pero el 80% del primer preu. Aquesta contraoferta es una acció "EnviarContraoferta" i s'envia a través de un graph amb el ultim preu passat per el transportista. L'**Agent Transportista** mira si el nou preu està dins del seu marge permès per reduir i si hi està retorna que ha acceptat la contraoferta, en canvi, si rebutja la contraoferta el que fa és retornar un nou preu dins dins del seu marge. Quan el centre logístic rep la resposta a la contraoferta si ha sigut acceptada continua amb el proces, si ha sigut rebutjada mira si el nou preu ofert per el Transportista quadra amb una reducció de % de reducció +5 i si quadra accepta sinó torna a començar el procés de negociació.

Un cop ha finalitzat el procés de negociació comença un subproces per comunicar al transportista que ha sigut assignat i esperar a que reculli el paquet. Aquest subproces crea un graph amb l'acció "AssignarTransportista" amb el id del lot i el transportista assignat i s'envia a l'**Agent Transportista** que té un log on indica que el transportista ha recollit el lot.

Continuant el procés principal i un cop s'ha acabat la negociació s'ha d'informar a l'usuari que ja li han assignat un transportistes a alguns productes de la seva comanda. Això es fa a través de l'acció "InformarEnviament". El graph d'informar enviament conte la comanda el lot, la data d'entrega i el transportista assignat. Aquesta acció s'envia al **Servei entrega**, aquest agent desglosa la informació que li



arriba i la concentra en un graph. Abans però accedeix a la base de dades per obtenir el nom del producte per a partir de la ontologia i afegeix al graph de l'acció.

Abans de comunicar els productes a l'**Agent Assistent** el **Servei Entrega** guarda a la base de dades de comandes que els productes ja han sigut assignats a un transportista amb el nom del transportista i la data d'entrega.

Finalment s'envia aquest graph a l'**Agent Assistent**. Aquest agent es guarda els productes que han sigut enviats en una llista que es mostrarà per pantalla en el centre de notifiacions.

### 3.3.5. Valorar productes

Aquesta acció la desencadena l'usuari mitjançant l'**Agent Assistent** quan desitja valorar un producte comprat. Per fer-ho, ha d'accedir a l'historial de comandes realitzades i entrar a veure el llistat de productes de la comanda on hi vol valorar algun producte.

L'usuari ha de completar un formulari amb el que pot especificar la seva valoració amb una escala numèrica.

Un cop enviada la valoració l'**Agent Assistent** recull les dades i construeix un missatge RDF que descriu l'acció ValorarProducte. El missatge inclou informació tals com nom del Producte, id de la Comanda i valoració donada.

El missatge serà enviat a l'agent **ServeiClients** que s'encarrega de processar les valoracions dels productes.

En rebre el missatge **ServeiClients** extreu la informació , consulta DB per verificar la valoració actual del Producte, actualitza la valoració basant-se en la nova entrada calculant el promig. A més actualitza la valoració del Producte en el context de la Comanda (ProducteComanda) per mantenir un registre detallat d'aquestes.

Un cop completada l'actualització de la valoració **ServeiClients** envia confirmació a l'**Agent Assistent** indicant l'èxit de l'operació.

L'impacte de les valoracions al sistema es veurà reflectit en la presentació de productes i les recomanacions als usuaris.

### 3.3.6. Retornar productes

Aquesta acció la desencadena l'usuari mitjançant l'**Agent Assistant** quan desitja retornar un producte comprat. Per fer-ho, ha d'accedir a l'historial de comandes realitzades i entrar a veure el llistat de productes de la comanda on hi vol retornar algun producte. Allà es mostra la resolució de retornament que tenen els productes de la comanda. Si el transportista no està assignat o el producte no és permès retornar-lo. Si aquestes dos condicions es compleixen, llavors surt al llistat de productes, per cada un que es pugui retornar, un botó que porta a una nova pantalla on l'usuari pot introduir el motiu de retornament.

Quan es retorna un producte l'**Agent Assistant** s'encarrega de generar un graf amb la acció **RetornarProducte**, relacionada amb el **ProducteComanda** que es desitja retornar i el motiu de la devolució. Aquest graf d'acció s'envia al **Servei Retornador** qui decidirà si s'accepta o no.

Per arribar a una decisió es mira el nombre de dies que fa que el producte ha estat entregat i el missatge de devolució. Si el missatge indica que no se satisfan les expectatives del producte i han passat 15 dies llavors es rebutja la petició. Si el missatge indica que el producte és defectuós o equivocat i no han passat més de 30 dies s'accepta la petició. Per simplicitat, al realitzar la demo no es realitzaran les comprobacions de dies. Quan una petició ha estat acceptada o rebutjada es registra al **ProducteComanda** l'estat de retornament i es retorna a l'agent assistent aquesta resolució. Quan una petició s'accepta, a més a més, el **Servei Retornador** s'encarrega de contactar amb el **Servei Entrega** per gestionar el retornament del producte. Li envia un graf amb el client destinatari, l'import i el producte. Aquest missatge representa l'acció **PagarUsuari** de la nostra ontologia i hi intervenen el **Servei Entrega** i l'agent extern **Agent Pagador**.

Quan li arriba el missatge al **Servei Entrega**, aquest s'encarrega de registrar a la base de dades de Fuseki la transacció per mantenir-hi un historial. A continuació li envia un missatge a l'**Agent Pagador** amb la mateixa informació que ha rebut del retornador per a que aquest realitzi la transacció monetaria. Com que l'**Agent Pagador** no es troba implementat per fer transaccions reals realitza un logg simulant la transacció i aquí acaba.

L'**Agent Assistant** rep un graf de resposta amb la resolució del **Servei Entrega**. Si la resolució és positiva es retorna també informació d'un transportista i de la data en que aquest transportista li anirà a buscar el paquet. Assignem un transportista per defecte que s'encarrega de tots els retornaments i la data son 5 dies després de la petició de devolució.

En el cas que el producte pertany a un venedor extern l'**Agent Assistent** indica a la taula de productes comprats que la devolució d'aquest es gestiona a través de l'empresa externa.

### 3.3.7. Recomanar productes

El procés de recomanació de productes involucra l'**Agent Assistent** l'agent **Servei Clients** i utilitza la informació de les preferències i valoracions prèvies dels usuaris per suggerir productes que puguin ser del seu interès.

Dins de **Servei Clients** existeix un procés automàtic que periodicament avalua les necessitatss de recomanacio per cada usuari basant en les interaccions passades i les valoracions fetes.

Es farà una consulta per identificar tots els usuaris que podria beneficiarse de les recomanacions basant nos en l'activitat recent o la manca d'interaccions.

Per cada client identificat es revisa el seu historial de comandes i valoracions per determinar les preferències. Això inclou un anàlisi de productes altament valorats i categories freqüentment a productes comprats. Amb la informació recopilada els sistema genera una llista de productes recomanats a través d'una consulta a la BD.

Per cada client es construeix un missatge RDF que conté els productes recomanats.

Aquests seran enviats a l'**Agent Assistent** que sera responsable d'interactuar directament amb els usuari encarregant se de mostrar les recomanacions a través d'una interfície.

Basant-nos en la interacció dels usuaris amb les recomanacions (per exemple ignorar-la) el sistema ajusta futures recomanacions per millorar la precisió i rellevància completant així el cicle de retroalimentació i optimització de les recomanacions.

### 3.3.8. Afegir producte extern

Aquesta acció succeeix quan un venedor extern, mitjançant l'**Agent Venedor Extern** indica que vol afegir un producte. Això ho fa al menú principal mitjançant un botó del html. Primerament es fa un register similar al que fa el client a l'**Agent Assistent** on l'administrador de l'empresa externa ha de seleccionar a quina empresa pertany (en un desplegable). A continuació pot triar si afegir un producte nou o llistar els productes que té afegits. Quan vagi a afegir un producte se li mostra un formulari on cal introduir la informació necessària (Nom, Preu, Marca, Categoria i Pes).

Quan s'inicia l'acció, l'**Agent Venedor Extern** construeix un graf d'acció `AfegirProducteExtern` que conté la informació del producte a afegir i li envia el missatge al **Servei Catàleg**. Aquest s'encarrega d'escriure a la base de dades de Fuseki el nou producte amb un id vàlid i inicialitzant les dades que falten i li retorna una resposta confirmant que s'ha afegit correctament.

### 3.3.9. Eliminar producte extern

L'**Agent Venedor Extern** també té la opció de llistar els productes que ha afegit. En aquest llistat apareix per cada producte un botó per borrar-lo del catàleg de productes oferts. El funcionament és similar a l'acció anterior. L'**Agent Venedor Extern** contacta amb el **Servei Catàleg** indicant-li l'ID del producte que cal esborrar. Aquest darrer s'encarrega de realitzar un update sobre la base dades esborrant la instància i llavors retorna la resposta confirmant la eliminació.

## 4. Tasques nivell avançat

La primera tasca extra que tenim es que no hi ha un sol agent transportista sino que hi ha 5 instàncies inicialitzades en 5 ports diferents per els 5 centres logístics que tenim en el sistema (Banyoles, Barcelona, Tarragona, Valencia i Zaragoza). Llavors el centre logístic contacte amb el transportistes de la seva ciutat per preguntar els preus de transport.

La segona tasca extra que tenim és la negociació complexa entre centre logístic i agent transportista. El centre logístic contacta amb l'agent transportista de la seva ciutat i aquest retorna el preu dels transportistes que hi té assignat. El centre logístic rep les ofertes dels transportistes i tria la del transportista mes barato llavors inicia una negociació amb aquest. Aquest dos es van tirant oferta i contraoferta fins que arriben a un acord.

La tercera tasca extra es que els productos no están en un sol centre logístic sinó que están distribuïts. Primer al crear els productes a través del random info s'assignen a un o varis centres logístics de forma aleatoria. A la hora de realitzar la compra el servei comandes busca per cada producte quin és el centre logístic més proper de l'usuari que te el producte i llavors quan ja ha trobat tots els productes li envia la comanda a cada centre logístic.

De la quarta tasca d'implementar l'agent pagador el que hem fet és que tots els serveis que el necessiten si poden comunicar però la lògica del pagador es tenir un registre amb els logs de l'acció que hi arriba. El servei entrega és qui si comunica directament i fa les peticions de cobrar el producte al l'usuari, pagar al venedor extern i retornar els diners.

Finalment hem implementat que a través de l'historial de compres el nostre sistema periòdicament recomana productes a l'usuari de la marca o de la categoria dels productes que havia valorat prèviament amb una puntuació de més de 3.

– Recordar valoració periòdicament –

## 5. Tasques nota extra

Coma feina extra el que hem implementar mes tasques de nivell avançat. A part el que hem fet és desplegar un servidor d'apache Jena/Fuseki que utilitzem com a base de dades per guardar les tripletes.

Llavors des del codi ens comuniquem amb aquesta base de dades a través de request i comandes SPARQL.

## 6. Planificació

En el nostre grup hem seguit una metodologia de desenvolupament en cascada on ens hem repartit el conjunt d'accions de la ontologia i les hem anat desenvolupant poc a poc en paral·lel fins a completar-les. Inicialment vam començar repartint les tasques segons els agents del sistema però ràpidament vam veure que un mateix agent desencadenava moltes accions que necessitaven comunicacions amb altres agents per poder-se completar. D'aquesta forma calia una sincronització massa elevada entre els membres de l'equip i ens impedia treballar per separat. És per aquest motiu que hem fet d'aquesta forma el repartiment. Adicionalment també hem separat alguna subtasca d'una acció massa grossa que es podia repartir.

A continuació es mostra una taula amb les accions i subtasques repartides:

Nil	Lluc	Adrià
Retornar producte	Implementació Jena-Fuseki server	Valorar producte
Afegir producte extern	Crear script generar productes aleatoris	Buscar productes
Eliminar producte extern	Buscar productes centre logístic	CSS templates
Registrar comandes	Creació de lots	Notificacions
Historial comandes	Negociació transportistes	Agent Assistant
	Registrar transaccions de l'aplicació	
	Recomanar productes	

## 7. Evaluació de resultats

Estem convençuts que hem desenvolupat un sistema complet que resol el problema plantejat. L'Agent Ssistent que hem creat pot gestionar totes les tasques assignades, des de la compra de productes fins a la valoració o devolució dels mateixos. A més, hem dissenyat un sistema intern distribuït en agents capaç de gestionar totes les peticions de l'usuari.

En primer lloc, hem aconseguit una cerca de productes eficient, capaç de buscar tant productes externs com propis del sistema, respectant totes les restriccions indicades. També hem desenvolupat un procés de compra que permet als usuaris adquirir qualsevol producte proporcionant la informació necessària. Hem creat diversos agents per distribuir el procés de compra i els seus subprocesos. Per exemple, hem dissenyat una entitat per al Centre Logístic, capaç de negociar enviaments de productes amb transportistes, els quals ofereixen diferents preus d'enviament. Hem fet que aquest procés de negociació sigui iteratiu i no es resolgui amb un sol procés de contraofertes.

Un altre element clau és el registre de transaccions amb pagaments i retorns gràcies a l'agent pagament i el servei entrega, mantenint un registre econòmic de totes les transaccions dins del sistema. El sistema també inclou funcionalitats per a la devolució i recomanació de productes, permetent als usuaris retornar articles complint certs requisits i recuperant els seus diners si és viable.

També s'ha aconseguit implementar un sistema de recomanacions, basant-se en l'historial de cerques i compres de l'usuari per suggerir productes que puguin ser del seu interès.

El nostre sistema cobreix un ampli ventall de funcionalitats que solucionen el problema plantejat, assegurant que l'objectiu de tenir un assistent per al client es compleixi.

En el futur, es podria ampliar la flexibilitat de les entitats que es relacionen amb el sistema ja que actualment només es pot afegir productes externs amb 3 empreses preconfigurades. Igualment, es podrien afegir i eliminar transportistes per fomentar la competència en aquest sector. Una altra millora seria permetre als usuaris deixar comentaris a més de valoracions dels productes, o millorar el sistema de pagaments i transaccions fent un control més extens dels mètodes de pagament.



Per a una avaluació completa dels resultats, també cal tenir en compte les limitacions de la nostra solució i les seves implicacions. Una de les principals limitacions ha estat el desconeixement de les biblioteques i eines utilitzades. Durant la implementació, hem après a mesura que les necessitàvem, però creiem que amb més experiència hauríem pogut aconseguir una solució de més qualitat i ampliar algunes seccions. L'ús de noves tecnologies que desconeixiem ha relantit molt el desenvolupament de tasques simples.

Pel que fa al disseny, hem trobat dificultats amb les eines de disseny, ja que és complicat trobar informació i exemples. Tot i disposar dels vídeos de l'assignatura, la manca de referències d'altres usuaris ha pogut afectar la qualitat del disseny. També hem creiem que podríem haver dedicat més hores a aquest projecte de no ser per la sobrecarrega de feina d'aquest trimestre. Al coincidir amb tots els projectes de la especialitat de Software com el de PES, ASW, SIM o el múltiples projectes de l'optativa de Videojocs no hem pogut destinar el temps que requereix un projecte d'aquestes dimensions.

Així doncs, el temps ha estat un factor limitant en termes de qualitat i complexitat de la solució proposada. Amb més temps podríem desenvolupar una solució que abordés més aspectes del problema. Tot i així estem molt contents amb el resultat obtingut.