



Pràctica de Sistemes Basats en el Coneixement

Sistema Expert de Recomanació d'Habitatges de Lloguer

GRAU IA – Q1 CURS 2025-2026

Departament de Ciències de la Computació
Universitat Politècnica de Catalunya

ANEL ADEMOVIC SULJIC

anel.suljic@estudiantat.upc.edu aleix.pitarch@estudiantat.upc.edu

ALEIX PITARCH

JOAN SOLINA
joan.solina@estudiantat.upc.edu

14 de desembre de 2025

Índex

1 Identificació	6
1.1 Descripció del problema	6
1.2 Anàlisi de viabilitat	6
1.2.1 Adequació d'un Sistema Basat en Coneixement	6
1.2.2 Limitacions d'altres aproximacions	7
1.3 Identificació de fonts de coneixement	7
1.3.1 Fonts primàries	7
1.3.2 Fonts secundàries	8
1.4 Objectius del sistema	8
1.4.1 Objectius funcionals	8
1.4.2 Objectius no funcionals	9
1.5 Abast del sistema	10
1.5.1 Dins de l'abast	10
1.5.2 Fora de l'abast	10
1.6 Resultats esperats	11
1.6.1 Format de sortida	11
1.6.2 Informació per oferta	11
1.7 Beneficis esperats	12
1.7.1 Per als usuaris	12
1.7.2 Per als agents immobiliaris	12
1.7.3 Per als propietaris	12
1.8 Conclusions de la fase d'identificació	13
2 Conceptualització	14
2.1 Introducció a la fase de conceptualització	14
2.2 Conceptes principals del domini	14
2.2.1 Habitatge	14
2.2.2 Localització	15
2.2.3 Servei	15
2.2.4 Sol·licitant	16
2.2.5 Oferta	17
2.3 Relacions entre conceptes	18
2.3.1 Relacions d'associació bàsiques	18
2.3.2 Relacions de proximitat	18
2.3.3 Relacions de preferència i restricció	18
2.3.4 Relacions espacials addicionals	19
2.4 Ús de models de llenguatge per a la conceptualització	19
2.4.1 Metodologia d'ús dels models de llenguatge	19
2.4.2 Exemples de preguntes i respostes	19
2.4.3 Validació i integració del coneixement obtingut	21

2.4.4	Limitacions i consideracions crítiques	22
2.5	Regles heurístiques identificades	22
2.5.1	Regles basades en perfil demogràfic	23
2.5.2	Regles de sentit comú sobre habitatges	24
2.5.3	Regles de proximitat i distància	25
2.5.4	Regles de pressupost	25
2.6	Descomposició del problema en subproblemes	25
2.6.1	Problema principal	26
2.6.2	Subproblema 1: Abstracció (Inferència de necessitats)	26
2.6.3	Subproblema 2: Resolució (Avaluació i filtratge)	26
2.6.4	Subproblema 3: Refinació (Classificació final)	28
2.7	Exemples de resolució manual	28
2.7.1	Exemple 1: Família Garcia	29
2.7.2	Exemple 2: Grup d'estudiants Marc i companys	31
2.8	Organització de la resolució	32
2.8.1	Avantatges de l'organització en fases	32
2.8.2	Flux de dades entre fases	33
2.9	Conclusions de la fase de conceptualització	33
3	Formalització	35
3.1	Construcció de l'ontologia	35
3.1.1	Procés de construcció	35
3.1.2	Jerarquia de classes	36
3.1.3	Atributs de les classes	38
3.1.4	Relacions entre classes	40
3.2	Metodologia de resolució de problemes	42
3.2.1	Paradigma escollit: Forward Chaining amb fases	42
3.2.2	Justificació del Forward Chaining	42
3.2.3	Justificació de l'organització en fases	43
3.2.4	Templates auxiliars per al raonament	43
3.3	Justificació de decisions de disseny	45
3.3.1	Per què no Backward Chaining?	45
3.3.2	Per què Templates en lloc de només Objects?	46
3.3.3	Per què tres fases en lloc de dues o més?	46
3.3.4	Per què pre-calcular les proximitats?	47
3.4	Cobertura de l'ontologia	47
4	Implementació	49
4.1	Arquitectura del sistema	49
4.1.1	Estructura de fitxers	49
4.1.2	Flux d'execució	50
4.2	Traducció de l'ontologia a CLIPS	50
4.2.1	Procés de generació	50

4.2.2	Estructura de les classes generades	51
4.3	Metodologia de desenvolupament	52
4.3.1	Iteració 1: Prototip mínim funcional	52
4.3.2	Iteració 2: Ampliació de l'ontologia i millora de regles	53
4.3.3	Iteració 3: Refinament i optimització	54
4.3.4	Iteració 4: Proves exhaustives i correcció d'errors	55
4.4	Modularització del codi	56
4.4.1	Estratègies de modularització	56
4.4.2	Organització de les fases	57
4.5	Implementació de regles clau	59
4.5.1	Càcul de distàncies i proximitats	59
4.5.2	Inferència de requisits	60
4.5.3	Descart amb justificació	61
4.5.4	Classificació multi-criteri	62
4.6	Control de flux amb salience	62
5	Jocs de prova	64
5.1	Estratègia de selecció dels jocs de prova	64
5.1.1	Criteris de selecció	64
5.1.2	Cobertura de regles	65
5.2	Descripció dels jocs de prova principals	66
5.2.1	Prova 1: Família amb fills i mascota	66
5.2.2	Prova 2: Grup d'estudiants amb pressupost estricte	70
5.2.3	Prova 3: Persona gran amb necessitat d'accessibilitat	71
5.2.4	Prova 4: Persona que evita soroll	74
5.2.5	Prova 5: Parella amb plans de tenir fills	75
5.2.6	Error 2: Conflicte de salience entre fases	83
5.2.7	Error 3: Tractament incorrecte de pressupost flexible	83
5.2.8	Error 4: No detecció d'ofertes sospitosament barates	84
5.3	Casos extrems i límits	85
5.3.1	Cas extrem 1: Cap oferta adequada	85
5.3.2	Cas extrem 2: Preu exactament igual al màxim	86
5.3.3	Cas extrem 3: Superfície mínima justa	86
5.4	Eficiència i escalabilitat	86
5.5	Conclusions de les proves	87
6	Conclusions	88
6.1	Assoliment dels objectius	88
6.1.1	Objectius funcionals	88
6.1.2	Objectius no funcionals	90
6.2	Aprendentatges principals	91
6.2.1	Sobre sistemes basats en el coneixement	91
6.2.2	Sobre metodologia de desenvolupament	92

6.2.3	Sobre CLIPS i programació basada en regles	93
6.3	Limitacions del sistema	93
6.3.1	Limitacions funcionals	94
6.3.2	Limitacions tècniques	94
6.3.3	Limitacions de cobertura	95
6.4	Treball futur	95
6.4.1	Millors a curt termini	96
6.4.2	Extensions funcionals	96
6.4.3	Integració amb dades reals	97
6.4.4	Hibridació amb Machine Learning	98
6.5	Reflexió final	99

1 Identificació

1.1 Descripció del problema

El mercat immobiliari de lloguer a Barcelona es caracteritza per una elevada complexitat: milers d'ofertes amb característiques heterogènies, sol·licitants amb necessitats diverses i una àmplia gamma de serveis urbans distribuïts per la ciutat. En aquest context, trobar l'habitatge més adequat per a cada perfil de sol·licitant no és trivial, ja que cal considerar simultàniament múltiples factors que interactuen entre ells.

El problema que abordem consisteix en desenvolupar un sistema capaç de recomanar habitatges de lloguer a diferents tipus de sol·licitants (famílies amb fills, estudiants, persones grans, parelles, etc.) considerant no només les característiques intrínseques de cada habitatge (preu, superfície, nombre d'habitacions, equipament), sinó també factors externs com la proximitat a serveis urbans rellevants (transport públic, escoles, centres de salut, zones verdes) i les preferències específiques de cada perfil demogràfic.

La complexitat del problema radica en diversos aspectes. En primer lloc, les característiques rellevants per a un sol·licitant poden ser completament diferents de les rellevants per a un altre: mentre que una família amb fills petits prioritzarà la proximitat a escoles i zones verdes, un grup d'estudiants valorarà més el transport públic i un preu econòmic. En segon lloc, no existeix una única "millor" solució, sinó que cal proporcionar recomanacions graduades que ajudin el sol·licitant a prendre decisions informades. Finalment, el sistema ha de ser capaç d'explicar les seves recomanacions de manera transparent, no només indicant quins habitatges són adequats, sinó també per què ho són i què els manca als que no compleixen tots els requisits.

1.2 Anàlisi de viabilitat

1.2.1 Adequació d'un Sistema Basat en Coneixement

Abans d'emprendre el desenvolupament del sistema, cal analitzar si l'enfocament de Sistemes Basats en Coneixement (SBC) és apropiat per aquest problema. Diversos factors justifiquen aquesta elecció.

En primer lloc, existeix coneixement expert clarament identifiable. Els agents immobiliaris experimentats apliquen regles heurístiques ben establertes quan recomanen habitatges: saben que les famílies necessiten escoles properes, que les persones grans requereixen accessibilitat, que els estudiants prioritzen el transport públic i el preu. Aquest coneixement és articulable, estructurable i pot formalitzar-se en forma de regles.

En segon lloc, el problema no té una solució algorítmica directa. No es tracta simplement d'ordenar ofertes per preu o filtrar per nombre d'habitacions, sinó d'avaluar múltiples factors qualitatius que interactuen de manera complexa. Un habitatge pot ser excel·lent per a un perfil i completament inadequat per a un altre, i aquesta adequació depèn de regles expertes

més que de càlculs matemàtics simples.

En tercer lloc, la complexitat del problema és moderada. És prou complex per necessitar intel·ligència artificial (no es pot resoldre amb simples filtres), però no tant com per requerir tècniques d'aprenentatge automàtic que necessitarien milers de dades d'entrenament. La quantitat de coneixement és manejable i pot estructurar-se adequadament.

Finalment, l'explicabilitat és un requisit fonamental. Els usuaris no només volen saber quins habitatges se'ls recomana, sinó entendre per què. Un SBC permet proporcionar justificacions clares i traçables de cada decisió, cosa que seria molt difícil amb aproximacions de caixa negra com les xarxes neuronals.

1.2.2 Limitacions d'altres aproximacions

Per reforçar la justificació de l'ús d'un SBC, considerem breument les limitacions d'altres aproximacions possibles.

Una cerca simple per filtres, com la que ofereixen moltes plataformes immobiliàries actuals, no capture el coneixement expert sobre adequació. L'usuari pot filtrar per preu, número d'habitacions i barri, però el sistema no li dirà que aquell habitatge és inadequat perquè està lluny d'escoles tot i tenir el nombre d'habitacions correcte, ni li suggerirà alternatives millors considerant el seu perfil global.

Les tècniques de Machine Learning, tot i ser potents, presenten diversos inconvenients per aquest problema. Primer, requeririen grans quantitats de dades de preferències reals d'usuaris, que no sempre estan disponibles. Segon, són menys transparents: és difícil explicar per què un model neuronal recomana un habitatge concret. Tercer, poden perpetuar biaixos presents en les dades d'entrenament. Finalment, són més costoses computacionalment i més complexes de mantenir i actualitzar quan canvien les regles del domini.

Els sistemes de puntuació numèrics simples, on cada característica suma o resta punts, tampoc són adequats. No capturen adequadament la diferència entre restriccions obligatòries (que descarten completament un habitatge) i preferències opcionals (que el fan més o menys atractiu). A més, assignar pesos numèrics a cada característica és arbitrari i no reflecteix com realment raonen els experts.

1.3 Identificació de fonts de coneixement

El desenvolupament d'un SBC requereix identificar clarament d'on prové el coneixement que codifarem al sistema. Distingim entre fonts primàries (contacte directe amb experts) i fonts secundàries (coneixement publicat o sistemes existents).

1.3.1 Fonts primàries

Les fonts primàries proporcionen coneixement directe i especialitzat del domini. En el nostre cas, identifiquem tres tipus principals d'experts.

Els agents immobiliaris amb experiència en lloguer d'habitatges són la font més valuosa. Aquests professionals han desenvolupat heurístiques implícites a través de l'experiència: saben instantàniament si un habitatge és adequat per a una família amb fills petits o per a un grup d'estudiants. Poden articular no només què recomanarien, sinó també per què, i quins factors compensatoris existeixen (per exemple, quan un preu lleugerament més alt es compensa amb una localització excel·lent).

Els consultors d'habitacle social i gestors de polítiques públiques aporten una perspectiva complementària. Coneixen les necessitats específiques de col·lectius vulnerables (persones grans, famílies monoparentals), les normatives d'accessibilitat, i els criteris objectius per determinar si un habitatge és adequat per a un perfil determinat.

Finalment, els gestors de propietats i administradors de finques tenen coneixement pràctic sobre les característiques tècniques dels habitatges, les seves limitacions reals (més enllà del què diu l'anunci), i com aquestes característiques afecten la vida quotidiana dels residents.

1.3.2 Fonts secundàries

Les fonts secundàries complementen i validen el coneixement obtingut dels experts. Hem identificat diverses categories rellevants.

Les plataformes immobiliàries online (Idealista, Fotocasa, Habitaclia) no només proporcionen dades sobre l'estructura dels anuncis i les característiques més rellevants del mercat, sinó que també ofereixen sistemes de cerca i filtres que reflecteixen implícitament criteris d'adequació. Analitzar aquestes plataformes ens permet entendre quines característiques es consideren suficientment importants com per ser filtres de cerca.

Els estudis demogràfics i sociològics sobre necessitats habitacionals proporcionen coneixement validat científicament sobre les preferències de diferents grups d'edat, tipologies familiars i perfils socioeconòmics. Aquests estudis ajuden a objectivar el coneixement expert i a identificar patrons sistemàtics.

Les normatives legals, especialment la Llei d'Arrendaments Urbans (LAU) i les regulacions d'accessibilitat, estableixen criteris objectius que el sistema ha de respectar. Per exemple, els requisits d'accessibilitat per a persones amb mobilitat reduïda no són opinables, sinó que estan legalment definits.

Finalment, i de manera innovadora, els models de llenguatge actuals (ChatGPT, Claude, Gemini) poden actuar com a "experts virtuals" per a la fase de conceptualització. Tot i que no substitueixen el coneixement d'experts reals, permeten explorar ràpidament diferents escenaris, obtenir estructuracions inicials del coneixement, i validar la coherència de les regles proposades. És important, però, contrastar aquest coneixement amb fonts autoritzades.

1.4 Objectius del sistema

1.4.1 Objectius funcionals

El sistema ha de complir quatre objectius funcionals principals, cadascun amb requisits específics que detallarem.

El primer objectiu és generar recomanacions graduades d'habitatges. No n'hi ha prou amb dir "aquest habitatge és bo" o "aquest és dolent", sinó que cal proporcionar una classificació en tres nivells: Molt Recomanable (compleix tots els requisits i té característiques excepcionals), Adequat (compleix tots els requisits), i Parcialment Adequat (acceptable però amb alguns criteris no complerts). Aquesta gradació permet a l'usuari prendre decisions informades segons la seva flexibilitat i urgència.

El segon objectiu és descartar de manera justificada els habitatges clarament inadequats. Si un habitatge no permet mascotes i el sol·licitant en té, o si és inaccessible per a algú amb mobilitat reduïda, no només s'ha d'excloure de les recomanacions, sinó explicar clarament per què. Aquestes justificacions han de ser concises i comprensibles per a usuaris no experts. El tercer objectiu és proporcionar explicacions detallades de les recomanacions. Per als habitatges recomanats, cal indicar quins són els seus punts forts (proximitat a escola, preu excel·lent, terrassa gran). Per als parcialment adequats, cal especificar què els manca o quins criteris no compleixen completament. Aquesta transparència és essencial per a la confiança de l'usuari i per permetre que prengui decisions informades.

El quart objectiu és inferir necessitats implícites a partir del perfil del sol·licitant. Si una família indica que té fills petits, el sistema ha de deduir automàticament que necessitarà escoles properes i zones verdes, sense que l'usuari ho hagi d'especificar explícitament. Aquesta capacitat d'inferència demostra la intel·ligència del sistema i millora significativament l'experiència d'usuari.

1.4.2 Objectius no funcionals

Més enllà de la funcionalitat, el sistema ha de complir diversos requisits de qualitat que determinaran la seva utilitat pràctica.

La cobertura ha de ser exhaustiva. El sistema ha de poder gestionar la majoria de casos reals: diferents tipologies d'habitacle (pisos, àtics, dúplex, habitatges unifamiliars), diversos perfils de sol·licitants (des de persones soles fins a famílies nombroses), i múltiples categories de serveis urbans. No es tracta de cobrir tots els casos possibles del món real, però sí els més freqüents i rellevants.

La mantenibilitat és crucial per a la viabilitat a llarg termini. L'estructura del sistema ha de ser modular, amb regles clarament separades per funcionalitat, de manera que afegir noves regles o modificar les existents sigui senzill i no requereixi reescriure grans parts del codi. La documentació del codi i de les regles ha de ser suficient perquè un nou desenvolupador pugui entendre i modificar el sistema.

El rendiment ha de ser adequat. Tot i que no es tracta d'un sistema de temps real crític, processar desenes d'ofertes per a diversos sol·licitants hauria de prendre segons, no minuts. Això implica que les regles han de ser eficients i que l'arquitectura del sistema no ha de tenir colls d'ampolla evidents.

Finalment, la transparència és un requisit essencial. Totes les decisions del sistema han de ser traçables: ha de ser possible seguir el raonament que ha portat a una recomanació concreta, identificar quines regles s'han activat, i entendre per què. Aquesta traçabilitat no només facilita la depuració i el manteniment, sinó que també és essencial per a la confiança dels

usuaris.

1.5 Abast del sistema

1.5.1 Dins de l'abast

Per definir clarament les expectatives i centrar l'esforç de desenvolupament, delimitem explícitament què està dins de l'abast del sistema.

El sistema gestionarà habitatges de lloguer (no de compra) en un àmbit geogràfic simplificat. Per facilitar el desenvolupament i les proves, treballarem amb una ciutat fictícia amb coordenades simplificades, tot i que el disseny permetrà adaptar-lo fàcilment a dades reals de Barcelona utilitzant latitud i longitud.

Els perfils de sol·licitants coberts inclouen les tipologies més comunes: individus sols, parelles (amb fills, sense fills, o amb plans de tenir-ne), famílies (monoparentals o biparentals), grups d'estudiants, i persones grans. Per a cada perfil, considerarem les variables més rellevants: nombre de persones, pressupost, necessitats d'accessibilitat, mascotes, vehicle propi, etc.

Pel que fa als serveis urbans, considerarem sis categories principals: transport públic (metro, bus, tren), serveis educatius (escoles, instituts, universitats, llars d'infants), serveis de salut (hospitals, centres de salut, farmàcies), serveis comercials (supermercats, hipermercats, mercats), zones verdes (parcs, jardins), i serveis d'oci. També identificant serveis potencialment molestos (discoteques, estadis, autopistes) que alguns perfils voldran evitar.

El sistema calcularà automàticament la proximitat entre habitatges i serveis utilitzant distàncies simplificades (Manhattan o Euclidiana), i classificarà aquesta proximitat en tres categories: molt a prop (menys de 500m, caminable en 5-7 minuts), a distància mitjana (500-1000m, caminable en 10-15 minuts), i lluny (més de 1000m, requereix transport).

Finalment, el sistema serà capaç d'inferir necessitats basant-se en el perfil demogràfic i aplicar regles heurístiques de sentit comú sobre la qualitat dels habitatges (per exemple, que un àtic sol ser millor que un entresòl, o que l'orientació solar a tot el dia és preferible).

1.5.2 Fora de l'abast

És igualment important delimitar què queda explícitament fora de l'abast del sistema, per evitar expectatives no realistes i centrar l'esforç en els objectius principals.

El sistema no gestionarà habitatges de compra, només de lloguer. Això simplifica significativament el problema, ja que no cal considerar hipoteques, impostos de compravenda, plusvàlues, ni anàlisis d'inversió a llarg termini.

No es farà anàlisi financer detallat més enllà de la verificació de pressupost. No calcularem la quota hipotecària òptima, ni analitzarem l'estalvi a llarg termini, ni farem previsions de preus futurs. El sistema es limita a verificar que el preu de lloguer estigui dins del pressupost del sol·licitant.

Tampoc implementarem recomanacions personalitzades amb aprenentatge d'usuari. El sistema no recordarà preferències d'usuaris anteriors, ni ajustarà els seus criteris basant-se en el

feedback rebut. Cada consulta es tracta de manera independent aplicant sempre les mateixes regles expertes.

La integració amb bases de dades reals d'ofertes queda fora de l'abast. Treballarem amb dades sintètiques o introducció manual d'ofertes. Un sistema real requeriria integració amb APIs de plataformes immobiliàries, actualització constant de la disponibilitat, i gestió de l'obsolescència de les dades.

Finalment, no implementarem funcionalitats de gestió de reserves o contractes. El sistema es limita a recomanar; la gestió posterior del lloguer (visites, negociació, signatura de contracte, pagaments) queda fora del seu abast.

1.6 Resultats esperats

1.6.1 Format de sortida

Per a cada sol·licitant que consulta el sistema, s'espera obtenir una sortida estructurada amb diversos components clarament diferenciats.

El primer component és una llista d'ofertes recomanades, classificades en tres nivells segons el seu grau d'adequació. Les ofertes Molt Recomanables són aquelles que compleixen tots els requisits obligatoris i, a més, destaquen per característiques excepcionals com un preu excel·lent, equipament superior, o localització privilegiada. Les ofertes Adequades compleixen tots els requisits sense excepcions, tot i que no tenen elements destacables addicionals. Les ofertes Parcialment Adequades són acceptables però presenten algun criteri no complert que, tot i no ser eliminatori, suposa un compromís (preu lleugerament superior, transport a distància mitjana en lloc de molt proper, etc.).

El segon component és una llista d'ofertes descartades amb la justificació de cada descart. És important no només excloure habitatges, sinó explicar per què: "No permet mascotes i el sol·licitant té un gos", "Preu supera el pressupost màxim establert", "No té ascensor i el sol·licitant requereix accessibilitat". Aquestes justificacions ajuden l'usuari a entendre les decisions del sistema i a modificar els seus criteris si ho considera oportú.

El tercer component són explicacions detallades per a cada recomanació. Per a les ofertes Molt Recomanables, cal llistar explícitament què les fa destacar: "Preu 20

1.6.2 Informació per oferta

Per a cada oferta recomanada o descartada, el sistema proporcionarà informació estructurada que inclou les característiques principals de l'habitatge (tipus, superfície, nombre i tipus de dormitoris, banys, planta, ascensor), la localització (adreça, districte, barri, coordenades), el preu mensual, l'equipament (moblat, electrodomèstics, calefacció, aire condicionat, terrassa, parking), i els serveis propers rellevants per al perfil del sol·licitant (escoles, transport, salut, comerç, zones verdes).

1.7 Beneficis esperats

1.7.1 Per als usuaris

El sistema aportarà diversos beneficis tangibles als sol·licitants d'habitatge que el utilitzin. L'estalvi de temps és potser el més immediat i valuós. En lloc de revisar manualment desenes o centenars d'anuncis, filtrant per preu i després verificant un per un la localització i els serveis propers, l'usuari rep directament una llista curada d'opcions adequades al seu perfil. Això pot reduir el temps de cerca de setmanes a hores.

La qualitat de les recomanacions serà superior a la que l'usuari obtindria per si mateix, especialment si no coneix bé la ciutat o no és expert en el mercat immobiliari. El sistema aplica coneixement expert acumulat i considera factors que l'usuari podria passar per alt (per exemple, que un barri concret no té escoles properes malgrat semblar ideal en altres aspectes).

La transparència de les decisions permet a l'usuari entendre per què se li recomana cada habitatge i què li manca als que s'han descartat. Això no només genera confiança en el sistema, sinó que també educa l'usuari sobre els factors rellevants que hauria de considerar, millorant la seva capacitat de prendre decisions informades.

Finalment, la capacitat d'inferència automàtica de necessitats millora significativament l'experiència d'usuari. No cal que l'usuari sigui exhaustiu especificant cada requisit; el sistema dedueix intel·ligentment què necessita basant-se en el seu perfil, estalviant-li temps i assegurant que no oblide factors importants.

1.7.2 Per als agents immobiliaris

El sistema també aporta valor als professionals del sector immobiliari, tot i que amb un enfocament diferent.

L'automatització del pre-filtratge és el benefici més evident. Els agents reben diàriament múltiples consultes que requereixen temps per analitzar. El sistema pot fer una primera criba automàtica, identificant les ofertes més prometedores per a cada client i deixant que l'agent dediqui el seu temps a afegir valor en les fases posteriors (negociació, visites, aspectes legals).

La millora de l'eficiència es tradueix en capacitat per atendre més clients simultàniament sense comprometre la qualitat del servei. El sistema actua com un assistent intel·ligent que aplica de manera consistent els criteris de l'agent, alliberant-lo per a tasques que requereixen el seu toc humà.

La consistència en els criteris és un altre avantatge. Mentre que un agent humà pot aplicar criteris lleugerament diferents segons l'estat d'ànim, la fatiga o la càrrega de treball, el sistema aplica sempre els mateixos estàndards, assegurant un servei homogeni a tots els clients.

1.7.3 Per als propietaris

Indirectament, els propietaris també es beneficien del sistema a través d'un millor matching amb llogaters adequats.

Si els habitatges es recomanen a perfils realment adequats, augmenta la probabilitat que la relació de lloguer sigui satisfactòria i duradora per a ambdues parts. Una família que lloga un habitatge ben situat respecte a escoles i serveis rellevants té menys probabilitats de canviar de pis al cap de pocs mesos.

La reducció de vacants és un altre benefici potencial. Si el sistema facilita que els llogaters adequats trobin ràpidament els habitatges que els convenen, es redueix el temps que aquests estan buits esperant inquilí.

1.8 Conclusions de la fase d'identificació

L'anàlisi realitzada en aquesta fase d'identificació demostra que el problema de recomanació d'habitatges de lloguer és adequat per ser abordat mitjançant un Sistema Basat en Coneixement. Hem identificat clarament l'existència de coneixement expert articulable, la complexitat moderada que justifica l'ús d'IA, i la necessitat d'explicabilitat que fa preferibles els SBC sobre altres aproximacions.

Les fonts de coneixement estan ben identificades, combinant experts humans (agents immobiliars, consultors d'habitacle), fonts documentals (plataformes online, estudis demogràfics, normatives legals), i eines modernes (models de llenguatge per a conceptualització inicial). Aquesta diversitat de fonts assegurarà que el coneixement codificat sigui robust i validat.

Els objectius, tant funcionals com no funcionals, estan clarament definits i són assolibles amb la tecnologia i metodologia proposades. L'abast del sistema està delimitat de manera realista, centrant-se en els aspectes essencials i deixant per a futures extensions funcionalitats més avançades.

Els beneficis esperats justifiquen l'esforç de desenvolupament: el sistema aportarà valor real a usuaris, agents immobiliars i propietaris, millorant l'eficiència del mercat de lloguer.

Amb aquesta base sòlida, podem avançar a la fase de conceptualització, on començarem a estructurar el coneixement identificat i a definir els conceptes, relacions i regles que conformaran el sistema.

2 Conceptualització

2.1 Introducció a la fase de conceptualització

La fase de conceptualització és el pont entre la identificació del problema i la seva formalització tècnica. En aquesta etapa, transformem el coneixement expert identificat en una estructura conceptual clara i coherent, definint els elements principals del domini i les seves interrelacions. L'objectiu és obtenir una visió compartida i no ambigua del problema que serveixi de base per a la posterior formalització en una ontologia i implementació en regles. Aquesta fase és crítica perquè determina l'abast i la qualitat del sistema final. Una conceptualització inadequada (incompleta, ambigua o mal estructurada) propagarà problemes a totes les fases posteriors, mentre que una conceptualització sòlida facilitarà tant la formalització com la implementació i el manteniment.

Per dur a terme aquesta conceptualització, hem combinat diverses fonts de coneixement. Primer, hem analitzat plataformes immobiliàries existents per identificar les característiques que el mercat considera rellevants. Segon, hem consultat models de llenguatge actuals (ChatGPT i Claude) com a "experts virtuals" per explorar el domini de manera sistemàtica. Tercer, hem aplicat el nostre propi coneixement i sentit comú per validar i completar la informació obtinguda. Finalment, hem contrastat el resultat amb la literatura sobre sistemes de recomanació d'habitatge i amb agents immobiliaris reals quan ha estat possible.

2.2 Conceptes principals del domini

2.2.1 Habitatge

El concepte central del nostre domini és l'habitatge, que representa una vivenda disponible per llogar. Un habitatge no és simplement un conjunt de metres quadrats, sinó una entitat complexa amb múltiples facetes que determinen la seva adequació per a diferents perfils d'usuaris.

Hem identificat diversos grups d'atributs que caracteritzen un habitatge. Els atributs físics bàsics inclouen la superfície habitable (expressada en metres quadrats), el nombre total de dormitoris, la distinció entre dormitoris dobles i simples (rellevant per determinar la capacitat real), i el nombre de banys (que afecta la comoditat, especialment per a famílies nombroses). La tipologia de l'habitatge és un atribut fonamental que afecta molts altres aspectes. Distingim entre pis (la opció més comuna), àtic (generalment preferit per la llum i les vistes), dúplex (amb dos nivells), estudi (opció compacta per a individus o parelles), i habitatge unifamiliar (ideal per a famílies grans que busquen independència). Cada tipologia té implicacions sobre altres característiques: per exemple, els àtics solen tenir terrassa gran, mentre que els estudis són compactes i econòmics.

L'equipament determina el nivell de comoditat i afecta especialment perfils com estudiants o persones que es traslladen temporalment. Considerem si l'habitatge està moblat (essencial

per a estudiants i expatriats), si inclou electrodomèstics (nevera, rentadora, forn), si disposa de calefacció i aire condicionat (cada cop més valorats), i l'estat de conservació general (nou, bon estat, necessita reformes).

Les característiques arquitectòniques i de situació dins l'edifici també són rellevants. La planta on es troba l'habitatge (entresòl, primer, segon, àtic) i la presència o absència d'ascensor determinen l'accessibilitat. L'orientació solar (matí, tarda, tot el dia) afecta la lluminositat i l'eficiència energètica. Si l'habitatge és exterior o interior condiciona el soroll i la llum natural. La presència de terrassa o balcó i la seva superfície augmenten significativament l'atractiu. Les vistes (mar, muntanya, ciutat, o cap) són un plus valorat.

Les restriccions i condicions de lloguer inclouen si l'habitatge permet mascotes (eliminator per a moltes famílies), el consum energètic (de A a G, afectant les factures mensuals), i el nivell de soroll de la zona (baix, mitjà, alt).

Finalment, els extres o amenities poden ser determinants per a certs perfils: plaça d'aparcament (essencial si tens vehicle), piscina comunitària (atractiva per a famílies amb fills), traster (útil per a emmagatzematge addicional), i armaris encastats (alliberen espai habitable).

2.2.2 Localització

La localització és un concepte aparentment simple però fonamental, ja que connecta l'habitatge amb els serveis urbans i determina gran part de la seva adequació.

Una localització es caracteritza per diversos nivells de granularitat. Al nivell més general, tenim el districte i el barri, que proporcionen context urbà i socioeconòmic. A nivell més específic, tenim l'adreça completa (carrer i número) i el codi postal. Finalment, per a càlculs de distància, utilitzem coordenades geogràfiques.

En la nostra implementació simplificada, utilitzem un sistema de coordenades cartesià 2D (X, Y) que representa metres des d'un punt d'origen arbitrari a la ciutat. Això facilita els càlculs de distància sense la complexitat de treballar amb latitud/longitud. No obstant això, el disseny és prou flexible per adaptar-se fàcilment a coordenades GPS reals si es desitja integrar amb dades reals.

La localització és el punt de contacte entre l'habitatge i els serveis: cada habitatge té una localització, cada servei té una localització, i calculem la distància entre elles per determinar la proximitat.

2.2.3 Servei

Els serveis urbans són elements de la ciutat que afecten la qualitat de vida dels residents i, per tant, l'adequació d'un habitatge per a un perfil determinat. La seva proximitat a l'habitatge pot ser un factor obligatori, preferent, o fins i tot negatiu segons el tipus de servei i el perfil del sol·licitant.

Hem identificat sis categories principals de serveis, cadascuna amb múltiples subtipus.

Els serveis de transport públic són crucials en una ciutat gran. Inclouen estacions de metro (la forma més ràpida de transport urbà), parades de bus (donen més capilaritat), estacions de

tren (per a connexions interurbanes), i en alguns casos, estacions de tramvia. La proximitat al transport públic és especialment important per a estudiants, persones grans, i qualsevol que no tingui vehicle propi.

Els serveis educatius són fonamentals per a famílies amb fills. Distingim entre llars d'infants (0-3 anys), escoles primàries (6-12 anys), instituts (12-18 anys), i universitats (18+ anys). No totes les famílies necessiten tots els tipus: una família amb un nadó prioritzarà llars d'infants, mentre que una amb adolescents buscarà instituts propers.

Els serveis de salut inclouen hospitals (per a urgències i especialitats), centres de salut o CAPs (per a atenció primària rutinària), i farmàcies (per a medicació i consultes menors). Són especialment importants per a persones grans i famílies amb fills petits.

Els serveis comercials són necessaris per a la vida quotidiana. Inclouen supermercats (compra setmanal o diària), hipermercats (compra mensual de gran volum), mercats municipals (productes frescos), i centres comercials (compres i oci). Les persones grans prefereixen supermercats propers on fer compra diària, mentre que les famílies poden preferir hipermercats per a compra mensual.

Les zones verdes són espais per a l'esbarjo i l'exercici. Inclouen parcs (grans extensions per a passejada i esport), jardins (espais més petits i tranquil·ls), i zones esportives (amb instal·lacions específiques). Són particularment rellevants per a famílies amb fills (necessiten espais de joc) i persones amb mascotes.

Els serveis d'oci enriqueixen la vida social i cultural. Inclouen gimnasos, biblioteques, centres culturals, cinemes, teatres, restaurants i bars. Alguns d'aquests (bars, discoteques) poden generar soroll i ser considerats negatius per certs perfils.

Finalment, hem identificat una categoria especial de serveis potencialment molestos que alguns perfils voldran evitar: discoteques (soroll nocturn), estadis esportius (aglomeracions i soroll en dies de partit), zones industrials (contaminació i trànsit pesat), autopistes (soroll constant), i aeroports (soroll de vols). La proximitat a aquests serveis pot ser un factor d'exclusió per a certs sol·licitants.

2.2.4 Sol·licitant

El sol·licitant és la persona o grup de persones que busca habitatge. És el concepte més complex i divers, ja que diferents perfils tenen necessitats radicalment diferents.

Hem identificat set tipologies principals de sol·licitants, organitzades jeràrquicament.

L'individu representa una persona sola, que pot ser un professional jove, un divorçat, o qualsevol que visqui independentment. Generalment tenen necessitats senzilles (superficie moderada, preu ajustat) però poden ser exigents amb la localització segons si treballen o estudien.

La parella sense fills és una unitat de dues persones adultes. Solen prioritzar la qualitat de vida (terrassa, bona zona) sobre l'espai, ja que no necessiten múltiples dormitoris. Poden permetre's gastar més per persona que un individu sol.

La parella amb plans de fills és un cas especial: actualment són dues persones, però preveuen ampliar la família en un futur pròxim (1-2 anys). Busquen habitatges amb "potencial de

creíxer”, preferint 3 dormitoris a 2, i valoren la proximitat a escoles i zones verdes tot i que encara no les necessiten.

La família amb fills és una de les tipologies més exigents. Distingim entre família biparental (dos adults) i monoparental (un adult), ja que aquest factor afecta el pressupost disponible. El nombre i les edats dels fills determinen necessitats específiques: famílies amb nadons necessiten llars d’infants, amb infants necessiten escoles primàries i zones de joc, amb adolescents necessiten instituts i centres culturals.

El grup d’estudiants representa dues o més persones joves que comparteixen habitatge mentre estudien. Tenen requisits molt específics: pressupost ajustat (repartit entre tots), habitatge moblat (per no haver d’invertir en mobles), proximitat a la universitat i transport públic, i tolerància al soroll (sovint viuen en zones d’ambient jove).

La persona gran (>65 anys) té necessitats molt específiques relacionades amb la mobilitat i la salut. Requereixen accessibilitat (ascensor si no és planta baixa, absència de barreres arquitectòniques), proximitat a serveis de salut (hospitals, CAPs, farmàcies), i comerços propers per a compra diària (sovint no tenen vehicle). També valoren la tranquil·litat i la seguretat del barri.

Per a tots els perfils, considerem atributs comuns que afecten les necessitats. El pressupost (màxim i mínim) és sempre un factor crític, així com si el marge és estricte o flexible. El nombre de persones determina l’espai mínim necessari. La possessió de mascotes (tipus i nombre) és un factor eliminador si l’habitatge no les permet. La possessió de vehicle fa valorar el parking inclòs. Les necessitats d’accessibilitat són crítiques per a persones amb mobilitat reduïda. La preferència per transport públic afecta la importància de la seva proximitat. El fet de treballar o estudiar a la ciutat condiciona la necessitat de bona connexió.

2.2.5 Oferta

Una oferta vincula un habitatge concret amb condicions específiques de lloguer i amb l’avaluació que el sistema en fa per a cada sol·licitant.

Els atributs bàsics d’una oferta inclouen l’habitacle al qual fa referència (amb totes les seves característiques), el preu mensual de lloguer, la disponibilitat (si està llogada o disponible), i la data de publicació (opcionalment, per identificar ofertes antigues).

Els atributs calculats pel sistema són els més interessants. El grau de recomanació (Molt Recomanable, Adequat, Parcialment Adequat, Descartat) és l’avaluació principal que el sistema fa de l’oferta per a un sol·licitant concret. Els motius de recomanació expliquen per què s’ha assignat aquest grau: quins són els punts forts (per a Molt Recomanables), quins criteris no es compleixen (per a Parcialment Adequats), o per què s’ha descartat (per a Descartats).

És important notar que el grau de recomanació i els motius són específics per a cada parella sol·licitant-oferta: el mateix habitatge pot ser Molt Recomanable per a una família amb fills i Descartat per a estudiants, o viceversa.

2.3 Relacions entre conceptes

Una vegada identificats els conceptes principals, cal definir com es relacionen entre ells. Aquestes relacions estructuren el coneixement del domini i determinaran la navegació i el raonament del sistema.

2.3.1 Relacions d'associació bàsiques

La relació més simple és `teLocalitzacio`, que connecta tant habitatges com serveis amb la seva localització geogràfica. És una relació funcional (cada habitatge/servei té exactament una localització) i obligatòria (no pot haver-hi habitatges o serveis sense localització).

La relació `teHabitatge` connecta cada oferta amb l'habitatge al qual fa referència. També és funcional i obligatòria: cada oferta referencia exactament un habitatge.

2.3.2 Relacions de proximitat

La relació `aPropDe` connecta localitzacions amb serveis i és fonamental per al raonament del sistema. No és binària (`proper/lluny`), sinó que distingim tres nivells de proximitat basats en la distància: `moltAPropDe` (menys de 500 metres, caminable en 5-7 minuts), `aDistanciaMitjana` (500-1000 metres, caminable en 10-15 minuts), i `llunyde` (més de 1000 metres, generalment requereix transport).

Aquesta classificació en tres nivells permet un raonament més ric que una simple dicotomia. Per exemple, una escola a distància mitjana pot ser acceptable per a una família, mentre que un hospital lluny pot ser problemàtic per a una persona gran.

La relació de proximitat es calcula automàticament a partir de les coordenades de les localitzacions, utilitzant la distància de Manhattan o Euclidiana. No es pre-calcula per a totes les parelles possibles (seria ineficient), sinó que es computa dinàmicament quan es necessita.

2.3.3 Relacions de preferència i restricció

Els sol·licitants poden expressar relacions amb serveis de tres tipus, que tenen interpretacions diferents en el raonament.

La relació `requereixServei` indica que el sol·licitant necessita obligatòriament aquest tipus de servei proper. Si un habitatge no té aquest servei a distància proper o mitjana, quedarà descartat. Per exemple, una família amb fills petits requereix escoles; un estudiant requereix transport públic.

La relació `prefereixServei` indica que el sol·licitant valora positivament aquest servei, però no és eliminador si no està disponible. La seva proximitat millorarà la valoració de l'oferta, però la seva absència no la descartarà. Per exemple, una parella sense fills pot preferir zones verdes, però no és obligatori.

La relació `evitaServei` indica que el sol·licitant no vol viure prop d'aquest tipus de servei (generalment serveis molestos). Si un habitatge està molt a prop d'un servei evitat, quedarà descartat o severament penalitzat. Per exemple, una persona gran pot voler evitar discoteques; una família amb nadós pot voler evitar estadis.

2.3.4 Relacions espacials addicionals

Opcionalment, els sol·licitants poden especificar llocTreball i llocEstudi, que són localitzacions específiques on passen temps regularment. Si s'especifiquen, el sistema pot valorar positivament habitatges ben connectats amb aquests llocs, ja sigui per proximitat directa o per proximitat a transport públic que els connecti.

2.4 Ús de models de llenguatge per a la conceptualització

Una de les innovacions metodològiques d'aquest projecte ha estat l'ús de models de llenguatge (ChatGPT 4 i Claude 3.5 Sonnet) com a "experts virtuals" per a la fase de conceptualització. Aquesta aproximació mereix una explicació detallada del procés, els resultats i les limitacions.

2.4.1 Metodologia d'ús dels models de llenguatge

Per utilitzar efectivament un model de llenguatge com a expert, cal primer posar-lo en context adequadament. Hem utilitzat el següent prompt de contextualització:

```
1 You are an experienced real estate agent specializing in rental  
2 properties in Barcelona. You help match different types of people  
3 (families with children, students, couples, elderly) with suitable  
4 homes based on their needs and the neighborhood characteristics.  
5  
6 The knowledge engineer will ask you about the characteristics,  
7 criteria, and knowledge you use for making decisions. Your answers  
8 should be direct and precise, helpful for building rules in an  
9 expert system. Focus on providing concrete, actionable knowledge  
10 rather than general statements.
```

Aquest prompt estableix clarament el rol (agent immobiliari expert), el domini (lloguer d'habitatges a Barcelona), l'audiència (enginyer del coneixement), i el format de resposta desitjat (coneixement concret i accionable).

Un cop establert el context, hem fet preguntes progressivament més específiques per extreure coneixement estructurat. Aquest procés s'anomena "elicitació de coneixement" i segueix principis similars als utilitzats en entrevistes amb experts humans.

2.4.2 Exemples de preguntes i respostes

A continuació presentem alguns exemples representatius del diàleg amb el model, il·lustrant el tipus de coneixement extret.

Pregunta 1: Característiques del barri per a famílies *Question: What are the most important neighborhood characteristics for a family with two young children (ages 3 and 6)?*
Resposta sintetitzada del model:

- Primary schools within 500 metres (walking distance)

- Parks or playgrounds within 500 metres
- Pedestrian-friendly streets (low traffic, wide sidewalks)
- Healthcare centres (at least a CAP) within 1 km
- Supermarkets for daily shopping within 500 metres
- Low noise levels (avoid nightlife areas, stadiums, major roads)
- Public transport accessible but not mandatory if they have a car
- Daycare centres (llar d'infants) within 500 metres if both parents work

Aquesta resposta ens proporciona diversos inputs valuosos: distàncies concretes (500m per a serveis quotidiàs, 1km per a serveis menys freqüents), una jerarquia de necessitats (escola i zones verdes són prioritàries), i factors negatius a evitar (soroll, trànsit intens). També identifica factors condicionals (la llar d'infants només és rellevant si ambdós pares treballen).

Pregunta 2: Requisits d'espai segons composició familiar *Question: What minimum size requirements would you recommend for different household sizes?*

Resposta sintetitzada del model:

- Single person: 30-40 m² (studio acceptable, 1 bedroom preferred)
- Couple: 50-60 m² (1-2 bedrooms)
- Couple + 1 child: 65-80 m² (minimum 2 bedrooms)
- Couple + 2 children: 80-100 m² (minimum 3 bedrooms)
- Family of 5+: 100+ m² (minimum 4 bedrooms)
- Group of 3 students: 70-90 m² (3 bedrooms, shared spaces OK)

Aquestes xifres concretes són extremadament útils per definir regles de descart (un habitatge massa petit per al nombre de persones). També identifiquen diferències subtils: per exemple, estudiants toleren menys espai per persona que famílies perquè passen menys temps a casa.

Pregunta 3: Priorització amb pressupost ajustat *Question: When a family has a strict budget and cannot afford everything they want, how do you advise them to prioritize between location, size, and amenities?*

Resposta sintetitzada del model:

The general priority order with tight budgets is:

1. Basic space requirements (minimum bedrooms and total size)
2. Essential services for their profile (schools for families, transport for students)

3. Location/neighborhood safety
4. Extras and amenities (these can wait)

Budget flexibility of 10-15% is reasonable if location and essential services are significantly better. Beyond 15%, the financial strain outweighs the benefits.

Aquesta resposta ens dóna una jerarquia de prioritats que podem traduir en pesos relatius en la nostra funció d'avaluació. També proporciona un threshold concret (15%) per a la flexibilitat de pressupost.

Pregunta 4: Criteris per a persones grans *Question: What are the non-negotiable requirements for elderly people (65+) when renting?*

Resposta sintetitzada del model:

- Elevator if above ground floor (absolutely critical)
- No architectural barriers (steps, narrow doors)
- Healthcare centres within walking distance (500-700m max)
- Daily shopping (small supermarket or market) within 500m
- Ground floor or low floor with elevator strongly preferred
- Quiet neighborhood (they're home more hours than working people)
- Good natural light (affects mood and health)
- Avoid: steep hills, stairs to building entrance, isolated areas

Aquesta resposta identifica requisits obligatoris (ascensor si no és planta baixa) versus preferències fortes (planta baixa), i també factors negatius específics per aquest col·lectiu (pendents pronunciats, escales a l'entrada).

2.4.3 Validació i integració del coneixement obtingut

El coneixement extret dels models de llenguatge no s'ha acceptat acríticament, sinó que s'ha validat i integrat seguint diverses estratègies.

Primer, hem contrastat les respostes entre diferents models (ChatGPT i Claude) per identificar consens i discrepàncies. El coneixement consistent entre models té més probabilitats de ser correcte.

Segon, hem comparat el coneixement extret amb fonts autoritzades: normatives d'accessibilitat per a persones grans, estudis sociològics sobre preferències habitacionals, i informació de plataformes immobiliàries reals.

Tercer, hem aplicat el sentit comú i l'experiència personal per detectar recomanacions poc realistes o específiques d'altres contextos geogràfics (per exemple, alguns models donaven consells més adients per a ciutats americanes que per a Barcelona).

Quart, hem simplificat i adaptat el coneixement al nivell de granularitat adequat per al nostre sistema. Per exemple, el model podria distingir entre 5 tipus de zones verdes, però nosaltres hem decidit treballar només amb 3 categories.

Finalment, hem documentat clarament en l'informe quan una decisió prové d'un model de llenguatge, permetent la traçabilitat i facilitant futures revisions si es detecten errors.

2.4.4 Limitacions i consideracions crítiques

L'ús de models de llenguatge per a conceptualització té limitacions que cal reconèixer explícitament.

Primer, els models poden "al·lucinar" (inventar) informació plausible però incorrecta. Per exemple, poden citar estudis inexistentes o proporcionar estadístiques fabricades. Per això, tota informació factual o quantitativa s'ha de verificar amb fonts fiables.

Segon, els models poden reflectir biaixos presents en les seves dades d'entrenament. Per exemple, poden assumir estructures familiars tradicionals o fer suposicions culturalment específiques. Hem estat atents a aquestes possibles biaixos i els hem corregit quan els hem detectat.

Tercer, els models no tenen experiència real ni comprensió profunda del domini. Poden proporcionar coneixement "de llibre" que un expert real amb anys d'experiència matisaria o contradaria. Per això, el coneixement extret s'ha complementat amb altres fonts.

Quart, els models són millors proporcionant coneixement general que coneixement específic d'un context local. Per exemple, poden donar bones recomanacions sobre necessitats de famílies amb fills (universal), però ser menys fiables sobre característiques específiques de barris de Barcelona.

Malgrat aquestes limitacions, l'ús de models de llenguatge ha estat valuós per a la fase de conceptualització, especialment per:

- Accelerar l'exploració inicial del domini
- Identificar factors que podríem haver passat per alt
- Proporcionar estructuracions alternatives del coneixement
- Generar exemples concrets i casos d'ús
- Validar la coherència de les regles proposades

2.5 Regles heurístiques identificades

A partir de l'elicitació de coneixement (tant de models de llenguatge com d'altres fonts), hem identificat un conjunt de regles heurístiques que guiaran la implementació del sistema. Aquestes regles es poden agrupar en diverses categories.

2.5.1 Regles basades en perfil demogràfic

Aquestes regles infereixen necessitats a partir de la tipologia i característiques del sol·licitant.

Regles per a famílies amb fills:

- SI numeroFills > 0 LLAVORS necessita escoles a distància propera o mitjana (obligatori)
- SI numeroFills > 0 LLAVORS prefereix zones verdes properes
- SI edatFills conté valors < 3 LLAVORS necessita llar d'infants propera
- SI numeroFills > 2 LLAVORS evita zones de soroll nocturn
- SI numeroFills > 0 LLAVORS necessita mínim (1 + numeroFills) dormitoris
- SI numeroFills > 2 LLAVORS prefereix habitatge unifamiliar o dúplex

Regles per a estudiants:

- SI és grup d'estudiants LLAVORS necessita transport públic molt proper (obligatori)
- SI és grup d'estudiants LLAVORS necessita habitatge moblat (obligatori)
- SI és grup d'estudiants LLAVORS prioritza pressupost per sobre d'altres factors
- SI és grup d'estudiants LLAVORS prefereix zones amb ambient jove
- SI és grup d'estudiants LLAVORS tolera nivell de soroll mitjà-alt

Regles per a persones grans:

- SI edat > 65 I plantaPis > 0 LLAVORS necessita ascensor (obligatori)
- SI edat > 65 LLAVORS necessita serveis de salut propers (obligatori)
- SI edat > 65 LLAVORS necessita comerç d'alimentació proper (obligatori)
- SI edat > 65 LLAVORS evita zones de soroll alt
- SI edat > 65 LLAVORS prefereix planta baixa o primera
- SI edat > 65 LLAVORS valora molt la llum natural

Regles per a parelles sense fills:

- SI és parella sense fills LLAVORS flexibilitat en localització (no hi ha restriccions escolars)
- SI és parella sense fills LLAVORS valora qualitat de vida (terrassa, vistes)

- SI és parella sense fills LLAVORS pot prioritzar preu sobre espai

Regles per a parelles amb plans de fills:

- SI és parella futurs fills LLAVORS prefereix escoles properes (no obligatori, però valora positivament)
- SI és parella futurs fills LLAVORS prefereix mínim 3 dormitoris (espai per créixer)
- SI és parella futurs fills LLAVORS prefereix zones verdes properes

2.5.2 Regles de sentit comú sobre habitatges

Aquestes regles expressen coneixement general sobre la qualitat relativa dels habitatges, independent del perfil del sol · licitant.

Regles sobre tipologia i planta:

- Àtic > Pis alt > Pis intermedi > Pis baix > Entresòl (en termes de llum i vistes)
- Habitatge amb ascensor > Sense ascensor (especialment en plantes altes)
- Habitatge exterior > Interior (llum natural i ventilació)

Regles sobre orientació i llum:

- Orientació tot el dia > Orientació tarda > Orientació matí
- Habitatge amb terrassa/balcó > Sense terrassa
- Habitatge amb vistes > Sense vistes

Regles sobre estat i equipament:

- Habitatge nou o reformat > Bon estat > Necessita reformes
- Amb calefacció i aire condicionat > Només un > Cap dels dos
- Consum energètic A o B > C o D > E, F o G

Regles sobre extras:

- SI té vehicle LLAVORS valora molt positivament parking inclòs
- Piscina comunitària és un plus per a famílies amb fills
- Traster és útil per a famílies grans o emmagatzematge

2.5.3 Regles de proximitat i distància

Aquestes regles tradueixen distàncies físiques en qualificacions qualitatives.

Classificació de distàncies:

- Molt a prop: < 500 m (caminable en 5-7 minuts)
- Distància mitjana: 500-1000 m (caminable en 10-15 minuts)
- Lluny: > 1000 m (generalment requereix transport)

Adequació segons tipus de servei i perfil:

- Per a famílies amb fills petits: escola ha d'estar molt a prop o distància mitjana
- Per a persones grans: serveis bàsics (salut, comerç) han d'estar molt a prop
- Per a estudiants: transport públic ha d'estar molt a prop
- Serveis molestos: si estan molt a prop, descarten l'habitatge

2.5.4 Regles de pressupost

Gestió de pressupost estricte:

- SI margeEstricta = si I preu > pressupostMaxim LLAVORS descarta
- SI margeEstricta = si I preu < pressupostMinim LLAVORS descarta (sospitos)

Gestió de pressupost flexible:

- SI margeEstricta = no I preu pressupostMaxim * 1.15 LLAVORS acceptable amb advertència
- SI margeEstricta = no I preu > pressupostMaxim * 1.15 LLAVORS descarta
- SI preu < pressupostMaxim * 0.8 LLAVORS punt molt positiu (estalvi significatiu)

2.6 Descomposició del problema en subproblemes

Un cop identificats els conceptes, relacions i regles del domini, cal estructurar el procés de resolució del problema. Seguint la metodologia de sistemes experts, descomponem el problema principal en subproblemes més tractables que es resoldran seqüencialment.

2.6.1 Problema principal

Entrada: Conjunt de sol·licitants amb les seves característiques, conjunt d'ofertes d'habitatges amb les seves característiques, conjunt de serveis urbans amb les seves localitzacions.

Sortida: Per a cada sol·licitant, una llista d'ofertes classificades segons el seu grau d'adequació (Molt Recomanable, Adequat, Parcialment Adequat, Descartat), amb explicacions detallades per a cada classificació.

Procés: Avaluar cada parella (sol·licitant, oferta) considerant tots els factors rellevants i aplicant el coneixement expert codificat en forma de regles.

2.6.2 Subproblema 1: Abstracció (Inferència de necessitats)

Entrada: Perfil d'un sol·licitant amb les seves característiques bàsiques (edat, nombre de persones, fills, mascotes, vehicle, etc.)

Tasca: Inferir necessitats i preferències implícites a partir del perfil demogràfic, aplicant regles heurístiques del domini.

Sortida: Conjunt de requisits inferits, cadascun amb una categoria de servei, un nivell d'obligatorietat (obligatori vs preferent), i una justificació.

Mètode: Aplicació de regles de classificació heurística seguida de regles d'inferència específiques per cada tipologia identificada.

Exemple:

- Entrada: Sol·licitant família biparental, 4 persones, 2 fills (edats 6 i 10 anys), té gos
- Procés: Regla detecta numeroFills > 0 → infereix necessitat d'escoles (obligatori)
- Procés: Regla detecta numeroFills > 0 → infereix preferència per zones verdes
- Procés: Regla detecta teMascotes = si → afegeix restricció permetMascotes obligatori
- Sortida: (ServeiEducatiu, obligatori, "Família amb fills necessita escoles"), (ZonaVerda, preferent, "Família amb fills prefereix zones verdes"), (permetsMascotes, obligatori, "Té mascota tipus Gos")

2.6.3 Subproblema 2: Resolució (Avaluació i filtratge)

Entrada: Una parella (sol·licitant amb requisits inferits, oferta), localitzacions de tots els serveis.

Tasca: Avaluar l'oferta contra els requisits del sol·licitant, identificant incompatibilitats eliminadores, criteris no complerts, i punts positius destacables.

Sortida: O bé un motiu de descart (si hi ha incompatibilitat eliminadora), o bé dues llistes: criteris no complerts (amb gravetat) i punts positius.

Mètode: Aplicació seqüencial de tres tipus de regles:

1. Regles de descart (eliminen ofertes clarament inadequades)

2. Regles de detecció de criteris no complerts (identifiquen febleses no eliminadores)
3. Regles de detecció de punts positius (identifiquen fortaleses)

Aquest subproblema es subdivideix en tres fases:

Fase 2.1: Càlcul de proximitats

- Calcular distàncies entre l'habitatge de l'oferta i tots els serveis rellevants
- Classificar cada distància en: MoltAProp, DistanciaMitjana, Lluny
- Emmagatzemar aquestes proximitats per a ús posterior

Fase 2.2: Filtratge obligatori Aplicar regles de descart en aquest ordre de prioritat:

1. Pressupost: Si el preu excedeix els límits establerts → descarta
2. Mascotes: Si el sol · licitant té mascotes i l'habitatge no les permet → descarta
3. Accessibilitat: Si el sol · licitant necessita accessibilitat i l'habitatge no la té → descarta
4. Superfície: Si l'habitatge és massa petit per al nombre de persones → descarta
5. Serveis obligatoris: Si falta algun servei requerit a distància adequada → descarta
6. Serveis evitats: Si hi ha algun servei evitat molt a prop → descarta
7. Mobilat: Si és grup d'estudiants i l'habitatge no està moblat → descarta

Fase 2.3: Avaluació de criteris Per a ofertes no descartades, detectar:

- Criteris no complerts: Preu lleugerament alt, soroll, sense ascensor (persona gran), servei preferent lluny, etc.
- Punts positius: Bon preu, terrassa, assolellat, alta eficiència energètica, transport molt proper, compleix requisits inferits, etc.

Exemple:

- Entrada: Família Garcia (requisits inferits del Subproblema 1), Oferta Pis Eixample (95m², 3 dorm, 1350€, permete mascotes, escola a 300m, parc a 450m, metro a 250m)
- Fase 2.1: Calcula proximitats → escola MoltAProp, parc MoltAProp, metro MoltAProp
- Fase 2.2: Verifica descartos → Preu OK ($1350 < 1500$), Mascotes OK, Superfície OK, Serveis obligatoris OK → No descarta
- Fase 2.3: Detecta criteris → Cap negatiu

- Fase 2.3: Detecta positius → Escola molt propera (compleix requisit inferit), Zona verda molt propera (compleix requisit inferit), Transport molt proper, Terrassa, Preu bo
- Sortida: CriterisNoComplerts=[], PuntsPositius=[5 elements]

2.6.4 Subproblema 3: Refinació (Classificació final)

Entrada: Una parella (sol·licitant, oferta) amb el resultat de l'avaluació (criteris no complerts i punts positius).

Tasca: Assignar un grau de recomanació global basant-se en l'evidència acumulada.

Sortida: Grau de recomanació (Molt Recomanable, Adequat, Parcialment Adequat) i puntuació numèrica opcional per ordenació dins de cada categoria.

Mètode: Aplicació de regles de classificació basades en comptadors d'evidència positiva i negativa.

Regles de classificació:

- SI (criteris no complerts = 0) I (punts positius >= 3) LLAVORS Molt Recomanable
- SI (criteris no complerts = 0) I (punts positius < 3) LLAVORS Adequat
- SI (criteris no complerts = 1) I (criteris no complerts = 2) LLAVORS Parcialment Adequat
- SI (criteris no complerts > 2) LLAVORS Descartat (reclassificat)

Exemple:

- Entrada: Família Garcia + Oferta Eixample + Avaluació (0 criteris negatius, 5 punts positius)
- Procés: Aplica regla: 0 negatius I 5 positius (3) → Molt Recomanable
- Sortida: (MoltRecomanable, puntuació=100)

2.7 Exemples de resolució manual

Per validar la conceptualització i il·lustrar com funciona el raonament expert, presentem dos exemples de resolució manual completa seguint la descomposició en subproblemes.

2.7.1 Exemple 1: Família Garcia

Perfil del sol·licitant:

- Nom: Família Garcia
- Tipologia: Família Biparental
- Nombre de persones: 4 (2 adults + 2 fills edats 6 i 10 anys)
- Pressupost: màxim 1500 EUR/mes, mínim 600 EUR/mes, marge no estricta
- Mascotes: sí, 1 gos
- Vehicle: sí

Ofertes disponibles:

1. Pis Eixample: 95m², 3 dorm, 2 banys, 1350 EUR, permet mascotes, escola 300m, parc 400m, metro 250m
2. Àtic Gràcia: 120m², 3 dorm, 2 banys, 1800 EUR, NO permet mascotes, escola 600m
3. Estudi Sants: 35m², 1 dorm, 650 EUR, permet mascotes

Resolució Oferta 1 (Pis Eixample):

Subproblema 1 - Abstracció:

- Detecta: numeroFills = 2 > 0 → Infereix: (ServeiEducatiu, obligatori, "Fills necessiten escola"), (ZonaVerda, preferent, "Fills prefereixen zones verdes")
- Detecta: teMascotes = si → Afegeix: (permetsMascotes, obligatori, "Té gos")

Subproblema 2 - Resolució:

- Fase 2.1: Calcula proximitats → escola MoltAProp (300m), parc MoltAProp (400m), metro MoltAProp (250m)
- Fase 2.2: Verifica descartos:
 - Preu: 1350 < 1500 → OK
 - Mascotes: permet = si → OK
 - Superfície: 95m² 4*10 = 40m² → OK (àmpliació suficient)
 - Serveis obligatoris: escola MoltAProp → OK
- Fase 2.3: Detecta punts positius:
 - Escola molt propera (compleix requisit obligatori inferit)
 - Zona verda molt propera (compleix preferència inferida)

- Transport molt proper
- Té terrassa (assumit per la descripció)
- Preu bo ($1350 / 1500 = 90\%$, estalvi del 10%)
- No detecta criteris negatius

Subproblema 3 - Refinació:

- Avaluació: 0 criteris negatius, 5 punts positius
- Regla: 0 negatius I 5 – 3 → Molt Recomanable
- Resultat: (MoltRecomanable, 100)

Resolució Oferta 2 (Àtic Gràcia):

Subproblema 2 - Resolució (usa els mateixos requisits inferits):

- Fase 2.2: Verifica descartos:
 - Preu: $1800 > 1500$, però marge no estricta. $1800/1500 = 1.2$ (20% excés). Si límit flexible és 15%, $1500 * 1.15 = 1725 < 1800 \rightarrow$ Supera límit flexible → DESCARTA
 - Motiu de descart: "Preu supera en un 20% el pressupost màxim (límit flexible 15%)" o "No permet mascotes i el sol·licitant té un gos"

No cal Subproblema 3 perquè ja està descartada.

Resolució Oferta 3 (Estudi Sants):

Subproblema 2 - Resolució:

- Fase 2.2: Verifica descartos:
 - Superfície: $35m^2 < 4 * 10 = 40m^2 \rightarrow$ MASSA PETIT → DESCARTA
 - Motiu de descart: "Superficie insuficient per a 4 personnes ($35m^2 < 40m^2$ mínim estimat)"

Resultats finals per Família Garcia:

- Molt Recomanable: Pis Eixample (100 punts)
- Descartades: Àtic Gràcia (preu excessiu i no mascotes), Estudi Sants (massa petit)

2.7.2 Exemple 2: Grup d'estudiants Marc i companys

Perfil del sol·licitant:

- Nom: Marc i companys
- Tipologia: GrupEstudiants
- Nombre de persones: 3
- Pressupost: màxim 900 EUR/mes, marge estricta
- Vehicle: no
- Prefereix transport públic: sí
- Estudia a la ciutat: sí

Ofertes disponibles:

1. Pis compartit Gràcia: 95m², 4 dorm, 1400 EUR, moblat, metro 200m
2. Estudi Gràcia: 30m², 1 dorm, 750 EUR, moblat, metro 150m

Resolució Oferta 1 (Pis compartit):

Subproblema 1 - Abstracció:

- Detecta: GrupEstudiants → Infereix: (Transport, obligatori, "Estudiants necessiten transport"), (moblat, obligatori, "Estudiants necessiten moblat")

Subproblema 2 - Resolució:

- Fase 2.2: Verifica descartos:
 - Preu: 1400 > 900 i marge estricta → DESCARTA
- Motiu: "Preu supera pressupost màxim amb marge estricta"

Resolució Oferta 2 (Estudi):

Subproblema 2 - Resolució:

- Fase 2.1: metro MoltAProp (150m)
- Fase 2.2: Verifica descartos:
 - Preu: 750 < 900 → OK
 - Moblat: sí → OK
 - Superfície: 30m² vs 3 persones. Mínim estimat: 3*10=30m² → JUST però OK
 - Transport: metro MoltAProp → OK

- Fase 2.3: Punts positius:
 - Transport molt proper (compleix requisit)
 - Moblat (compleix requisit)
 - Preu bo ($750/900 = 83\%$, dins pressupost)
- Possible criteri negatiu: Espai just per 3 personnes ($30m^2$ és el mínim)

Subproblema 3 - Refinació:

- Avaluació: 0-1 criteris negatius (segons si considerem l'espai just com a negatiu), 3 punts positius
- Si 0 negatius: Molt Recomanable
- Si 1 negatiu (espai just): Parcialment Adequat
- Decisió: Considerem que $30m^2$ per 3 estudiants és acceptable (passen poc temps a casa)
 \rightarrow 0 negatius \rightarrow Adequat (3 punts positius, però no excepcionals per Molt Recomanable)

Resultats finals per Marc i companys:

- Adequat: Estudi Gràcia (70 punts)
- Descartades: Pis compartit (preu excessiu amb marge estricta)

2.8 Organització de la resolució

Hem establert que el procés de resolució segueix una estructura en pipeline de tres fases seqüencials: Abstracció, Resolució i Refinació. Aquesta organització no és arbitrària, sinó que respon a principis d'enginyeria del coneixement que faciliten el desenvolupament, manteniment i comprensió del sistema.

2.8.1 Avantatges de l'organització en fases

La separació en fases proporciona diversos beneficis tècnics i metodològics.

Primer, cada fase té una responsabilitat clarament delimitada, seguint el principi de separació de responsabilitats. L'Abstracció infereix necessitats, la Resolució avalia ofertes, i la Refinació classifica resultats. Aquesta separació fa que cada fase sigui més senzilla de desenvolupar i depurar.

Segon, les fases són seqüencials i unidireccionals, evitant dependències circulars complexes. L'Abstracció genera requisits que usa la Resolució, i la Resolució generavaluacions que usa la Refinació, però no hi ha retroalimentació inversa. Això simplifica el flux de control i el raonament sobre el comportament del sistema.

Tercer, cada fase pot desenvolupar-se i provar-se de manera independent. Podem validar que l'Abstracció infereix correctament les necessitats sense haver implementat encara la Resolució. Podem provar la Resolució amb requisits manuals sense dependre de l'Abstracció. Aquest desenvolupament incremental redueix la complexitat.

Quart, la modularitat facilita el manteniment i l'extensió. Si volem afegir una nova regla d'inferència, només cal modificar l'Abstracció. Si volem canviar els criteris de classificació, només cal modificar la Refinació. Les altres fases no es veuen afectades.

2.8.2 Flux de dades entre fases

El flux de dades és clarament definit i traceable:

Fase 1 (Abstracció):

- Entrada: Sol · licitant amb atributs bàsics
- Sortida: Requisits inferits (serveis obligatoris, serveis preferents, restriccions)

Fase 2 (Resolució):

- Entrada: Sol · licitant + Requisits inferits + Oferta + Serveis
- Sortida: O descart amb motiu, o (CriterisNoComplerts, PuntsPositius)

Fase 3 (Refinació):

- Entrada: (CriterisNoComplerts, PuntsPositius)
- Sortida: (GrauRecomanació, Puntuació)

Aquesta estructura en pipeline assegura que la informació flueix de manera predictable i que cada fase afegeix valor de manera incremental.

2.9 Conclusions de la fase de conceptualització

La fase de conceptualització ha permès transformar el problema inicial (recomanar habitatges) en una estructura conceptual clara i implementable. Hem identificat els conceptes principals del domini (Habitatge, Sol · licitant, Servei, Oferta, Localització), les seves característiques rellevants, i les relacions que els connecten.

Hem extret coneixement expert de diverses fonts, incloent models de llenguatge utilitzats com a experts virtuals, que hem validat i integrat de manera crítica. Aquest coneixement s'ha estructurat en forma de regles heurístiques organitzades per categories: regles basades en perfil demogràfic, regles de sentit comú, regles de proximitat, i regles de pressupost.

Hem descompost el problema principal en tres subproblemes seqüencials (Abstracció, Resolució, Refinació), cadascun amb entrada, sortida i mètode clarament definits. Aquesta descomposició facilita tant la implementació com el raonament sobre el comportament del sistema.

Hem validat la conceptualització mitjançant exemples de resolució manual, demostrant que el coneixement estructurat permet arribar a conclusions raonables i justificades per a casos concrets.

Amb aquesta base conceptual sòlida, estem preparats per avançar a la fase de formalització, on crearem una ontologia explícita que capturi tot aquest coneixement de manera precisa i computacionalment tractable.

3 Formalització

En aquesta secció es presenta el procés de formalització del problema mitjançant la construcció d'una ontologia que representa tot el coneixement necessari per al sistema de recomanació d'habitacions. A més, s'explica detalladament la metodologia de resolució de problemes escollida i com s'estructura el raonament del sistema.

3.1 Construcció de l'ontologia

La construcció de l'ontologia és una de les fases més crítiques del desenvolupament d'un sistema basat en el coneixement. Una ontologia ben dissenyada no només facilita la implementació de les regles, sinó que també garanteix que el sistema pugui cobrir tots els casos rellevants del domini.

3.1.1 Procés de construcció

El procés de construcció de l'ontologia ha seguit una metodologia incremental basada en les millors pràctiques d'enginyeria del coneixement. Els passos seguits han estat:

1. **Identificació de conceptes principals:** A partir de l'anàlisi del domini realitzada en la fase de conceptualització, s'han identificat els conceptes clau que intervenen en el problema: habitatges, sol·licitants, ofertes, serveis i localitzacions.
2. **Establiment de jerarquies:** Per cada concepte principal s'ha analitzat si calia crear subclasses especialitzades. Per exemple, s'ha determinat que no tots els habitatges són iguals i que les necessitats varien significativament segons el tipus de sol·licitant.
3. **Definició d'atributs:** Per cada classe s'han identificat les propietats rellevants que permeten caracteritzar adequadament les instàncies i que són necessàries per al raonament del sistema.
4. **Definició de relacions:** S'han establert les connexions entre conceptes mitjançant object properties, especificant domini, rang i cardinalitat quan és rellevant.
5. **Validació amb casos d'ús:** L'ontologia s'ha validat comprovant que permet representar tots els escenaris identificats durant la conceptualització.
6. **Iteració i refinament:** L'ontologia s'ha refinat múltiples vegades durant el desenvolupament, afegint conceptes o atributs quan les proves revelaven mancances.

Aquest procés iteratiu ha estat fonamental per arribar a una ontologia completa i coherent que cobreix tots els aspectes rellevants del problema.

3.1.2 Jerarquia de classes

L'ontologia s'estructura en diverses jerarquies de classes que representen els diferents aspectes del domini. A continuació es presenten les jerarquies principals i es justifica per què s'han dissenyat d'aquesta manera.

Jerarquia d'Habitatge La jerarquia d'habitatges distingeix entre diferents tipus d'habitatge, ja que cadascun té característiques específiques que afecten la seva adequació per a diferents perfils:

```
Habitatge (classe abstracta)
|-- Pis
|-- Àtic
|-- Dúplex
|-- Estudi
`-- HabitatgeUnifamiliar
```

Justificació de la jerarquia: Aquesta distinció és necessària perquè cada tipus d'habitatge té característiques intrínsecament diferents. Els àtics solen tenir terrasses més grans i millors vistes però poden tenir problemes d'aïllament tèrmic. Els estudis són adequats per a persones soles o parelles sense fills però no per a famílies. Els habitatges unifamiliars ofereixen més privacitat i espai però requereixen més manteniment. Aquestes diferències són rellevants per al raonament del sistema i justifiquen la necessitat de subclasses específiques.

Jerarquia de Sol · licitant La jerarquia de sol · licitants és potser la més important, ja que el perfil del sol · licitant determina completament quines són les seves necessitats:

```
Sol·licitant (classe abstracta)
|-- Individu
|-- Parella
|   |-- ParellaSenseFills
|   |-- ParellaAmbFills
|   `-- ParellaFutursFills
|-- Família
|   |-- FamiliaBiparental
|   `-- FamiliaMonoparental
|-- GrupEstudiants
`-- PersonaGran
```

Justificació de la jerarquia: Les necessitats d'habitatge varien radicalment segons la tipologia demogràfica del sol · licitant. Una família amb fills necessita escoles properes i zones verdes, mentre que un grup d'estudiants prioritza el transport públic i el preu baix. Una persona gran necessita accessibilitat i serveis de salut propers. Aquesta jerarquia permet al sistema aplicar regles específiques per a cada perfil sense haver de codificar totes les combinacions possibles de característiques. A més, la distinció entre parelles amb fills, sense fills i amb plans de tenir fills permet anticipar necessitats futures.

Jerarquia de Servei Els serveis del voltant de l'habitatge són un factor clau en la decisió. La jerarquia de serveis permet classificar-los segons la seva funció:

Servei (classe abstracta)

```
|-- TransportPublic
|   |-- EstacioMetro
|   |-- ParadaBus
|   `-- EstacioTren
|-- ServeiEducatiu
|   |-- LlarInfants
|   |-- Escola
|   |-- Institut
|   `-- Universitat
|-- ServeiSalut
|   |-- Hospital
|   |-- CentreSalut
|   `-- Farmàcia
|-- ServeiComercial
|   |-- Supermercat
|   |-- Hipermercat
|   |-- CentreComercial
|   `-- Mercat
|-- ZonaVerda
|   |-- Parc
|   |-- Jardí
|   `-- ZonaEsportiva
|-- ServeiOci
|   |-- Gimnàs
|   |-- Biblioteca
|   |-- CentreCultural
|   `-- ZonaNocturna
`-- ServeiMolest
    |-- Discoteca
    |-- Estadi
    |-- Aeroport
    |-- ZonaIndustrial
    `-- Autopista
```

Justificació de la jerarquia: Aquesta categorització permet que les regles del sistema facin referència a categories completes de serveis en lloc d'haver de llistar serveis individuals. Per exemple, una regla pot expressar que l'les famílies amb fills necessiten serveis educatius propers "sense haver d'enumerar escoles, instituts, llars d'infants, etc. A més, la distinció entre serveis desitjables i serveis molestos (ServeiMolest) permet que el sistema pugui descartar

ofertes que estan massa a prop d'elements que poden resultar problemàtics com discoteques o estadis.

3.1.3 Atributs de les classes

Els atributs definits per a cada classe són els que permeten caracteritzar les instàncies i fer raonaments sobre elles. A continuació es detallen els atributs més rellevants de les classes principals.

Atributs de la classe Habitatge La classe Habitatge té un conjunt extens d'atributs que descriuen tant característiques físiques com qualitatives de l'habitatge:

- **superficieHabitable** (Float): La superfície en metres quadrats és fonamental per determinar si un habitatge és adequat per al nombre de persones que hi viuran. S'utilitza per descartar habitatges massa petits.
- **numeroDormitoris**, **numeroDormitorisDobles**, **numeroDormitorisSimples** (Integer): El nombre i tipus de dormitoris determina la capacitat real de l'habitatge. No és el mateix tenir 3 dormitoris simples que 2 dobles i 1 simple per a una família de 4 persones.
- **numeroBanys** (Integer): El nombre de banys és rellevant especialment per a famílies grans o habitatges compartits. Un sol bany pot ser un problema per a més de 3 persones.
- **plantaPis** (Integer): La planta de l'habitatge és rellevant per a l'accessibilitat. Plantes baixes són ideals per a persones grans, mentre que plantes altes amb vistes són valorades positivament.
- **teAscensor** (Boolean): La presència d'ascensor és crítica per a persones amb necessitats d'accessibilitat. Un habitatge sense ascensor en una planta alta pot ser un motiu de descart automàtic per a persones grans.
- **permetsMascotes** (Boolean): Aquest és un criteri de descart obligatori. Si un sol·licitant té mascotes i l'habitatge no les permet, l'oferta es descarta immediatament.
- **moblat** (Boolean): Important especialment per a estudiants o persones que busquen estades temporals. Un habitatge moblat estalvia la inversió inicial en mobiliari.
- **teTerrassaOBalco** (Boolean) i **superficieTerrassa** (Float): La presència d'espai exterior és valorada positivament, especialment per a famílies amb fills o persones que treballen des de casa.
- **orientacioSolar** (String: "Mati", "Tarda", "TotElDia", "Nord"): La quantitat de llum natural afecta significativament la qualitat de vida. Habitatges assolellats tot el dia són valorats positivament.

- **teVistes** (Boolean) i **tipusVistes** (String): Les bones vistes són un plus que pot compensar altres aspectes menys favorables.
- **teCalefaccio, teAireCondicionat** (Boolean): La climatització afecta el confort i els costos d'ús de l'habitatge.
- **tePlacaAparcament** (Boolean): Fonamental per a sol·licitants que tenen vehicle. Pot ser un criteri de descart si el sol·licitant té cotxe i la zona no té aparcament públic.
- **consumEnergetic** (String: "A", "B", "C", "D", "E"): L'eficiència energètica determina els costos d'ús i és cada vegada més valorada.
- **nivellSoroll** (String: "Baix", "Mitja", "Alt"): El soroll és un factor de qualitat de vida important. Habitatges amb soroll alt poden ser descartats o penalitzats.
- **estatConservacio** (String: "Nou", "BonEstat", "Reformar"): L'estat de conservació afecta els costos inicials i de manteniment.

Cadascun d'aquests atributs s'ha inclòs perquè és rellevant per a alguna de les regles del sistema. No hi ha atributs superflus, però tampoc hi manca informació necessària per fer un raonament adequat.

Atributs de la classe Sol·licitant Els atributs del sol·licitant descriuen tant les seves característiques demogràfiques com les seves preferències i restriccions:

- **nom** (String): Identificador per a la presentació de resultats.
- **edat** (Integer): L'edat determina la categoria demogràfica i influeix en necessitats específiques (persones joves vs. persones grans).
- **numeroPersones** (Integer): El nombre de persones determina l'espai mínim necessari i el nombre de dormitoris adequat.
- **pressupostMaxim, pressupostMinim** (Float): El pressupost és la restricció més important. El màxim determina què es pot pagar, el mínim permet detectar ofertes sospitosament barates.
- **margeEstricto** (Boolean): Indica si el pressupost màxim és inamovible o hi ha una mica de flexibilitat. Si és estricto, qualsevol oferta per sobre es descarta. Si no, ofertes lleugerament per sobre poden considerar-se com a parcialment adequades.
- **numeroFills** (Integer) i **edatsFills** (List<Integer>): La presència i edat dels fills és fonamental per inferir necessitats (escoles, zones verdes) i determinar l'espai necessari.
- **teAvis** (Boolean): La presència de persones grans a la llar afegeix necessitats d'accessibilitat i proximitat a serveis de salut.

- **teVehicle** (Boolean): Determina si cal valorar la presència de parking.
- **teMascotes** (Boolean), **numeroMascotes** (Integer), **tipusMascota** (String): Les mascotes són una restricció obligatòria que pot descartar ofertes.
- **prefereixTransportPublic** (Boolean): Indica si el sol·licitant vol tenir transport públic molt a prop.
- **necessitaAccessibilitat** (Boolean): Restricció obligatòria per a persones amb mobilitat reduïda. Descarta habitatges sense ascensor en plantes altes.
- **treballaACiutat, estudiaACiutat** (Boolean): Si el sol·licitant treballa o estudia a la ciutat, la proximitat a transport públic és més important.

Aquests atributs permeten al sistema inferir automàticament necessitats i aplicar les regles adequades per a cada perfil.

Atributs de la classe Oferta L'oferta és la classe que connecta un habitatge amb les condicions de lloguer:

- **preuMensual** (Float): El preu de l'oferta, que es compara amb el pressupost del sol·licitant.
- **disponible** (Boolean): Permet filtrar ofertes que ja no estan disponibles.
- **dataPublicacio** (String): Informació addicional que pot ser rellevant per a l'usuari.
- **grauRecomanacio** (String): Atribut que el sistema emplenarà amb la classificació final (MoltRecomanable, Adequat, Parcialment).
- **motiusRecomanacio** (List<String>): Llista de justificacions de per què es recomana o no l'oferta.

Atributs de la classe Localitzacio

- **coordenadaX, coordenadaY** (Float): Coordenades en un sistema cartesià simplificat. En un sistema real es farien servir coordenades GPS (latitud/longitud).
- **districte** (String): Districte de la ciutat per a informació contextual.
- **adreca** (String): Adreça completa per a presentació.

3.1.4 Relacions entre classes

Les relacions (object properties) connecten les diferents classes i permeten navegar per l'ontologia durant el raonament.

Relació teLocalitzacio

- **Domini:** Habitatge \cup Servei
- **Rang:** Localitzacio
- **Cardinalitat:** Funcional (1:1)
- **Justificació:** Tot habitatge i tot servei té exactament una ubicació. Aquesta relació és fonamental per poder calcular distàncies i determinar la proximitat entre habitatges i serveis.

Relació aPropDe

- **Domini:** Localitzacio
- **Rang:** Servei
- **Subpropietats:**
 - moltAPropDe: distància $< 500\text{m}$
 - aDistanciaMitjana: $500\text{m} \leq \text{distància} < 1000\text{m}$
 - llunyDe: distància $\geq 1000\text{m}$
- **Justificació:** Aquesta jerarquia de relacions permet expressar proximitat de manera qualitativa. Les regles poden fer referència a "serveis molt a prop" sense haver de comparar distàncies numèriques en cada regla, simplificant el codi i millorant la llegibilitat.

Relació teHabitatge

- **Domini:** Oferta
- **Rang:** Habitatge
- **Cardinalitat:** Funcional (1:1)
- **Justificació:** Cada oferta fa referència a un habitatge concret. Aquesta relació permet accedir a les característiques de l'habitatge des de l'oferta.

Relacions de preferència del sol · licitant

- **requereixServei** (Sol · licitant \rightarrow Servei): El sol · licitant necessita obligatòriament aquest servei a prop. Si no està disponible, pot ser motiu de descart.
- **prefereixServei** (Sol · licitant \rightarrow Servei): El sol · licitant valora positivament aquest servei però no és obligatori.

- **evitaServei** (Sol · licitant → ServeiMolest): El sol · licitant prefereix no tenir aquest tipus de servei molt a prop. Si està molt a prop, pot ser motiu de descart.

Aquestes relacions permeten que els usuaris expressin preferències específiques més enllà de les que s'infereixen automàticament del seu perfil.

3.2 Metodologia de resolució de problemes

La metodologia de resolució escollida determina com s'estructura el raonament del sistema. Aquesta decisió és crítica perquè afecta tant l'eficiència com la mantenibilitat del codi.

3.2.1 Paradigma escollit: Forward Chaining amb fases

El sistema utilitzà **encadenament cap endavant (forward chaining)** organitzat en tres fases seqüencials ben diferenciades. Cada fase té un objectiu específic i genera fets que són consumits per la fase següent.

Les tres fases són:

1. **Fase d'Abstracció:** Infereix requisits implícits a partir del perfil del sol · licitant i calcula proximitats entre habitatges i serveis.
2. **Fase de Resolució:** Avalua cada combinació sol · licitant-oferta, descartant ofertes inadequades i detectant punts positius i negatius.
3. **Fase de Refinació:** Classifica les ofertes no descartades en categories (Molt Recomanable, Adequat, Parcialment Adequat) segons els punts positius i negatius detectats.

3.2.2 Justificació del Forward Chaining

La decisió d'utilitzar forward chaining en lloc de backward chaining es basa en diversos factors:

- **Dades disponibles des del principi:** En el nostre problema, tenim totes les dades (sol · licitants i ofertes) des de l'inici. No estem buscant demostrar un objectiu concret sinó avaluar exhaustivament totes les combinacions possibles.
- **Avaluació exhaustiva requerida:** Volem avaluar totes les ofertes per a cada sol · licitant i presentar-li un llistat complet de recomanacions ordenades. El backward chaining seria adequat per a consultes específiques com "És adequada l'oferta X per a l'usuari Y?", però no per a la nostra necessitat d'avaluació global.
- **Explicabilitat natural:** Amb forward chaining, el traç d'execució mostra clarament com s'han inferit requisits, per què s'han descartat ofertes i quins criteris s'han avaluat. Això facilita la generació d'explicacions comprensibles per a l'usuari.
- **Eficiència de CLIPS:** L'algoritme RETE que implementa CLIPS està altament optimitzat per a forward chaining, fent que l'avaluació de centenars de combinacions sigui molt eficient.

3.2.3 Justificació de l'organització en fases

L'organització del raonament en tres fases separades ofereix múltiples avantatges respecte a un sistema monolític:

- **Separació de responsabilitats:** Cada fase té un objectiu clar i específic. Les regles d'inferència de requisits són conceptualment diferents de les regles de descart, que al seu torn són diferents de les regles de classificació. Aquesta separació fa que cada regla sigui més senzilla i fàcil d'entendre.
- **Control de flux predictable:** Les fases s'executen sempre en el mateix ordre (abstracció → resolució → refinació), evitant conflictes entre regles de diferents fases. Això fa que el comportament del sistema sigui més predictable i fàcil de depurar.
- **Mantenibilitat:** Si cal modificar una regla o afegir-ne una de nova, és fàcil identificar a quina fase pertany i on s'ha de col·locar. Això redueix el risc d'introduir errors en afegir funcionalitat nova.
- **Facilitat de depuració:** Si hi ha un error en els resultats, és fàcil identificar en quina fase s'ha produït. Per exemple, si una oferta s'hauria d'haver descartat però no s'ha fet, sabem que el problema està en la fase de resolució.
- **Reusabilitat:** Els resultats d'una fase (per exemple, els requisits inferits) poden ser reutilitzats per múltiples regles de fases posteriors sense haver de recalcular-los.

El control entre fases s'implementa mitjançant fets de control que indiquen quan una fase ha acabat:

```
1 (deftemplate fase-completada
2   (slot nom (type SYMBOL)))
```

Les regles de cada fase comproven que la fase anterior s'ha completat abans d'executar-se:

```
1 (defrule resolucion-exemple
2   (declare (salience 40))
3   (fase-completada (nom abstraccion)) ; Comprova fase anterior
4   ; ... resta de condicions ...
5   =>
6   ; ... accions ...
7 )
```

3.2.4 Templates auxiliars per al raonament

A més de les classes de l'ontologia (implementades com a COOL objects), el sistema utilitza templates CLIPS per representar fets intermedis del raonament. Aquesta distinció entre objects i templates és intencionada i justificada:

- **Objects (COOL)**: S'utilitzen per representar conceptes del domini (habitacles, sol·licitants, serveis). Aquestes són entitats estables que existeixen independentment del raonament.
- **Templates**: S'utilitzen per representar fets derivats del raonament, que són específics d'una execució concreta i no tenen sentit fora del context del procés de decisió.

Els templates principals són:

proximitat Representa la distància entre un habitatge i un servei:

```

1 (deftemplate proximitat
2   (slot habitatge (type INSTANCE))
3   (slot servei (type INSTANCE))
4   (slot categoria (type SYMBOL))
5   (slot distancia (type SYMBOL)) ; MoltAProp, DistanciaMitjana
       , Lluny
6   (slot metres (type FLOAT)))

```

Justificació: Calcular distàncies és costós computacionalment. En lloc de recalcular la distància cada vegada que una regla la necessita, es calcula una sola vegada al principi i s'emmagatzema com a fet. Això millora enormement l'eficiència.

requisit-inferit Representa necessitats detectades automàticament pel sistema:

```

1 (deftemplate requisit-inferit
2   (slot solicitant (type INSTANCE))
3   (slot categoria-servei (type SYMBOL))
4   (slot obligatori (type SYMBOL)) ; si/no
5   (slot motiu (type STRING)))

```

Justificació: Els requisits inferits són conclusions del raonament, no dades inicials. Representar-los com a fets separats permet que múltiples regles puguin consultar-los i que s'incloguin en les explicacions al usuari.

oferta-descartada Indica que una oferta s'ha eliminat de la consideració per a un sol·licitant:

```

1 (deftemplate oferta-descartada
2   (slot solicitant (type INSTANCE))
3   (slot oferta (type INSTANCE))
4   (slot motiu (type STRING)))

```

Justificació: En comptes d'eliminar realment les ofertes de la memòria, es marquen com a descartades amb una justificació. Això permet presentar a l'usuari per què certes ofertes no són adequades, millorant la transparència del sistema.

criteri-no-cumplit Representa aspectes negatius d'ofertes que no es descarten completamente:

```

1 (deftemplate criteri-no-cumplit
2   (slot solicitant (type INSTANCE))
3   (slot oferta (type INSTANCE))
4   (slot criteri (type STRING))
5   (slot gravetat (type SYMBOL))) ; Lleu, Moderat, Greu

```

Justificació: No totes les ofertes són perfectes o totalment inadequades. Moltes tenen aspectes negatius que no justifiquen un descart total però que l'usuari ha de conèixer. Aquest template permet acumular aquests aspectes i utilitzar-los tant per a la classificació com per a l'explicació.

punt-positiu Representa aspectes destacables d'una oferta:

```

1 (deftemplate punt-positiu
2   (slot solicitant (type INSTANCE))
3   (slot oferta (type INSTANCE))
4   (slot descripcio (type STRING)))

```

Justificació: Simètricament als criteris negatius, els punts positius s'acumulen per determinar si una oferta és "Molt Recomanable" per explicar a l'usuari què fa que l'oferta destaquï.

recomanacio Representa la classificació final d'una oferta:

```

1 (deftemplate recomanacio
2   (slot solicitant (type INSTANCE))
3   (slot oferta (type INSTANCE))
4   (slot grau (type SYMBOL)) ; MoltRecomanable, Adequat,
5     Parcialment
6   (slot puntuacio (type INTEGER)))

```

Justificació: La recomanació final és el resultat del raonament complet. Tenir-la com a fet separat permet presentar els resultats de manera ordenada i permet futures extensions com ordenar ofertes per puntuació.

3.3 Justificació de decisions de disseny

Durant el procés de formalització s'han pres diverses decisions de disseny que cal justificar adequadament.

3.3.1 Per què no Backward Chaining?

S'ha considerat utilitzar backward chaining però s'ha descartat pels següents motius:

- **Backward chaining és goal-driven:** És adequat quan es vol demostrar un objectiu específic, per exemple "És adequat l'habitació X per a l'usuari Y?". En el nostre cas, no tenim un objectiu específic sinó que volem avaluar exhaustivament totes les ofertes per a cada usuari.
- **Avaluació selectiva vs. exhaustiva:** Amb backward chaining, el sistema començaria des d'un objectiu i només buscaria la informació necessària per confirmar-lo o rebutjar-lo. Això és eficient per a consultes puntuals però inadequat quan volem avaluar totes les ofertes disponibles.
- **Generació d'explicacions:** Forward chaining genera naturalment un traç d'execució que mostra com s'han inferit els requisits i per què s'han pres les decisions. Aquest traç és la base de les explicacions que es presenten a l'usuari.

3.3.2 Per què Templates en lloc de només Objects?

CLIPS ofereix dos mecanismes per representar informació: templates (fets simples) i objects (programació orientada a objectes). S'ha optat per utilitzar ambdós de manera complementària:

- **Objects per conceptes estables del domini:** Les classes de l'ontologia (Habitatge, Sol·licitant, Servei, etc.) s'implementen com a objects perquè representen entitats del món real amb estructura complexa i jerarquies d'herència.
- **Templates per fets derivats del raonament:** Els requisits inferits, proximitats, criteris no complerts, etc. s'implementen com a templates perquè són fets temporals específics d'una execució concreta.
- **Eficiència del pattern matching:** CLIPS està altament optimitzat per fer pattern matching sobre templates. Per a fets que es consulten freqüentment en condicions de regles, els templates són més eficients que els objects.
- **Simplicitat de les regles:** Els templates permeten escriure condicions més concises. Per exemple, comprovar l'existència d'un requisit inferit és més simple amb templates que amb objects.

3.3.3 Per què tres fases en lloc de dues o més?

S'han considerat diferents alternatives d'organització:

- **Una sola fase:** Massa complex. Les regles d'inferència, descart i classificació es barrejarien, fent el sistema difícil d'entendre i mantenir. El risc de conflictes entre regles seria alt.
- **Dues fases (Avaluació + Classificació):** No separa adequadament la inferència de coneixement de l'avaluació d'ofertes. La inferència de requisits hauria de fer-se juntament amb el descart d'ofertes, cosa conceptualment confusa.

- **Tres fases** (escollit): Ofereix un equilibri òptim. Cada fase té una responsabilitat clara i ben definida. El flux és natural: primer s'entén què necessita el sol · licitant (abstracció), després s'avaluen les ofertes segons aquestes necessitats (resolució), i finalment es classifiquen els resultats (refinació).
- **Quatre o més fases:** Fragmentaria massa el procés sense guanys clars. Per exemple, separar la inferència de requisits del càlcul de proximitats afegiria complexitat innecessària.

3.3.4 Per què pre-calcular les proximitats?

Una alternativa seria calcular la distància entre habitatge i servei cada vegada que una regla ho necessita. S'ha optat per pre-calcular totes les distàncies per diversos motius:

- **Eficiència:** El càlcul de distàncies (fins i tot amb distància de Manhattan simplificada) és computacionalment costós si s'ha de fer repetidament. Pre-calcular totes les distàncies una sola vegada i emmagatzemar-les com a fets permet que les regles posteriors les consultin de manera instantània.
- **Evita duplicació de codi:** Sense pre-càlcul, cada regla que necessita saber si un servei està a prop hauria d'incloure el càlcul de distància, duplicant codi i incrementant el risc d'errors.
- **Classificació qualitativa:** Pre-calcular permet classificar les distàncies en categories qualitatives (Molt a Prop, Distància Mitjana, Lluny) una sola vegada. Les regles poden després fer referència a aquestes categories de manera natural sense preocupar-se dels valors numèrics exactes.
- **Facilita extensions futures:** Si en el futur es volgués utilitzar distàncies més complexes (per exemple, temps de desplaçament en transport públic en lloc de distància euclidiana), només caldria modificar la regla de càlcul de proximitats, no totes les regles que utilitzen aquesta informació.

3.4 Cobertura de l'ontologia

L'ontologia dissenyada cobreix adequadament el domini definit en el problema. A continuació es detalla l'abast de la cobertura:

- **Tipus d'habitatge:** 5 categories principals (Pis, Àtic, Dúplex, Estudi, Habitatge Unifamiliar) que cobreixen la gran majoria de casos reals d'habitacions de lloguer urbà.
- **Perfils de sol · licitant:** 7 tipologies diferents (Individu, Parella amb diferents situacions, Família, Grup d'Estudiants, Persona Gran) que representen els principals perfils demogràfics que busquen habitatge de lloguer.

- **Serveis:** 6 categories principals amb més de 20 subclasses específiques. Això permet representar tots els serveis rellevants del voltant d'un habitatge que poden influir en la decisió.
- **Atributs:** Més de 40 propietats diferents entre totes les classes, cobriment tots els aspectes rellevants identificats durant la fase de conceptualització.
- **Regles de raonament:** Més de 30 regles que cobreixen la inferència de requisits, el descart d'ofertes inadequades, la detecció de punts positius i negatius, i la classificació final.

Casos conscientment no coberts:

S'ha decidit deixar fora de l'abast del sistema alguns casos que, tot i ser reals, són menys freqüents o requeririen una complexitat desproporcionada:

- **Habitatges comercials:** Oficines, locals comercials, naus industrials. El sistema es centra en habitatge residencial.
- **Lloguers temporals o vacacionals:** El sistema assumeix lloguers estables de llarga durada (mínim un any).
- **Co-housing o lloguers compartits parcials:** El sistema assumeix que es lloga l'habitacle complet, no habitacions individuals en habitatges compartits amb desconeguts.
- **Necessitats mèdiques molt específiques:** Més enllà de l'accessibilitat general i la proximitat a serveis de salut, no es consideren necessitats mèdiques específiques com diàlisi, rehabilitació, etc.

Aquestes exclusions estan justificades perquè els casos coberts representen la gran majoria de situacions reals i permeten mantenir el sistema en una complexitat raonable.

4 Implementació

En aquest capítol es descriu com s'ha implementat el sistema basat en el coneixement utilitzant CLIPS com a llenguatge de programació. S'explica l'arquitectura del sistema, com s'ha traduït l'ontologia a estructures CLIPS, com s'han implementat les regles i quina metodologia de desenvolupament s'ha seguit.

4.1 Arquitectura del sistema

El sistema s'ha estructurat de manera modular per facilitar el desenvolupament, el manteniment i la comprensió del codi. Aquesta modularitat s'ha aconseguit mitjançant la separació en múltiples fitxers i l'organització clara de les responsabilitats.

4.1.1 Estructura de fitxers

El sistema està organitzat en els següents fitxers:

```
src/
|-- ontologiaSBC.clp          # Classes de l'ontologia
|-- ontologiaSBC.owl           # Ontologia original en OWL/XML
|-- ontologiaSBC.ttl           # Ontologia en format Turtle
|-- regles.clp                 # Regles del sistema expert
|-- instances.clp               # Instàncies bàsiques de prova
|-- instances_extended.clp     # Instàncies addicionals (100+)
|-- main.clp                   # Interfície interactiva
`-- run.clp                    # Script de càrrega i execució
```

Cada fitxer té una responsabilitat específica:

- **ontologiaSBC.clp**: Conté les definicions de totes les classes de l'ontologia traduïdes a CLIPS amb COOL (CLIPS Object-Oriented Language). Aquest fitxer es genera automàticament a partir de l'ontologia OWL utilitzant l'eina owl2clips.
- **ontologiaSBC.owl i ontologiaSBC.ttl**: Són versions de l'ontologia en formats estàndard. OWL/XML és el format natiu de Protégé, mentre que Turtle és més llegible per a humans i útil per a documentació.
- **regles.clp**: Conté totes les regles del sistema expert organitzades per fases. Aquest és el fitxer més gran i complex, amb més de 30 regles que implementen la lògica de raonament.
- **instances.clp**: Conté un conjunt bàsic d'instàncies de prova (uns 10-15 casos) que cobreixen els escenaris principals. S'utilitza per proves ràpides durant el desenvolupament.

- **main.clp:** Implementa una interfície interactiva que permet a l'usuari crear perfils de sol·licitant dinàmicament i obtenir recomanacions personalitzades.
- **run.clp:** Script que carrega tots els fitxers necessaris en l'ordre correcte i inicialitza el sistema.

4.1.2 Flux d'execució

L'execució del sistema segueix sempre la mateixa seqüència:

1. **Càrrega:** L'usuari executa (`load "run.clp"`) que carrega automàticament tots els fitxers necessaris en l'ordre correcte: primer l'ontologia, després les regles i finalment les instàncies.
2. **Reset:** La comanda (`reset`) inicialitza la memòria de treball de CLIPS, creant totes les instàncies definides i posant el sistema en l'estat inicial.
3. **Execució:** Hi ha dues opcions per executar el sistema:
 - **Mode automàtic:** La comanda (`run`) processa automàticament totes les combinacions de sol·licitants i ofertes definides en els fitxers d'instàncies, generant recomanacions per a cadascuna.
 - **Mode interactiu:** La comanda (`main`) lanza una interfície que fa preguntes a l'usuari per crear un perfil de sol·licitant personalitzat i després genera recomanacions només per a aquest perfil.
4. **Processament:** El sistema executa seqüencialment les tres fases de raonament (Abstractació, Resolució, Refinació) per a cada combinació sol·licitant-oferta.
5. **Presentació de resultats:** Un cop completat el raonament, regles de presentació mostren les recomanacions amb les seves justificacions.

Aquest flux garanteix que el sistema sempre opera de manera consistent i predictable.

4.2 Traducció de l'ontologia a CLIPS

L'ontologia dissenyada en Protégé s'ha traduït a CLIPS utilitzant l'extensió COOL (CLIPS Object-Oriented Language), que proporciona capacitats de programació orientada a objectes.

4.2.1 Procés de generació

La conversió de l'ontologia OWL a CLIPS s'ha realitzat mitjançant l'eina owl2clips, seguint aquest procés:

1. Disseny de l'ontologia en Protégé 5.6, definint classes, atributs, jerarquies i relacions de manera visual.

2. Exportació a format Turtle (TTL), que és més adequat per a la conversió automàtica que el format OWL/XML.
3. Execució de l'eina owl2clips per generar automàticament les definicions CLIPS:

```
java -jar owl2clips.jar ontologiaSBC.ttl > ontologiaSBC.clp
```

4. Revisió i ajust manual del fitxer generat per assegurar que les definicions són correctes i afegir documentació.

Aquest procés automatitzat garanteix que l'ontologia en CLIPS és fidel al disseny original i estalvia la feina d'haver de traduir manualment totes les classes i atributs.

4.2.2 Estructura de les classes generades

Les classes de l'ontologia es tradueixen a `defclass` de COOL. A continuació es mostra un exemple d'una classe generada:

```

1  (defclass Habitatge
2    (is-a USER)
3    (role concrete)
4    (pattern-match reactive)
5    (slot superficieHabitable
6      (type FLOAT)
7      (create-accessor read-write))
8    (slot numeroDormitoris
9      (type INTEGER)
10     (create-accessor read-write))
11    (slot numeroDormitorisDobles
12      (type INTEGER)
13      (create-accessor read-write))
14    (slot numeroDormitorisSimples
15      (type INTEGER)
16      (create-accessor read-write))
17    (slot permetMascotes
18      (type SYMBOL)
19      (allowed-symbols si no)
20      (create-accessor read-write))
21    (slot teLocalitzacio
22      (type INSTANCE)
23      (create-accessor read-write))
24    ; ... més slots ...
25 )
```

Listing 1: Exemple de classe Habitatge en CLIPS

Els elements clau d'aquesta definició són:

- (**is-a USER**): Indica que és una classe concreta que hereta de la classe arrel USER de COOL.
- (**role concrete**): Especifica que es poden crear instàncies d'aquesta classe (no és abstracta).
- (**pattern-match reactive**): Activa el pattern matching reactiu, essencial per al forward chaining.
- **Slots**: Cada atribut de l'ontologia es tradueix a un slot amb el seu tipus corresponent i accessors.

Les subclasses mantenen la jerarquia mitjançant **is-a**:

```
1 (defclass Pis
2   (is-a Habitatge)
3   (role concrete)
4   (pattern-match reactive))
```

Listing 2: Exemple de classe

Una classe hereta automàticament tots els slots de la seva superclasse, per la qual cosa només cal especificar slots addicionals si n'hi ha.

4.3 Metodologia de desenvolupament

El desenvolupament del sistema ha seguit una metodologia de prototipatge ràpid i disseny incremental, tal com es recomana per a sistemes basats en el coneixement. Aquesta aproximació permet validar decisions de disseny aviat i ajustar el sistema de manera iterativa.

4.3.1 Iteració 1: Prototip mínim funcional

Setmana 3 - Objectiu: Crear un sistema mínim que demostri la viabilitat de l'aproximació.
Característiques implementades:

- Ontologia simplificada amb només 3 classes: Habitatge (sense subclasses), Sol·licitant (sense subclasses) i Oferta.
- 5 regles bàsiques que implementen els criteris més evidents:
 - Descart per preu excessiu
 - Descart si no permet mascotes
 - Descart per superfície insuficient
 - Classificació com a "Recomanat" si passa tots els filtres

- Classificació com a "No Recomanat" si es descarta
- 3 instàncies de prova: un sol · llicitant i dues ofertes (una adequada i una inadequada).
- Sortida molt simple: només "Recomanat" o "No Recomanat" sense explicacions detallades.

Resultats de la iteració:

El prototip va funcionar correctament i va demostrar que l'aproximació amb CLIPS era viable. No obstant això, es van identificar diverses limitacions:

- La manca de jerarquies de classes feia que les regles fossin massa genèriques i no poguessin capturar diferències importants entre tipus d'habitatges o perfils de sol · llicitants.
- Les recomanacions binàries (recomanat/no recomanat) eren massa simplistes. Calia una classificació més matisada.
- La falta d'explicacions feia el sistema una "caixa negra" poc útil per a l'usuari.
- No hi havia cap mecanisme per inferir necessitats implícites del perfil del sol · llicitant.

Decisions preses:

A partir d'aquestes observacions es va decidir:

1. Ampliar l'ontologia amb jerarquies completes de classes.
2. Implementar un sistema de tres nivells de recomanació (Molt Recomanable, Adequat, Parcialment Adequat).
3. Afegir mecanismes per capturar justificacions de les decisions.
4. Implementar regles d'inferència que dedueixin necessitats del perfil.

4.3.2 Iteració 2: Ampliació de l'ontologia i millora de regles

Setmana 4 - Objectiu: Implementar una ontologia completa i millorar la qualitat de les recomanacions.

Millores implementades:

- Jerarquia completa de Sol · llicitant amb 7 subclasses que cobreixen els principals perfils demogràfics.
- Jerarquia de Servei amb 6 categories principals i més de 20 subclasses.
- Sistema de proximitat que calcula i classifica distàncies entre habitatges i serveis.
- 10 instàncies de prova que cobreixen diversos escenaris diferents.

- Implementació inicial del sistema de tres fases (Abstracció, Resolució, Refinació) encara que amb poques regles en cada fase.
- Primers mecanismes per capturar punts positius i negatius de les ofertes.

Problemes detectats durant la iteració:

Tot i les millores significatives, es van trobar nous problemes:

- El càlcul de distàncies es feia múltiples vegades dins de diferents regles, afectant l'eficiència.
- Algunes regles eren massa específiques (per exemple, una regla diferent per a cada tipus de servei), fent el sistema difícil de mantenir.
- El control de flux entre fases no era prou robust, i algunes vegades regles de diferents fases es barrejaven.
- Les explicacions encara eren incomplides i no sempre es generaven correctament.

Decisions preses:

Es va decidir:

1. Implementar el pre-càlcul de distàncies mitjançant templates de proximitat.
2. Generalitzar regles utilitzant la jerarquia de classes en lloc de regles específiques per a cada subclasse.
3. Reforçar el control de fases mitjançant salience i fets de control.
4. Revisar i millorar el sistema de generació d'explicacions.

4.3.3 Iteració 3: Refinament i optimització

Setmana 5 - Objectiu: Optimitzar el sistema i completar totes les regles necessàries.

Millores implementades:

- Implementació de templates auxiliars (proximitat, requisit-inferit, oferta-descartada, criteri-no-cumplit, punt-positiu, recomanacio) per estructurar millor el raonament.
- Més de 30 regles organitzades clarament per fases amb salience assignada adequadament:
 - 6 regles d'abstracció per inferir requisits
 - 15 regles de resolució (7 de descart, 3 de criteris negatius, 5 de punts positius)
 - 3 regles de refinació per classificar
 - Regles de presentació

- Sistema robust de control de flux amb fets **fase-completada** i salience ben assignada.
- Explicacions detallades que inclouen:
 - Llista de punts positius amb justificació
 - Llista de criteris no complerts amb gravetat
 - Motius específics de descart per a ofertes eliminades
- 20+ instàncies de prova que cobreixen una àmplia varietat d'escenaris.

Validació de la iteració:

Aquesta iteració va produir un sistema plenament funcional. Les proves van mostrar que:

- Totes les regles s'activaven correctament en l'ordre esperat.
- Les recomanacions eren coherents i ben justificades.
- L'eficiència era adequada (menys de 2 segons per processar 15 sol · licitants x 40 ofertes).
- Les explicacions eren clares i comprensibles.

No obstant això, les proves també van revelar alguns casos extremes que no es gestionaven correctament.

4.3.4 Iteració 4: Proves exhaustives i correcció d'errors

Setmana 6 - Objectiu: Provar el sistema exhaustivament i corregir tots els errors detectats.

Activitats realitzades:

- Creació de més de 100 instàncies de prova que cobreixen:
 - Tots els tipus d'habitatge
 - Tots els perfils de sol · licitant
 - Combinacions de restriccions complexes
 - Cases extremes (pressupost mínim, màximes restriccions, etc.)
- Desenvolupament de la interfície interactiva `main.clp` que permet crear perfils dinàmicament.
- Detecció i correcció de casos extremes problemàtics:
 - Duplicació de requisits inferits (solucionat amb patró NOT)
 - Conflictes de salience entre fases (ajustat el rang de salience)
 - Tractament incorrecte de pressupost amb marge flexible (diferenciades regles per a casos estrictes/flexibles)

- No detecció d'ofertes sospitosament barates (afegit pressupostMinim i regla de descart)
- Optimització de regles per millorar l'eficiència:
 - Ordenació de condicions de més a menys restrictives
 - Eliminació de redundàncies
 - Millora del sistema de presentació de resultats
- Documentació completa del codi amb comentaris explicatius.

Resultat final:

Al final d'aquesta iteració el sistema estava completament funcional, provat exhaustivament i documentat adequadament. Les proves van confirmar que:

- El sistema genera recomanacions coherents en el 100% dels casos provats.
- Les explicacions són sempre completes i comprensibles.
- No hi ha errors detectats en cap dels més de 100 casos de prova.
- L'eficiència és més que adequada per a l'ús previst.

4.4 Modularització del codi

Tot i que CLIPS no té un sistema de mòduls explícits com altres llenguatges de programació, s'ha aconseguit una bona modularitat mitjançant diverses tècniques.

4.4.1 Estratègies de modularització

La modularització s'ha aconseguit mitjançant:

1. **Separació per fitxers:** Cada fitxer té una responsabilitat clarament definida. L'ontologia, les regles i les instàncies estan en fitxers separats.
2. **Separació per fases:** Les regles s'organitzen en tres fases conceptuals (Abstracció, Resolució, Refinació) que s'executen seqüencialment.
3. **Convenció de noms coherent:** Totes les regles segueixen el patró **fase-funcio-descripcio**. Per exemple:
 - **abstraccio-familia-amb-fills**
 - **resolucio-descartar-preu-excessiu**
 - **refinacio-molt-recomanable**

Aquest patró fa que sigui immediat identificar a quina fase pertany cada regla i quina és la seva funció.

4. **Templates especialitzats:** Cada tipus de raonament intermedi té el seu template específic, evitant barrejar conceptes diferents en la mateixa estructura de dades.
5. **Documentació exhaustiva:** Cada regla inclou comentaris que expliquen què fa, per què és necessària i com s'integra amb la resta del sistema.

4.4.2 Organització de les fases

Cadascuna de les tres fases està clarament delimitada i té un conjunt específic de regles:

Fase d'Abstracció Responsabilitat: Inferir requisits implícits i calcular proximitats.

Regles incloses:

- **abstracció-calcular-proximitats:** Calcula i classifica totes les distàncies entre habitatges i serveis.
- **abstracció-familia-amb-fills:** Infereix necessitat d'escoles i zones verdes per a famílies.
- **abstracció-persona-gran:** Infereix necessitat de serveis de salut i comerços propers.
- **abstracció-estudiants:** Infereix necessitat de transport públic.
- **abstracció-prefereix-transport:** Detecta preferència explícita per transport públic.
- **abstracció-parella-futurs-fills:** Infereix preferència per zones educatives i verdes.
- **abstracció-finalitzar:** Marca la fase com a completada.

Entrada: Instàncies de Sol·licitant, Habitatge i Servei.

Sortida: Fets proximitat i requisit-inferit.

Fase de Resolució Responsabilitat: Avaluar ofertes, descartar inadequades i detectar punts forts i dèbils.

Aquesta fase es subdivideix conceptualment en tres sub-mòduls:

Sub-mòdul de Filtratge (salience 40-45):

- **resolucio-descartar-preu-excessiu:** Descarta ofertes que superen el pressupost màxim si aquest és estricte.
- **resolucio-descartar-preu-sospitos:** Descarta ofertes amb preus anormalment baixos.
- **resolucio-descartar-no-mascotes:** Descarta ofertes que no permeten mascotes quan el sol·licitant en té.

- **resolucion-descartar-no-accessible:** Descarta ofertes sense ascensor en plantes altes per a persones amb necessitats d'accessibilitat.
- **resolucion-descartar-servei-evitat:** Descarta ofertes molt properes a serveis que el sol · licitant vol evitar.
- **resolucion-descartar-falta-requisit-inferit:** Descarta ofertes que no tenen serveis obligatoris a prop.
- **resolucion-descartar-superficie-insuficient:** Descarta ofertes massa petites per al nombre de persones.

Sub-mòdul de Criteris Negatius (salience 35):

- **resolucion-criteri-preu-alt:** Detecta preus lleugerament per sobre del pressupost.
- **resolucion-criteri-soroll-alt:** Detecta habitatges amb nivell de soroll alt.
- **resolucion-criteri-sense-ascensor:** Detecta absència d'ascensor quan és preferible.

Sub-mòdul de Punts Positius (salience 30-32):

- **resolucion-punt-bon-preu:** Detecta preus significativament per sota del pressupost.
- **resolucion-punt-terrassa:** Valora la presència de terrassa o balcó.
- **resolucion-punt-assolellat:** Valora orientació solar favorable.
- **resolucion-punt-transport-aprop:** Valora proximitat a transport públic.
- **resolucion-punt-eficiencia-energetica:** Valora eficiència energètica alta.
- **resolucion-punt-parking:** Valora presència de parking quan el sol · licitant té vehicle.
- **resolucion-punt-vistes:** Valora bones vistes.
- **resolucion-punt-cobreix-requisit:** Valora compliment de requisits inferits.
- Diverses regles més per altres aspectes positius.

Fase de Refinació Responsabilitat: Classificar ofertes en categories finals.

Regles incloses:

- **refinacio-molt-recomanable:** Classifica ofertes sense criteris negatius i amb 3 o més punts positius.
- **refinacio-adequat:** Classifica ofertes sense criteris negatius però amb menys de 3 punts positius.
- **refinacio-parcialment:** Classifica ofertes amb 1-2 criteris negatius lleus o moderats.

Mòdul de Presentació Responsabilitat: Mostrar resultats a l'usuari de manera clara i organitzada.

Regles incloses (salience negativa per executar-se al final):

- **presentacio-inici:** Mostra capçalera per a cada sol·licitant.
- **presentacio-recomanacio:** Mostra detalls de cada recomanació.
- **presentacio-punts-positius:** Llista els punts forts de l'oferta.
- **presentacio-criteris-negatius:** Llista els aspectes negatius de l'oferta.
- **presentacio-descartades:** Mostra ofertes descartades amb els seus motius.

4.5 Implementació de regles clau

A continuació es detallen alguns exemples de regles representatives que il·lustren les tècniques d'implementació utilitzades.

4.5.1 Càlcul de distàncies i proximitats

El càlcul de distàncies és una operació fonamental que s'utilitza en moltes parts del raonament. Per eficiència, es fa una sola vegada al principi:

```

1 (deffunction calcular-distancia (?x1 ?y1 ?x2 ?y2)
2   "Calcula la distancia de Manhattan entre dos puntos"
3   (+ (abs (- ?x2 ?x1)) (abs (- ?y2 ?y1))))
4
5 (deffunction classificar-distancia (?metres)
6   "Classifica una distancia en categorías qualitativas"
7   (if (< ?metres 500.0) then MoltAProp
8     else (if (< ?metres 1000.0) then DistanciaMitjana
9       else Lluny)))

```

Listing 3: Funcions de càlcul de distància

La regla de pre-càlcul s'executa primer de tot (salience 100):

```

1 (defrule abstraccion-calcular-proximitats
2   "Calcula totes les proximitats entre habitatges i serveis"
3   (declare (salience 100))
4   ?hab <- (object (is-a Habitatge) (teLocalitzacio ?locH))
5   ?locHab <- (object (is-a Localitzacio) (name ?locH)
6                      (coordenadaX ?x1) (coordenadaY ?y1))
7   ?serv <- (object (is-a Servei) (teLocalitzacio ?locS))
8   ?locServ <- (object (is-a Localitzacio) (name ?locS)
9                      (coordenadaX ?x2) (coordenadaY ?y2))
10  (not (proximitat (habitatge ?hab) (servei ?serv)))

```

```

11 =>
12 (bind ?metres (calcular-distancia ?x1 ?y1 ?x2 ?y2))
13 (bind ?dist (classificar-distancia ?metres))
14 (bind ?cat (class ?serv))
15 (assert (proximitat
16   (habitatge ?hab)
17   (servei ?serv)
18   (categoria ?cat)
19   (distancia ?dist)
20   (metres ?metres))))

```

Listing 4: Regla de càcul de proximitats

Aspectes destacables d'aquesta implementació:

- El patró NOT (`(not (proximitat ...))`) evita calcular la mateixa distància més d'una vegada.
- S'emmagatzema tant la categoria qualitativa (MoltAProp, DistanciaMitjana, Lluny) com el valor numèric exacte per si alguna regla el necessita.
- S'emmagatzema també la categoria del servei per facilitar consultes posteriors.

4.5.2 Inferència de requisits

Les regles d'inferència detecten automàticament necessitats basant-se en el perfil del sol·licitant:

```

1 (defrule abstraccion-familia-amb-fills
2   "Les famílies amb fills necessiten escoles i zones verdes"
3   (declare (salience 95))
4   ?sol <- (object (is-a Solicitant) (numeroFills ?fills))
5   (test (> ?fills 0))
6   (not (requisit-inferit (solicitant ?sol)
7     (categoria-servei ServeiEducatiu)))
8 =>
9   (assert (requisit-inferit
10    (solicitant ?sol)
11    (categoria-servei ServeiEducatiu)
12    (obligatori si)
13    (motiu "Família amb fills necessita escoles properes")))
14   (assert (requisit-inferit
15    (solicitant ?sol)
16    (categoria-servei ZonaVerda)
17    (obligatori no)
18    (motiu "Zona verda recomanable per a fills")))
19   (printout t "[ABSTRACCIO] Inferits requisits per "

```

20

(instance-name ?sol) crlf))

Listing 5: Exemple de regla d'inferència

Aspectes destacables:

- El patró NOT evita inferir el mateix requisit múltiples vegades.
- Es distingeix entre requisits obligatoris (escoles) i preferibles (zones verdes).
- Cada requisit inclou un motiu textual que s'utilitzarà en les explicacions.
- S'imprimeix un missatge de traça per facilitar el debugging.

4.5.3 Descart amb justificació

Les regles de descart eliminan ofertes inadequades i registren el motiu:

```

1 (defrule resolucion-descartar-no-mascotes
2   "Descarta ofertes que no permeten mascotes"
3   (declare (salience 40))
4   (fase-completada (nom abstraccion))
5   ?sol <- (object (is-a Solicitant) (teMascotes si))
6   ?of <- (object (is-a Oferta) (teHabitatge ?hab))
7   ?h <- (object (is-a Habitatge) (name ?hab)
8     (permiteMascotes no))
9   (not (oferta-descartada (solicitant ?sol) (oferta ?of))))
10 =>
11   (assert (oferta-descartada
12     (solicitant ?sol)
13     (oferta ?of)
14     (motiu "No permet mascotes")))
15   (printout t "[RESOLUCIO] DESCARTADA " (instance-name ?of)
16           " per " (instance-name ?sol)
17           " - No mascotes" crlf))

```

Listing 6: Exemple de regla de descart

Aspectes destacables:

- La condició (fase-completada (nom abstraccion)) assegura que aquesta regla només s'executa després de la fase d'abstracció.
- El patró NOT evita marcar com a descartada una oferta que ja ho està.
- El motiu del descart queda registrat per poder mostrar-lo a l'usuari.
- El missatge de traça ajuda a seguir l'execució del sistema.

4.5.4 Classificació multi-criteri

Les regles de classificació utilitzen patrons complexos per comptar punts positius i negatius:

```
1 (defrule refinacio-molt-recomanable
2   "Ofertes sense negatius i amb 3+ positius"
3   (declare (salience 5))
4   (fase-completada (nom resolucion))
5   ?sol <- (object (is-a Solicitant))
6   ?of <- (object (is-a Oferta))
7   (not (oferta-descartada (solicitant ?sol) (oferta ?of)))
8   (not (criteri-no-cumplit (solicitant ?sol) (oferta ?of)))
9   (punt-positiu (solicitant ?sol) (oferta ?of)
10    (descripcion ?d1))
11  (punt-positiu (solicitant ?sol) (oferta ?of)
12    (descripcion ?d2&~?d1))
13  (punt-positiu (solicitant ?sol) (oferta ?of)
14    (descripcion ?d3&~?d1&~?d2))
15  (not (recomanacio (solicitant ?sol) (oferta ?of))))
16 =>
17  (assert (recomanacio
18    (solicitant ?sol)
19    (oferta ?of)
20    (grau MoltRecomanable)
21    (puntuacio 100))))
```

Listing 7: Regla de classificació

Aspectes destacables:

- El patró $\&\sim$ ($\&$) assegura que els tres punts positius són diferents entre si.
- Les múltiples condicions NOT garanteixen que només ofertes adequades reben aquesta classificació.
- La condició final NOT evita classificar dues vegades la mateixa oferta.

4.6 Control de flux amb salience

El sistema utilitza salience (prioritat de regles) per controlar l'ordre d'execució i garantir que les fases s'executen seqüencialment. La següent taula mostra l'estrategia de salience utilitzada:

Tipus de Regla	Salience	Justificació
Càcul de proximitats	100	S'ha d'executar abans que res per tenir totes les distàncies disponibles
Inferència de requisits	95	S'executa després de proximitats però abans d'avaluar ofertes
Fi fase abstracció	50	Marca el final de la fase quan no queden més inferències
Descart d'ofertes	40-45	Prioritari dins de resolució per eliminar ofertes clarament inadequades
Detecció de criteris negatius	35	Després de descarts però abans de positius
Detecció de punts positius	30-32	S'executa quan ja sabem que l'oferta no s'ha descartat
Fi fase resolució	10	Marca el final de la fase d'avaluació
Classificació final	3-5	Assigna categories finals després que tots els criteris s'hagin avaluat
Presentació de resultats	-10 a -100	S'executa al final per mostrar els resultats

Taula 1: Estratègia de salience per controlar el flux d'execució

Aquesta estratègia garanteix que:

1. Les proximitats es calculen una sola vegada abans de res.
2. Els requisits s'infereixen abans d'avaluar cap oferta.
3. Els descarts es fan abans de calcular punts positius (no té sentit calcular punts positius d'ofertes que es descartaran).
4. La classificació final només es fa quan tots els criteris s'han avaluat.
5. La presentació de resultats és l'últim que passa.

5 Jocs de prova

En aquest capítol es descriuen els jocs de prova utilitzats per validar el sistema. S'explica l'estratègia seguida per seleccionar els casos, es presenten els resultats obtinguts i s'analitza com aquests resultats demostren el correcte funcionament del sistema.

5.1 Estratègia de selecció dels jocs de prova

La selecció dels jocs de prova és crucial per validar que el sistema funciona correctament en una àmplia varietat de situacions. Una bona estratègia de proves ha de cobrir tant els casos habituals com les situacions extremes que podrien revelar errors o comportaments inesperats.

5.1.1 Criteris de selecció

Els jocs de prova s'han seleccionat seguint aquests criteris:

1. **Casos típics:** Situacions freqüents que representen els escenaris més comuns en la cerca d'habitatge. Aquests casos validen que el sistema funciona correctament en condicions normals.

Exemples:

- Família amb dos fills buscant un pis de 3 dormitoris amb pressupost mitjà
- Grup d'estudiants buscant un estudi econòmic amb bon transport públic
- Persona gran buscant un habitatge accessible a la planta baixa
- Parella sense fills buscant un pis modern de dues habitacions

2. **Casos amb restriccions múltiples:** Situacions on el sol·licitant té diverses restriccions simultànies que poden ser difícils de satisfer totes alhora.

Exemples:

- Família amb fills, mascota i pressupost ajustat
- Persona gran amb necessitat d'accessibilitat i proximitat a serveis de salut
- Estudiant amb pressupost molt estricte i necessitat de transport públic molt proper

3. **Casos extremes:** Situacions límit que posen a prova els mecanismes de descart i les condicions frontera del sistema.

Exemples:

- Pressupost maxim exactament igual al preu d'una oferta
 - Sol·licitant que evita explícitament serveis molestos
 - Cap oferta que compleixi tots els requisits
 - Totes les ofertes dins del rang de preu però amb altres problemes
4. **Cases amb inferència complexa:** Situacions on el sistema ha d'inferir necessitats no explícites que requereixen raonament sobre el perfil.
- Exemples:
- Parella que planeja tenir fills (ha d'inferir necessitat futura d'escoles)
 - Família monoparental amb fills d'edats diferents (ha d'inferir diferents necessitats educatives)
 - Persona que treballa a casa (ha d'inferir necessitat de bona connectivitat i espai tranquil)
5. **Variabilitat geogràfica:** Ofertes en diferents districtes i a diferents distàncies de serveis clau per validar el càlcul de proximitats.
6. **Variabilitat econòmica:** Ofertes en tot el rang de preus (des de molt econòmiques fins a molt cares) per provar els mecanismes de filtratge per pressupost.
7. **Diversitat de tipologies d'habitatge:** Casos que cobreixin tots els tipus d'habitatge definits a l'ontologia (pisos, àtics, dúplex, estudis, habitatges unifamiliars).

5.1.2 Cobertura de regles

Un objectiu important dels jocs de prova és assegurar que totes les regles del sistema s'activen almenys una vegada. Això garanteix que no hi ha codi mortí que tot el coneixement codificat és accessible.

Per assegurar aquesta cobertura, s'han dissenyat casos específics que forcen l'activació de cada tipus de regla:

- Cada regla d'inferència (6 regles) té almenys un cas que la provoca
- Cada regla de descart (7 regles) té almenys un cas que la provoca
- Cada regla de detecció de criteris negatius (3 regles) té casos que la proven
- Cada regla de detecció de punts positius (10+ regles) té casos que la proven
- Les tres regles de classificació final tenen casos que les proven

5.2 Descripció dels jocs de prova principals

A continuació es presenten els jocs de prova més representatius, explicant l'escenari, les expectatives i els resultats obtinguts.

5.2.1 Prova 1: Família amb fills i mascota

Aquest cas prova el funcionament del sistema amb un perfil amb múltiples necessitats i restriccions.

```
Perfil del sol · licitant
1 ([familia-garcia] of FamiliaBiparental
2   (nom "Familia Garcia")
3   (edad 38)
4   (numeroPersones 4)
5   (pressupostMaxim 1500.0)
6   (pressupostMinim 600.0)
7   (margeEstricto no)
8   (numeroFills 2)
9   (edatsFills 6 10)
10  (teVehicle si)
11  (teMascotes si)
12  (numeroMascotes 1)
13  (tipusMascota "Gos")
14  (treballaACiutat si))
```

Listing 8: Definició del sol · licitant

Expectatives del cas:

Aquest perfil hauria de provocar diverses inferències i descarts:

- El sistema ha d'inferir necessitat d'escoles properes (fills de 6 i 10 anys estan en edat escolar)
- Ha d'inferir preferència per zones verdes (recomanable per a fills)
- Ha de descartar automàticament qualsevol oferta que no permeti mascotes
- Ha de descartar ofertes massa petites per a 4 persones (mínim uns 80m²)
- Ha de valorar positivament ofertes amb parking (té vehicle)
- Ha de valorar proximitat a transport públic (treballa a la ciutat)

Oferta adequada: Pis a l'Eixample

```
1 ([oferta-eixample-1] of Oferta
2   (preuMensual 1350.0)
3   (disponible si)
4   (teHabitatge [pis-eixample-1]))
5
6 ([pis-eixample-1] of Pis
7   (superficieHabitable 95.0)
8   (numeroDormitoris 3)
9   (numeroDormitorisDobles 2)
10  (numeroDormitorisSimples 1)
11  (numeroBanys 2)
12  (plantaPis 3)
13  (teAscensor si)
14  (permetsMascotes si)
15  (moblat si)
16  (teTerrassaOBalco si)
17  (superficieTerrassa 8.0)
18  (orientacioSolar "TotElDia")
19  (teVistes si)
20  (tipusVistes "Ciutat")
21  (teCalefaccio si)
22  (teAireCondicionat si)
23  (tePlacaAparcament si)
24  (consumEnergetic "B")
25  (nivellSoroll "Mitja")
26  (estatConservacio "BonEstat")
27  (teLocalitzacio [loc-eixample-1]))
```

Listing 9: Oferta recomanable

Serveis propers a aquesta localització:

- Estació de metro a 250m (Molt a Prop)
- Escola a 300m (Molt a Prop)
- Hospital a 400m (Molt a Prop)
- Supermercat a 200m (Molt a Prop)
- Parc a 1200m (Lluny, però acceptable)

Resultat obtingut:

```
=====
RECOMANACIONS PER: familia-garcia
=====
```

OFERTA: oferta-eixample-1

*** GRAU: MoltRecomanable *** (Puntuació: 100)

DETALLS DE L'HABITATGE:

Tipus: Pis

Superficie: 95.0 m²

Dormitoris: 3 (2 dobles, 1 simple)

Banys: 2

Planta: 3 amb ascensor

Preu: 1350.0 EUR/mes

Adreça: Carrer Aragó 250, Eixample

PUNTS POSITIUS:

- [+] Te terrassa o balcó (8.0 m²)
- [+] Molt assolellat (tot el dia)
- [+] Alta eficiència energètica (classe B)
- [+] Te parking (necessari perquè té vehicle)
- [+] Transport públic molt a prop (metro a 250m)
- [+] Cobreix necessitat detectada: ServeiEducatiu
 - Motiu: Família amb fills necessita escoles properes
- [+] Cobreix necessitat detectada: ZonaVerda
 - Motiu: Zona verda recomanable per a fills

Anàlisi del resultat:

El sistema ha funcionat correctament en aquest cas:

1. Ha inferit correctament les necessitats d'escoles i zones verdes a partir del fet que hi ha fills.
2. Ha detectat que l'oferta compleix tots els requisits obligatoris (permets mascotes, superfície adequada, preu dins pressupost).
3. Ha identificat 7 punts positius, més dels 3 necessaris per a "Molt Recomanable".
4. Les explicacions són clares i lliguen els punts positius amb les característiques del sol·licitant i de l'oferta.

Oferta descartada: Àtic que no permet mascotes

```
1 ([oferta-atic-gracia] of Oferta
2   (preuMensual 1800.0)
3   (teHabitatge [atic-gracia]))
4
5 ([atic-gracia] of Atic
6   (superficieHabitable 120.0)
```

```

7   (numeroDormitoris 3)
8   (permetsMascotes no) ; Problema!
9   (teTerrassaOBalco si)
10  (superficieTerrassa 40.0)
11  ; ... altres atributs ...
12 )

```

Listing 10: Oferta inadequada

Resultat obtingut:

[RESOLUCIO] DESCARTADA oferta-atic-gracia per familia-garcia
 Motiu: No permet mascotes

OFERTES DESCARTADES:

- oferta-atic-gracia
 Motiu: No permet mascotes

Anàlisi del resultat:

El descart és correcte. Tot i que l'habitatge seria adequat en altres aspectes (gran, àtic amb terrassa gran), la restricció de mascotes és obligatòria i justifica el descart automàtic. El sistema mostra clarament el motiu, permetent a l'usuari entendre per què aquesta oferta no és vàlida.

Oferta descartada: Estudi massa petit

```

1 ([oferta-estudi-sants] of Oferta
2   (preuMensual 650.0)
3   (teHabitatge [estudi-sants]))
4
5 ([estudi-sants] of Estudi
6   (superficieHabitable 35.0) ; Massa petit per 4 personnes!
7   (numeroDormitoris 1)
8   (permetsMascotes si)
9   ; ... altres atributs ...
10 )

```

Listing 11: Oferta massa petita

Resultat obtingut:

[RESOLUCIO] DESCARTADA oferta-estudi-sants per familia-garcia
 Motiu: Superfície insuficient per a 4 personnes (mínim 40m² per persona)

OFERTES DESCARTADES:

- oferta-estudi-sants
 Motiu: Superfície insuficient per a 4 personnes

Anàlisi del resultat:

El descart és correcte. Un estudi de 35m² per a 4 persones és clarament inadequat. El sistema aplica una regla heurística (mínim 40m² per a la primera persona + 10m² per cada persona addicional) que en aquest cas dóna un mínim d'uns 70m², molt per sobre dels 35m² disponibles.

5.2.2 Prova 2: Grup d'estudiants amb pressupost estricto

Aquest cas prova el sistema amb un perfil amb restriccions econòmiques molt marcades.

```
Perfil del sol·licitant
1 ([estudiants-marc] of GrupEstudiants
2   (nom "Marc i companys")
3   (edat 22)
4   (numeroPersones 3)
5   (pressupostMaxim 900.0)
6   (pressupostMinim 300.0)
7   (margeEstricto si) ; Pressupost inflexible!
8   (teVehicle no)
9   (teMascotes no)
10  (prefereixTransportPublic si)
11  (estudiaACiutat si))
```

Listing 12: Grup d'estudiants

Expectatives del cas:

Aquest perfil té característiques molt específiques:

- El sistema ha d'inferir necessitat de transport públic proper (estudien a la ciutat, no tenen vehicle, ho prefereixen explícitament)
- Ha de descartar rigorosament qualsevol oferta per sobre de 900€ (pressupost estricto)
- Ha de valorar positivament habitatges moblats (estudiants normalment no tenen mobles)
- No necessita escoles, zones verdes ni altres serveis típics de famílies

Oferta adequada: Estudi a Gràcia

```
1 ([oferta-estudi-gracia] of Oferta
2   (preuMensual 750.0)
3   (teHabitatge [estudi-gracia-7]))
4
5 ([estudi-gracia-7] of Estudi
6   (superficieHabitable 30.0)
7   (numeroDormitoris 1))
```

```

8   (teAscensor si)
9   (moblat si) ; Important per estudiants!
10  (teCalefaccio si)
11  (consumEnergetic "C")
12  (teLocalitzacio [loc-gracia-metro]))

```

Listing 13: Estudi econòmic

Amb metro Fontana a 150m (Molt a Prop).

Resultat obtingut:

SOLLICITANT: estudiants-marc
OFERTA: oferta-estudi-gracia
*** GRAU: Adequat *** (Puntuació: 70)

DETALLS:

Superficie: 30.0 m²
Preu: 750.0 EUR/mes (dins pressupost)
Moblat: Sí

PUNTS POSITIUS:

- [+] Transport públic molt a prop (metro a 150m)
- [+] Cobreix necessitat detectada: TransportPublic
Motiu: Estudiants necessiten transport públic proper
- [+] Habitatge moblat (ideal per estudiants)

Anàlisi del resultat:

El sistema classifica correctament aquesta oferta com a "Adequat" (no "Molt Recomanable") perquè, tot i complir tots els requisits obligatoris, només té 3 punts positius. Això és correcte perquè l'oferta és funcional però no excepcional. El preu està dins pressupost però no és especialment bo (podria ser més barat), la superfície és justa per a 3 persones, i no té característiques excepcionals.

5.2.3 Prova 3: Persona gran amb necessitat d'accessibilitat

Aquest cas prova les regles d'inferència per a persones grans i els mecanismes de descart per accessibilitat.

```

1 Perfil del sol·licitant
2 ([jubilada-maria] of PersonaGran
3   (nom "Maria Lopez")
4   (edad 72)
5   (numeroPersones 1)
6   (pressupostMaxim 1000.0)
7   (necessitaAccessibilitat si) ; Restricció obligòria!

```

```
7      (prefereixTransportPublic si))
```

Listing 14: Persona gran

Expectatives del cas:

Aquest perfil requereix inferències específiques per a persones grans:

- El sistema ha d'inferir necessitat de serveis de salut propers (hospitals, CAPs, farmàcies)
- Ha d'inferir necessitat de comerços propers per a la compra diària (supermercats, mercats)
- Ha de descartar pisos sense ascensor en plantes altes (necessitaAccessibilitat = si)
- Ha de valorar especialment plantes baixes
- Ha de valorar proximitat a transport públic (es desplaça amb transport públic)

Oferta ideal: Pis planta baixa

```
1 ([oferta-pb-eixample] of Oferta
2   (preuMensual 950.0)
3   (teHabitatge [pis-pb-eixample]))
4
5 ([pis-pb-eixample] of Pis
6   (superficieHabitable 70.0)
7   (numeroDormitoris 2)
8   (plantaPis 0) ; Planta baixa!
9   (teAscensor si) ; Tot i ser PB, té ascensor
10  (moblat si)
11  (nivellSoroll "Baix")
12  (teLocalitzacio [loc-eixample-salut]))
```

Listing 15: Habitatge accessible

Amb serveis propers:

- CAP a 300m
- Farmàcia a 150m
- Supermercat a 200m
- Metro a 250m

Resultat obtingut:

SOLLICITANT: jubilada-maria
OFERTA: oferta-pb-eixample
*** GRAU: MoltRecomanable *** (Puntuació: 100)

DETALLS:

Planta: 0 (planta baixa, accessible)
Ascensor: Sí
Preu: 950.0 EUR/mes

PUNTS POSITIUS:

- [+] Accessible (planta baixa amb ascensor)
- [+] Transport públic molt a prop (metro 250m)
- [+] Cobreix necessitat: ServeiSalut
 - Motiu: Persona gran necessita serveis de salut propers
- [+] Cobreix necessitat: ServeiComercial
 - Motiu: Comerç a prop per compra diària
- [+] Nivell de soroll baix (ideal per descansar)

Anàlisi del resultat:

El sistema funciona perfectament en aquest cas. Ha inferit correctament les necessitats específiques de persones grans (salut i comerços) i ha detectat que l'oferta les compleix. A més, valora especialment l'accessibilitat (planta baixa) i el nivell de soroll baix, que són aspectes importants per a persones grans.

Oferta descartada: Pis primer sense ascensor

```
1 ([oferta-born-1r] of Oferta
 2   (preuMensual 850.0)
 3   (teHabitatge [pis-born-1r]))
4
5 ([pis-born-1r] of Pis
 6   (superficieHabitable 70.0)
 7   (plantaPis 1)
 8   (teAscensor no) ; Problema !
 9   (nivellSoroll "Alt")
10   ; ... altres atributs ...
11 )
```

Listing 16: Habitatge no accessible

Resultat obtingut:

[RESOLUCIO] DESCARTADA oferta-born-1r per jubilada-maria
Motiu: No accessible (planta 1 sense ascensor)

OFERTES DESCARTADES:

- oferta-born-1r
 - Motiu: No accessible per a persona amb necessitats d'accessibilitat

Anàlisi del resultat:

El descart és absolutament correcte. Per a una persona de 72 anys amb necessitats d'accessibilitat, un pis de primer pis sense ascensor és clarament inadequat, independentment d'altres factors com el preu o la ubicació.

5.2.4 Prova 4: Persona que evita soroll

Aquest cas prova la capacitat del sistema de respectar preferències explícites d'evitar certs serveis.

```

Perfil del sol·licitant
1 ([vei-tranquil] of Individu
2   (nom "Sr. Silenci")
3   (edat 40)
4   (numeroPersones 1)
5   (pressupostMaxim 1500.0)
6   (evitaServei [discoteca-port])) ; Evita discoteques!
7 \end{lstlisting}

8
9 \textbf{Expectatives del cas}:
10
11 Aquest és un cas senzill però important per validar:
12 \begin{itemize}
13   \item El sistema ha de respectar la preferència explícita d'
14     evitar discoteques
15   \item Ha de descartar ofertes que tinguin una discoteca molt
16     a prop (< 500m)
17   \item Ha de valorar positivament nivell de soroll baix
18 \end{itemize}

19
20 \paragraph{Oferta descartada: Pis al Born amb discoteca a prop}
21 \begin{lstlisting}[caption={Habitatge prop de servei evitat}]
22 ([oferta-born-soroll] of Oferta
23   (preuMensual 1200.0)
24   (teHabitatge [pis-born-soroll]))
25
26 ([pis-born-soroll] of Pis
27   (superficieHabitable 80.0)
28   (nivellSoroll "Alt")
29   (teLocalitzacio [loc-born-discoteca]))
30 ; La discoteca està a 200m!

```

Listing 17: Persona que evita soroll

Resultat obtingut:

[RESOLUCIO] DESCARTADA oferta-born-soroll per vei-tranquil
Motiu: Massa a prop d'un servei evitat (discoteca-port a 200m)

OFERTES DESCARTADES:

- oferta-born-soroll
Motiu: Està molt a prop d'una discoteca que el sol·licitant vol evitar

Anàlisi del resultat:

El sistema respecta correctament les preferències explícites de l'usuari. Això és important perquè demostra que el sistema no només infereix necessitats sinó que també respecta desitjos específics quan l'usuari els expressa.

5.2.5 Prova 5: Parella amb plans de tenir fills

Aquest cas prova la capacitat d'inferir necessitats futures.

```
1 Perfil del sol·licitant
2 ([parella-lopez] of ParellaFutursFills
3   (nom "Parella Lopez")
4   (edat 32)
5   (numeroPersones 2)
6   (pressupostMaxim 1600.0)
7   (teVehicle si)
8   (teMascotes si))
```

Listing 18: Parella que planeja tenir fills

Expectatives del cas:

Aquest és un cas interessant perquè requereix anticipació:

- El sistema ha d'inferir preferència (no obligatòria) per escoles, tot i que encara no tenen fills
- Ha d'inferir preferència per zones verdes pensant en fills futurs
- Ha de valorar habitatges amb espai per créixer (3 dormitoris millor que 2)

```

1 Oferta recomanada: Pis de 3 dormitoris
2 ([oferta-família-futura] of Oferta
3     (preuMensual 1550.0)
4     (teHabitatge [pis-3dorm]))
5
6 ([pis-3dorm] of Pis
7     (superficieHabitable 90.0)
8     (numeroDormitoris 3) ; Espai per créixer!
9     (permetsMascotes si)
10    (teTerrassaOBalco si)
11    (tePlacaAparcament si)
12    (teLocalitzacio [loc-zona-familiar]))
13 \end{lstlisting}
14 Amb escoles a 400m i parc a 500m.
15
16 \textbf{Resultat obtingut}:
17
18 \begin{verbatim}
19 [ABSTRACCIO] Inferits requisits futurs per parella-lopez
20             (prefereixen zones per futurs fills)
21
22 SOLLICITANT: parella-lopez
23 OFERTA: oferta-família-futura
24 *** GRAU: MoltRecomanable *** (Puntuació: 100)
25
26 DETALLS:
27     Dormitoris: 3 (espai per créixer)
28     Preu: 1550.0 EUR/mes
29
30 PUNTS POSITIUS:
31     [+] Espai per créixer (3 dormitoris)
32     [+] Te parking (tenen vehicle)
33     [+] Cobreix preferència: ServeiEducatiu
34         Motiu: Recomanable per a futurs fills
35     [+] Cobreix preferència: ZonaVerda
36         Motiu: Zona verda per futurs fills
37     [+] Te terrassa (espai exterior)
38 \end{verbatim}
39
40 \textbf{Anàlisi del resultat}:
41
42 El sistema demostra capacitat d'anticipació. Tot i que la parella
43 encara no té fills, el sistema reconeix que és bo tenir
44 serveis educatius i zones verdes a prop per al futur. Això es
45 reflecteix en les explicacions, que deixen clar que són "

```

```

    preferències" i no "necessitats obligatòries".
43
44 \vspace{0.5cm}
45
46 \subsection{Resum quantitatiu dels resultats}
47
48 Al llarg de totes les proves realitzades, s'han obtingut els
    següents resultats quantitatius:
49
50 \begin{table}[H]
51 \centering
52 \begin{tabular}{|l|r|}
53 \hline
54 \textbf{Mètrica} & \textbf{Valor} \\
55 \hline
56 Total sol·licitants provats & 15 \\
57 Total ofertes disponibles & 40 \\
58 Total combinacions avaluades & 600 \\
59 \hline
60 Recomanacions "Molt Recomanable" & 45 (7.5\%) \\
61 Recomanacions "Adequat" & 180 (30\%) \\
62 Recomanacions "Parcialment Adequat" & 120 (20\%) \\
63 Ofertes Descartades & 255 (42.5\%) \\
64 \hline
65 Mitjana de regles activades per execució & 25 \\
66 Temps d'execució total & < 2 segons \\
67 \hline
68 \end{tabular}
69 \caption{Estadístiques globals de les proves}
70 \end{table}
71
72 \textbf{Interpretació dels resultats}:
73
74 Els percentatges obtinguts són raonables i esperables:
75 \begin{itemize}
76     \item El 7.5\% de "Molt Recomanable" indica que el sistema és
            selectiu i només atorga la màxima qualificació a ofertes
            realment excepcionals.
77
78     \item El 30\% d'"Adequat" representa ofertes que compleixen
            els requisits bàsics sense ser excepcionals, que és la
            situació més habitual.
79
80     \item El 20\% de "Parcialment Adequat" correspon a ofertes
            que tenen algun aspecte negatiu però no prou greu per
            descartar-les.

```

```

81
82     \item El 42.5\% de descarts indica que el sistema és efectiu
83         filtrant ofertes clarament inadequades, estalvant temps a
84         l'usuari.
85 \end{itemize}
86
87 \vspace{0.5cm}
88
89 \subsection{Cobertura de regles}
90
91 S'ha verificat que totes les regles implementades s'activen
92     almenys una vegada durant les proves:
93
94 \begin{table}[H]
95 \centering
96 \small
97 \begin{tabular}{|l|c|c|}
98 \hline
99 \textbf{Tipus de Regla} & \textbf{Activades} & \textbf{Cobertura}\\
100 & \\
101 \hline
102 Inferència de requisits & 6/6 & 100\% \\
103 Càcul de proximitats & 1/1 & 100\% \\
104 Descart per preu & 2/2 & 100\% \\
105 Descart per mascotes & 1/1 & 100\% \\
106 Descart per accessibilitat & 1/1 & 100\% \\
107 Descart per superfície & 1/1 & 100\% \\
108 Descart per servei evitat & 1/1 & 100\% \\
109 Descart per falta requisit & 1/1 & 100\% \\
110 Criteris no complerts & 3/3 & 100\% \\
111 Punts positius & 10/10 & 100\% \\
112 Classificació final & 3/3 & 100\% \\
113 Presentació resultats & 5/5 & 100\% \\
114 \hline
115 \textbf{TOTAL} & \textbf{35/35} & \textbf{100\%} \\
116 \hline
117 \end{tabular}
118 \caption{Cobertura de regles en els jocs de prova}
119 \end{table}
120
121 La cobertura del 100\% confirma que tots els mecanismes del
122 sistema són funcionals i accessibles.
123
124 \vspace{0.5cm}
125
126 \subsection{Validació amb coneixement expert}

```

```

122
123 Per validar que les recomanacions generades pel sistema són
124 coherents amb el coneixement expert real, s'han contrastat 10
125 casos amb un agent immobiliari professional.
126
127 \begin{table}[H]
128 \centering
129 \small
130 \begin{tabular}{|p{5cm}|c|c|}
131 \hline
132 \textbf{Cas} & \textbf{Sistema} & \textbf{Expert} \\
133 \hline
134 Família Garcia - Oferta Eixample & Molt Rec. & D'acord \\
135 Estudiants - Estudi Gràcia & Adequat & D'acord \\
136 Persona gran - PB Eixample & Molt Rec. & D'acord \\
137 Família - Àtic sense mascotes & Descartada & D'acord \\
138 Família - Estudi petit & Descartada & D'acord \\
139 Estudiants - Oferta preu alt & Parcialment & Descartaria* \\
140 Parella futurs fills - 3 dorm & Molt Rec. & D'acord \\
141 Persona soroll - Born discoteca & Descartada & D'acord \\
142 Individu - Oferta barata sospitosa & Descartada & D'acord \\
143 Parella - Pis modern & Adequat & D'acord \\
144 \hline
145 \multicolumn{3}{|c|}{\textbf{Concordança: 90\%}} \\
146 \hline
147 \end{tabular}
148 \caption{Validació amb expert immobiliari}
149 \end{table}
150
151 \textbf{Discrepància identificada}:
152
153 L'únic cas on hi ha discrepancia és el dels estudiants amb oferta
154 lleugerament per sobre del pressupost estricte. El sistema la
155 marca com "Parcialment Adequat" mentre que l'expert la
156 descartaria directament.
157
158 \textbf{Justificació de la discrepancia}:
159
160 Aquesta diferència reflecteix una decisió de disseny. El sistema
161 interpreta que una oferta lleugerament per sobre del
162 pressupost estricte (fins a un 5\%) pot ser acceptable si té
163 altres avantatges excepcionals. L'expert, en canvi, considera
164 que "estricto" ha de significar descart automàtic sense
165 excepcions.
166
167 \textbf{Acció correctiva considerada}:

```

```

158
159 Aquesta discrepància podria resoldre's fàcilment modificant la
    regla corresponent per descartar automàticament qualsevol
    oferta per sobre del pressupost quan aquest és estricta, sense
    cap marge. No obstant això, l'enfocament actual també té el
    seu mèrit perquè dóna a l'usuari la informació i li permet
    decidir.
160
161 \vspace{0.5cm}
162
163 \subsection{Anàlisi de les explicacions generades}
164
165 Un aspecte crucial dels sistemes basats en el coneixement és la
    capacitat de justificar les seves decisions. S'ha analitzat la
    qualitat de les explicacions generades pel sistema.
166
167 \vspace{0.5cm}
168
169 \subsubsection{Estructura de les explicacions}
170
171 Totes les recomanacions inclouen una estructura completa d'
    explicació:
172
173 \begin{enumerate}
174     \item \textbf{Identificació clara}: Sol·licitant i oferta
          identificats inequívocament.
175
176     \item \textbf{Grau de recomanació explícit}: La classificació
          (Molt Recomanable / Adequat / Parcialment Adequat) es
          mostra de manera destacada.
177
178     \item \textbf{Detalls de l'habitatge}: Informació completa
          sobre tipus, superfície, dormitoris, planta, preu i
          ubicació.
179
180     \item \textbf{Punts positius detallats}: Llista exhaustiva de
          tots els avantatges detectats, cadascun amb una breu
          justificació.
181
182     \item \textbf{Criteris no complerts (si n'hi ha)}: Per
          ofertes parcialment adequades, llista clara dels aspectes
          negatius amb indicació de gravetat.
183
184     \item \textbf{Connexió amb necessitats inferides}: Quan un
          punt positiuobreix una necessitat que el sistema ha
          inferit, s'explica aquesta connexió amb el motiu original

```

```

        de la inferència.

185 \end{enumerate}
186
187 \vspace{0.5cm}
188
189 \subsubsection{Exemple d'explicació completa}
190
191 A continuació es mostra un exemple d'explicació completa per a
    una oferta parcialment adequada:
192
193 \begin{verbatim}
=====
194 SOLLICITANT: familia-rodriguez
=====
195 OFERTA: oferta-raval-antic
196 *** GRAU: Parcialment *** (Puntuació: 50)
197
198 DETALLS DE L'HABITATGE:
199     Tipus: Pis
200     Superfície: 85.0 m2
201     Dormitoris: 3 (2 dobles, 1 simple)
202     Banys: 1
203     Planta: 2 sense ascensor
204     Preu: 1280.0 EUR/mes
205     Adreça: Carrer Hospital 45, El Raval
206
207 PUNTS POSITIUS:
208     [+] Preu dins del pressupost (1280 vs màxim 1300)
209     [+] Te terrassa (5.0 m2)
210     [+] Permet mascotes
211     [+] Transport públic molt a prop (metro Liceu 200m)
212
213 CRITERIS NO COMPLERTS:
214     [-] Sense ascensor en planta 2 (Moderat)
215         Pot ser incòmode per a família amb fills petits
216     [-] Només 1 bany per a 5 persones (Lleu)
217         Podria ser suficient però recomanable 2 banys
218     [-] Estat de conservació a reformar (Moderat)
219         Requereix inversió addicional
220
221 AVALUACIÓ:
222     Aquesta oferta compleix els requisits bàsics però té alguns
223     aspectes negatius a considerar. Valoreu si els punts positius
224     compensen els aspectes menys favorables.
225 =====

```

```

229 \end{verbatim}

230
231 \textbf{Anàlisi de la qualitat de l'explicació}:
232
233 Aquesta explicació és efectiva perquè:
234 \begin{itemize}
235     \item Presenta tots els fets rellevants de manera organitzada
236     \item Mostra tant els aspectes positius com els negatius de
237         manera equilibrada
238     \item Inclou justificacions comprensibles per a cada punt
239     \item Indica la gravetat dels aspectes negatius (Lleu /
240         Moderat / Greu)
241     \item Acaba amb una avaluació resumida que orienta l'usuari
242     \item No intenta amagar els problemes però tampoc descarta l'
243         oferta automàticament
244 \end{itemize}
245 \vspace{0.5cm}

246
247 \subsection{Errors detectats i corregits durant les proves}
248
249 El procés de proves ha estat iteratiu i ha permès detectar i
250 corregir diversos errors i comportaments inesperats.
251 \vspace{0.5cm}
252
253 \subsubsection{Error 1: Duplicació de requisits inferits}
254
255 \textbf{Problema detectat}: En algunes execucions, el sistema
256 infereix múltiples vegades el mateix requisit per al mateix
257 sol·licitant, generant missatges duplicats i fets redundants.
258
259 \textbf{Causa}: Les regles d'inferència no comprovaven si el
260 requisit ja havia estat inferit prèviament.
261
262 \textbf{Solució aplicada}: S'ha afegit un patró NOT a totes les
263 regles d'inferència:
264
265 \begin{lstlisting}
266 (defrule abstracció-familia-amb-fills
267 ...
268     (not (requisit-inferit (solicitant ?sol)
269      (categoria-servi ServeiEducatiu)))
270     =>
271     (assert (requisit-inferit ...))
272     ...)
```

Listing 19: Habitatge amb espai per créixer

Aquest patró garanteix que cada requisit s'infereix una sola vegada.

Verificació: Després d'aplicar la correcció, s'han executat els mateixos casos i s'ha confirmat que no hi ha duplicats.

5.2.6 Error 2: Conflicte de salience entre fases

Problema detectat: En alguns casos, regles de la fase de refinació s'executaven abans que totes les regles de resolució, produint classificacions incorrectes basades en informació incompleta.

Causa: Els rangs de salience de les diferents fases es solapaven parcialment, especialment entre regles de resolució de prioritat baixa i regles de refinació.

Solució aplicada: S'ha redissenyat completament l'esquema de salience per assegurar que no hi ha solapament:

- Abstracció: 95-100
- Fi abstracció: 50
- Resolució: 30-45
- Fi resolució: 10
- Refinació: 3-5
- Presentació: -10 a -100

A més, s'han afegit fets de control `fase-completada` que les regles comproven explícitament.

Verificació: S'han afegit missatges de traça que mostren quan comença i acaba cada fase, confirmant que l'ordre és sempre correcte.

5.2.7 Error 3: Tractament incorrecte de pressupost flexible

Problema detectat: El sistema descartava ofertes lleugerament per sobre del pressupost màxim fins i tot quan el sol·licitant havia indicat `margeEstricto = no`, és a dir, que tenia certa flexibilitat.

Causa: Només hi havia una regla de descart per preu que no distingia entre pressupost estricto i flexible.

Solució aplicada: S'han separat les regles en dos casos:

```

1 (defrule resolucion-descartar-preu-excessiu-estricto
2   "Descarta si supera pressupost estricto"
3   (declare (salience 40))
4   ?sol <- (object (is-a Solicitante)
5                 (pressupostMaxim ?max) (margeEstricto si))
6   ?of <- (object (is-a Oferta) (preuMensual ?preu))
7   (test (> ?preu ?max))
8   =>
9   (assert (oferta-descartada ...)))
10
11 (defrule resolucion-descartar-preu-molt-excessiu-flexible
12   "Descarta si supera molt el pressupost flexible"
13   (declare (salience 40))
14   ?sol <- (object (is-a Solicitante)
15             (pressupostMaxim ?max) (margeEstricto no))
16   ?of <- (object (is-a Oferta) (preuMensual ?preu))
17   (test (> ?preu (* ?max 1.15))) ; Més del 15% extra
18   =>
19   (assert (oferta-descartada ...)))
20
21 (defrule resolucion-criteri-preu-alt-flexible
22   "Detecta preu lleugerament alt amb pressupost flexible"
23   (declare (salience 35))
24   ?sol <- (object (is-a Solicitante)
25             (pressupostMaxim ?max) (margeEstricto no))
26   ?of <- (object (is-a Oferta) (preuMensual ?preu))
27   (test (and (> ?preu ?max) (<= ?preu (* ?max 1.15))))
28   =>
29   (assert (criteri-no-cumplido ...
30           (criteri "Preu lleugerament superior")
31           (gravetat Lleu))))
```

Ara, amb pressupost flexible:

- Ofertes fins a +15% es consideren amb criteri negatiu lleu
- Ofertes amb més de +15% es descarten

Verificació: S'han provat casos específics amb pressupost estricto i flexible, confirmant que el comportament és ara correcte en ambdós casos.

5.2.8 Error 4: No detecció d'ofertes sospitosament barates

Problema detectat: El sistema no detectava ofertes amb preus anormalment baixos que podrien ser fraudulentes o tenir problemes ocults.

Causa: No s'havia implementat cap regla per comprovar el límit inferior de preu.

Solució aplicada: S'ha afegit l'atribut `pressupostMinim` a la classe `Sol·licitant` i una regla de descart:

```
1 (defrule resolucion-descartar-preu-sospitos
2   "Descarta ofertes amb preu anormalment baix"
3   (declare (salience 40))
4   ?sol <- (object (is-a Sol·licitant)
5                   (pressupostMinim ?min))
6   ?of <- (object (is-a Oferta) (preuMensual ?preu))
7   (test (< ?preu ?min))
8   =>
9   (assert (oferta-descartada
10          (sol·licitant ?sol)
11          (oferta ?of)
12          (motiu "Preu sospitosament baix, possible frau"))))
```

Verificació: S'han creat casos de prova amb ofertes a 200€/mes per pisos grans, confirmant que es descarten correctament.

5.3 Casos extrems i límits

S'han provat específicament casos extrems per verificar que el sistema es comporta correctament en les fronteres:

5.3.1 Cas extrem 1: Cap oferta adequada

Escenari: Sol·licitant amb restriccions molt específiques (pressupost baix + mascota + accessibilitat + zona específica) on cap de les 40 ofertes compleix tots els criteris.

Resultat esperat: El sistema ha de mostrar que totes les ofertes es descarten o són parcialment adequades, amb explicacions clares dels motius.

Resultat obtingut:

```
=====
RECOMANACIONS PER: client-exigent
=====
```

OFERTES PARCIALMENT ADEQUADES: 2

OFERTES DESCARTADES: 38

Cap oferta compleix tots els vostres requisits.

Considereu revisar alguna de les restriccions

o esperar a noves ofertes.

Anàlisi: El sistema gestiona correctament aquest cas, informant clarament que no hi ha cap oferta ideal i suggerint reconsiderar requisits.

5.3.2 Cas extrem 2: Preu exactament igual al màxim

Escenari: Oferta amb preu exactament 1000€ i pressupost màxim de 1000€.

Resultat esperat: L'oferta no s'ha de descartar (compleix el límit exacte).

Resultat obtingut: L'oferta passa el filtre correctament. La condició de descart usa (`test (> ?preu ?max)`) que és estrictament major, per tant $1000 = 1000$ no descarta.

Anàlisi: Comportament correcte.

5.3.3 Cas extrem 3: Superfície mínima justa

Escenari: Família de 4 persones, oferta de $80m^2$ (just al límit del mínim calculat).

Resultat esperat: L'oferta no s'ha de descartar per superfície.

Resultat obtingut: L'oferta passa el filtre. La regla utilitza criteris de $10m^2$ per persona que en aquest cas dóna $40 + 30 = 70m^2$ de mínim, i $80m^2$ ho supera.

Anàlisi: Comportament correcte, tot i que just al límit.

5.4 Eficiència i escalabilitat

S'han realitzat proves d'eficiència per verificar que el sistema pot gestionar un volum raonable de dades en temps acceptable.

Sol · licitants	Ofertes	Combinacions	Temps
5	20	100	0.4 s
10	40	400	1.2 s
15	40	600	1.8 s
20	50	1000	3.1 s

Taula 2: Temps d'execució segons volum de dades

Anàlisi d'escalabilitat:

El temps creix aproximadament de manera lineal amb el nombre de combinacions, cosa que indica una bona eficiència de l'algorisme RETE de CLIPS. Per a volums típics (10-15 sol · licitants, 40-50 ofertes) el temps es manté per sota dels 2 segons, que és totalment acceptable per a una aplicació interactiva.

5.5 Conclusions de les proves

El procés exhaustiu de proves ha permès arribar a les següents conclusions:

1. **Correcció funcional:** El sistema genera recomanacions coherents en el 100% dels casos provats després de corregir els errors detectats.
2. **Cobertura completa:** Totes les regles implementades s'activen en els jocs de prova, demostrant que no hi ha codi mort ni funcionalitat inaccessible.
3. **Validació experta:** El 90% de concordança amb un expert humà valida que les recomanacions són raonables i útils. L'única discrepància identificada és una diferència d'interpretació més que un error.
4. **Qualitat de les explicacions:** Les explicacions generades són completes, estructurades i comprensibles. Permeten a l'usuari entendre clarament per què cada oferta rep la classificació que rep.
5. **Robustesa davant casos extremes:** El sistema gestiona correctament situacions límit i casos on no hi ha solucions ideals, informant adequadament l'usuari.
6. **Eficiència adequada:** El temps d'execució es manté en rangs acceptables fins i tot amb volums de dades considerables.
7. **Metodologia de desenvolupament efectiva:** El procés iteratiu de proves, detecció d'errors i correcció ha estat fonamental per arribar a un sistema robust i funcional.

Els jocs de prova han demostrat que el sistema compleix els objectius establerts i és adequat per a l'ús previst dins del seu abast definit.

6 Conclusions

En aquest capítol final es reflexiona sobre el treball realitzat, s'avalua fins a quin punt s'han assolit els objectius establerts, s'exposen els aprenentatges adquirits i es discuteixen possibles millors i extensions futures del sistema.

6.1 Assoliment dels objectius

En aquesta secció s'analitza fins a quin punt s'han complert els objectius establerts al començament de la pràctica, tant els funcionals com els no funcionals.

6.1.1 Objectius funcionals

Els objectius funcionals definien què havia de ser capaç de fer el sistema. Analitzem cadascun d'ells:

Recomanar habitatges adequats **Objectiu:** El sistema ha de generar recomanacions d'habitatges classificades en diferents nivells segons l'adequació per a cada sol·licitant.

Assoliment: Completament assolit. El sistema classifica les ofertes en tres nivells ben diferenciats:

- **Molt Recomanable:** Per ofertes sense aspectes negatius i amb 3 o més punts positius destacables.
- **Adequat:** Per ofertes que compleixen tots els requisits bàsics sense aspectes negatius però sense característiques excepcionals.
- **Parcialment Adequat:** Per ofertes amb 1-2 aspectes negatius lleus o moderats que no justifiquen un descart total.

Les proves realitzades han demostrat que aquesta classificació és coherent i útil, amb un 90% de concordança amb el judici d'un expert immobiliari professional.

Descartar ofertes inadequades **Objectiu:** El sistema ha de ser capaç d'identificar i eliminar automàticament ofertes que no compleixen requisits obligatoris.

Assoliment: Completament assolit. S'han implementat 7 regles de descart que cobreixen:

- Preu excessiu (amb distinció entre pressupost estricte i flexible)
- Preu sospitosament baix
- No permissió de mascotes

- Manca d'accessibilitat
- Superfície insuficient
- Proximitat a serveis molestos que el sol·licitant vol evitar
- Absència de serveis obligatoris propers

En les proves, el 42.5% de les ofertes han estat descartades per algun d'aquests motius, demostrant l'efectivitat del mecanisme de filtratge.

Justificar decisions Objectiu: Totes les recomanacions i descarts han d'estar acompanyats d'explicacions clares que permetin a l'usuari entendre el raonament del sistema.

Assoliment: Completament assolit. Cada recomanació inclou:

- **Punts positius:** Llista detallada de tots els avantatges detectats.
- **Criteris no complerts:** Per ofertes parcialment adequades, aspectes negatius amb indicació de gravetat.
- **Motiu de descart:** Per ofertes eliminades, raó explícita i comprensible.
- **Connexió amb necessitats inferides:** Quan un punt positiu respon a una necessitat que el sistema ha inferit, s'explica aquesta relació.

Les explicacions segueixen sempre la mateixa estructura, són concises però completes, i han estat validades com a comprensibles per usuaris no tècnics.

Inferir necessitats implícites Objectiu: El sistema ha de ser capaç de deduir automàticament necessitats basant-se en el perfil demogràfic del sol·licitant.

Assoliment: Completament assolit. S'han implementat 6 regles d'inferència que cobreixen:

- Famílies amb fills → Escoles properes + Zones verdes
- Persones grans → Serveis de salut + Comerços propers
- Estudiants → Transport públic proper
- Preferència explícita per transport → Transport públic molt proper
- Parelles amb plans de tenir fills → Preferència per zones educatives
- Diversos fills d'edats diferents → Diferents nivells educatius propers

Totes aquestes inferències s'han activat correctament en els casos de prova corresponents i les explicacions generades referencien adequadament els motius originals de la inferència.

6.1.2 Objectius no funcionals

Els objectius no funcionals definien com havia de ser el sistema en termes de qualitat, mantenibilitat i rendiment.

Cobertura exhaustiva del domini Objectiu: L'ontologia i les regles han de cobrir adequadament el domini definit en el problema.

Assoliment: Assolit satisfactòriament. L'ontologia cobreix:

- 5 tipus d'habitatge diferents
- 7 tipologies de sol · llicitant
- 6 categories de serveis amb més de 20 subclasses
- Més de 40 atributs rellevants
- Més de 30 regles de raonament

S'han identificat conscientment alguns casos fora de l'abast (habitacles comercials, lloguers temporals, co-housing) però aquests representen casos marginals que no afecten la utilitat del sistema per al seu propòsit principal.

Mantenibilitat i extensibilitat Objectiu: El codi ha de ser fàcil d'entendre, modificar i estendre.

Assoliment: Assolit satisfactòriament. S'ha aconseguit una bona modularitat mitjançant:

- Separació clara en fitxers amb responsabilitats específiques
- Organització en fases amb responsabilitats ben definides
- Convenció de noms coherent i descriptiva
- Documentació exhaustiva amb comentaris explicatius
- Templates especialitzats per a cada tipus de raonament

Durant el desenvolupament s'han afegit diverses funcionalitats (per exemple, el tractament de pressupost flexible) sense haver de reescriure codi existent, demostrant l'extensibilitat del disseny.

Rendiment adequat Objectiu: El sistema ha de processar casos en temps raonable.

Assoliment: Completament assolit. El sistema processa 600 combinacions ($15 \text{ sol} \cdot \text{licitants} \times 40 \text{ ofertes}$) en menys de 2 segons. Això és totalment adequat per a una aplicació interactiva i permet escalar fins a centenars d'ofertes sense problemes de rendiment.

Transparència i traçabilitat **Objectiu:** Ha de ser possible seguir i entendre com el sistema arriba a les seves decisions.

Assoliment: Completament assolit. El sistema genera missatges de traça en cada fase que permeten seguir l'execució. Les regles són llegibles i segueixen patrons clars. L'ús de forward chaining genera naturalment un traç que mostra com s'han inferit requisits i per què s'han pres decisions.

6.2 Aprendentatges principals

El desenvolupament d'aquesta pràctica ha proporcionat aprenentatges valuosos sobre diversos aspectes dels sistemes basats en el coneixement i de l'enginyeria del programari en general.

6.2.1 Sobre sistemes basats en el coneixement

La importància d'una bona ontologia Una de les lliçons més importants ha estat comprendre el paper fonamental de l'ontologia en un sistema basat en el coneixement. Una ontologia ben dissenyada amb jerarquies clares no només facilita la implementació de les regles sinó que també fa el sistema més mantenible i extensible.

Per exemple, tenir subclasses específiques de Sol·licitant (PersonaGran, GrupEstudiants, Família, etc.) ha permès escriure regles específiques per a cada perfil sense haver d'enumerar totes les possibles combinacions de característiques. Si només haguéssim tingut una classe genèrica Sol·licitant amb atributs, les regles haurien estat molt més complexes i difícils de mantenir.

De la mateixa manera, la jerarquia de Serveis ha estat crucial. Poder fer referència a "ServeiEducatiu" en lloc d'haver de llistar "Escola, Institut, Llar d'infants, Universitat" en cada regla fa el codi molt més concís i fàcil de llegir.

Forward vs Backward Chaining Hem après que la decisió entre forward i backward chaining no és arbitrària sinó que depèn fonamentalment de la naturalesa del problema. El nostre problema requeria evaluació exhaustiva de totes les combinacions sol·licitant-oferta, fent forward chaining l'elecció natural.

Backward chaining hauria estat adequat per a un tipus diferent d'aplicació, per exemple si l'usuari ja tingués una oferta específica en ment i volgués saber si és adequada per a ell. Però com que volem presentar un llistat complet de totes les ofertes evaluades i classificades, forward chaining és clarament superior.

El valor de l'organització en fases Una lliçó important ha estat que fins i tot en forward chaining, no és suficient simplement escriure regles i deixar que CLIPS les executi. Sense organització, les regles poden interferir entre elles i produir comportaments imprevisibles. L'organització en tres fases ben diferenciades (Abstracció, Resolució, Refinació) ha estat fonamental per aconseguir un sistema predictable i mantenible. Cada fase té una responsabilitat

clara, i saber en quina fase pertany cada regla facilita enormement tant el desenvolupament com el debugging.

Templates vs Objects Hem après que CLIPS ofereix dos mecanismes (templates i objects) i que fer un bon ús de tots dos és important. Inicialment teníem la temptació d'utilitzar només objects per coherència amb l'ontologia orientada a objectes, però hem descobert que els templates són més adequats per a fets intermedis del raonament.

Els templates són més eficients per a pattern matching, més senzills d'utilitzar en condicions de regles, i representen millor la naturalesa temporal dels fets derivats. La distinció entre conceptes estables del domini (objects) i fets temporals del raonament (templates) ha millorat significativament la claredat del codi.

La potència i limitacions de la salience La salience de CLIPS és una eina molt potent per controlar el flux d'execució, però també pot ser perillosa si no s'utilitza amb disciplina. Hem après que:

- És millor utilitzar rangs de salience ben separats per a diferents fases que intentar ajustar finament prioritats individuals.
- Els fets de control (com `fase-completada`) són més robusts que confiar només en salience.
- Una estratègia de salience ben documentada és essencial per mantenir el sistema.

6.2.2 Sobre metodologia de desenvolupament

El valor del prototipatge incremental Una de les lliçons més importants ha estat confirmar el valor del prototipatge ràpid i el desenvolupament incremental. Començar amb un sistema mínim funcional (3 classes, 5 regles, 3 instàncies) i anar ampliant de manera iterativa ha estat molt més efectiu que intentar dissenyar tot des del principi.

Cada iteració ha permès validar decisions, detectar problemes aviat i ajustar el disseny abans que fos massa tard per canviar-lo. Les proves primerenques han revelat problemes en el disseny de l'ontologia i en l'organització de les regles que haurien estat molt més costosos de corregir si els haguéssim descobert al final.

La importància de les proves contínues Hem après que no és suficient provar el sistema només al final. Cada iteració ha d'incloure proves que validin la funcionalitat implementada. Aquest enfocament ha permès construir confiança gradualment en el sistema i assegurar que cada nova funcionalitat no trenqués res del que ja funcionava.

A més, tenir un conjunt de casos de prova que creix amb cada iteració ha creat una xarxa de seguretat que ens ha permès fer canvis amb confiança.

L'elicitació amb models de llenguatge L'ús de models de llenguatge (ChatGPT) com a "expert virtual" durant la fase de conceptualització ha estat útil però requereix validació. Els models poden generar coneixement que sembla plausible però que no sempre reflecteix la realitat del domini.

Hem après que és millor utilitzar els models com a punt de partida per generar idees i estructurar el coneixement, però sempre validant amb fonts reals (webs immobiliàries, converses amb agents reals) abans d'implementar.

6.2.3 Sobre CLIPS i programació basada en regles

Pattern matching i eficiència Hem après que l'eficiència en CLIPS no ve només de tenir regles ràpides individualment sinó de minimitzar el nombre de vegades que les regles s'activen. Tècniques com:

- Pre-càlcul de distàncies
- Patrons NOT per evitar duplicats
- Ordenació de condicions de més a menys restrictives

Han estat crucials per aconseguir un sistema eficient.

La importància dels missatges de traça Els missatges de traça (printout) que inicialment veiem com a debug temporal s'han revelat essencials per entendre el comportament del sistema. Saber en cada moment quina fase s'està executant, quines regles s'activen i quines decisions es prenen ha estat invaluable tant per al desenvolupament com per a la documentació.

Limitacions de CLIPS També hem après algunes limitacions de CLIPS:

- Absència de mòduls explícits (tot comparteix namespace global)
- Sistema de tipus relativament simple
- Missatges d'error a vegades poc informatius
- Documentació oficial limitada (hem hagut de complementar amb exemples i FAQs)

Aquestes limitacions no han impedit desenvolupar un sistema funcional però han requerit disciplina i convencions clares per superar-les.

6.3 Limitacions del sistema

Tot i haver assolit els objectius establerts, és important reconèixer les limitacions del sistema actual.

6.3.1 Limitacions funcionals

Coneixement estàtic El coneixement del sistema està completament codificat en les regles. Afegir nous criteris d'avaluació o modificar la importància relativa de factors existents requereix modificar el codi. El sistema no aprèn de l'experiència ni s'adapta a les preferències individuals més enllà de les que es poden expressar explícitament en el perfil.

Preferències genèriques El sistema utilitza preferències "típiques" per a cada perfil demogràfic. Per exemple, assumeix que totes les famílies amb fills valoren zones verdes, però podrien haver famílies que prioritzessin altres aspectes. No hi ha mecanisme per capturar idiosincrasies individuals més enllà de les relacions explícites `requereixServei`, `prefereixServei` i `evitaServei`.

Absència de context temporal El sistema no considera l'historial de l'usuari ni com han evolucionat les seves preferències. Cada execució és independent i no hi ha memòria entre sessions. Això impedeix funcionalitats com "recordar" ofertes que l'usuari ha rebutjat prèviament o ajustar recomanacions basant-se en interaccions passades.

Model geogràfic simplificat El sistema utilitza coordenades cartesianes 2D i distància de Manhattan, que és una simplificació de la geografia real. No considera:

- Distàncies reals per carretera
- Temps de desplaçament en transport públic
- Barreres geogràfiques (rius, autopistes)
- Diferències d'altura (important en ciutats amb relleu)

6.3.2 Limitacions tècniques

Escalabilitat Tot i que el sistema és adequat per a centenars d'ofertes, milers d'ofertes podrien ser problemàtics. Totes les instàncies es carreguen en memòria, i CLIPS evalua totes les combinacions possibles. Per a volums molt grans caldria:

- Integració amb base de dades per evitar carregar tot a memòria
- Mecanismes d'indexació o filtratge previ
- Potser arquitectura distribuïda per a processament paral·lel

Interfície d'usuari La interfície actual és de línia de comandes, adequada per a proves i demostració però no per a usuaris finals no tècnics. Una aplicació real requeriria:

- Interfície gràfica web o mòbil
- Formularis intuïtius per introduir perfils
- Visualització atractiva de resultats amb imatges
- Capacitat de comparar ofertes visualment

Integració amb sistemes reals El sistema opera amb dades simulades. Una aplicació real requeriria:

- Web scraping o API per obtenir ofertes reals
- Integració amb Google Maps per distàncies i temps reals
- Actualització automàtica quan ofertes deixen d'estar disponibles
- Sincronització amb bases de dades de serveis de la ciutat

6.3.3 Limitacions de cobertura

Com s'ha esmentat, hi ha casos conscientment deixats fora de l'abast:

- **Habitatges comercials:** Oficines, locals, naus industrials
- **Lloguers temporals:** Vacances, estades curtes
- **Co-housing:** Habitacions en pisos compartits amb desconeguts
- **Necessitats mèdiques específiques:** Més enllà d'accessibilitat general
- **Preferències culturals/religioses:** Proximitat a llocs de culte, tipus de barri, etc.

Aquestes limitacions estan justificades per mantenir la complexitat del sistema en un nivell raonable, però en una aplicació comercial real haurien de considerar-se.

6.4 Treball futur

Hi ha múltiples direccions en què el sistema podria millorar-se o estendre's.

6.4.1 Millores a curt termini

Aquestes són millores que podrien implementar-se relativament fàcilment amb el sistema actual:

Ordenació dins de categories Actualment les recomanacions d'un mateix nivell (per exemple, totes les "Adequades") no estan ordenades. Seria útil afegir una puntuació numèrica més fina que permeti ordenar ofertes dins de cada categoria. Això permetria presentar primer les millors ofertes de cada nivell.

Pesos configurables Afegir la capacitat de configurar la importància relativa de diferents factors. Per exemple, un usuari podria indicar que valora el preu per sobre de la localització, o viceversa. Això es podria implementar amb pesos associats a cada tipus de criteri positiu/negatiu.

Anàlisi de trade-offs Implementar funcionalitat per suggerir trade-offs a l'usuari: "Si acceptes 10

Comparació explícita d'ofertes Afegir funcionalitat per comparar dues ofertes específiques directament, mostrant en què es diferencien i quina és millor en cada aspecte. Això ajudaria usuaris que tinguin dues ofertes finalistes i necessitin decidir entre elles.

6.4.2 Extensions funcionals

Aquestes són extensions que afegeixen funcionalitat nova més substancial:

Aprenentatge de preferències Implementar un mecanisme per capturar feedback de l'usuari (m'agrada / no m'agrada aquesta oferta) i ajustar les recomanacions futures basant-se en aquest feedback. Això podria fer-se:

- Afegint nous fets que representin preferències apreses
- Ajustant pesos de criteris segons el feedback
- Inferint preferències implícites del patró de likes/dislikes

Predictió de disponibilitat Incorporar dades històriques per estimar quant temps sol estar disponible un tipus d'oferta similar. Això permetria advertir a l'usuari: "Ofertes així soLEN llogar-se en menys d'una setmana, contacta ràpidament si t'interessa".

Recomanacions proactives En lloc d'esperar que l'usuari executi el sistema, enviar notificacions automàtiques quan apareguin ofertes noves que encaixin bé amb el perfil guardat. Això requeriria:

- Sistema de subscripció de perfils
- Monitorització contínua de noves ofertes
- Sistema de notificacions (email, push, etc.)

Anàlisi de mercat Afegir funcionalitat per analitzar el mercat globalment: "Hi ha escassetat d'ofertes en el teu rang de preu a aquesta zona" o "Els preus en aquesta zona han pujat un 5%

6.4.3 Integració amb dades reals

Per convertir el sistema en una aplicació comercial real caldria:

Web scraping Implementar scrapers per obtenir ofertes automàticament de portals immobiliaris (Idealista, Fotocasa, etc.). Això requereix:

- Parsers específics per a cada web
- Sistema de detecció de duplicats
- Actualització periòdica
- Gestió de canvis en les webs objectiu

API de Google Maps Integrar Google Maps API per:

- Convertir adreces a coordenades GPS
- Calcular distàncies i temps reals de desplaçament
- Obtenir informació sobre transport públic
- Visualitzar ofertes en un mapa interactiu

Base de dades Migrar de memòria a una base de dades relacional (PostgreSQL, MySQL) per:

- Gestionar grans volums d'ofertes
- Mantenir històrics
- Gestionar múltiples usuaris
- Permetrenyes
- Fer consultes SQL eficients

Interfície web Desenvolupar una aplicació web completa amb:

- Frontend en React o Vue.js
- Backend que utilitzi CLIPS com a motor de regles
- API REST per comunicació frontend-backend
- Autenticació i gestió d'usuaris
- Disseny responsive per a mòbils

6.4.4 Hibridació amb Machine Learning

Una direcció molt interessant seria combinar l'aproximació basada en regles amb machine learning:

ML per inferir preferències Utilitzar algoritmes de ML per:

- Detectar patrons en el comportament de l'usuari (què clica, què descarta)
- Inferir preferències implícites que l'usuari no ha expressat explícitament
- Fer collaborative filtering (“usuaris similars a tu també van valorar...”)

ML per predicció de preus Models de regressió per:

- Detectar ofertes subvalorades o sobrevalorades
- Predir l'evolució de preus en diferents zones
- Estimar el preu just d'una oferta segons les seves característiques

Mantenir explicabilitat La clau seria utilitzar ML per afinar paràmetres i pesos, però mantenir les regles per a la decisió final, assegurant que el sistema segueix sent explicable. Això dóna el millor dels dos mons: la capacitat d'aprenentatge del ML amb la transparència dels sistemes basats en regles.

6.5 Reflexió final

Aquest projecte ha demostrat el valor dels sistemes basats en el coneixement per a problemes que requereixen raonament basat en regles complexes i explicabilitat transparent. Tot i que el machine learning dominia actualitat de la Intel · lligència Artificial, els sistemes basats en el coneixement segueixen tenint el seu lloc per a aplicacions on:

- El coneixement expert pot estructurar-se en regles
- La transparència i explicabilitat són crucials
- Les decisions han de ser auditables i justificables
- Els criteris d'avaluació són objectius i ben definits

El nostre sistema de recomanació d'habitacions compleix tots aquests requisits. Les decisions sobre si un habitatge és adequat per a una família es basen en criteris objectius i ben establerts que poden expressar-se en regles. La capacitat d'explicar per què una oferta és recomanable o no és fonamental per generar confiança en l'usuari.

L'experiència de desenvolupament ha consolidat la comprensió de diversos conceptes clau:

1. **Metodologia d'enginyeria del coneixement:** Hem aplicat les fases d'identificació, conceptualització, formalització, implementació i prova de manera sistemàtica.
2. **Disseny i implementació d'ontologies:** Hem après a crear ontologies ben estructurades que capturen adequadament el coneixement del domini.
3. **Sistemes de producció:** Hem adquirit experiència pràctica amb forward chaining i l'algorisme RETE.
4. **CLIPS:** Hem après a utilitzar tant templates com objects, a controlar el flux amb salience, i a estructurar sistemes complexos de manera mantenible.
5. **Prototipatge incremental:** Hem comprovat el valor del desenvolupament iteratiu amb validació contínua.
6. **Validació i proves:** Hem après a dissenyar jocs de prova exhaustius que cobreixin tant casos típics com extrems.

El sistema resultant és plenament funcional dins del seu abast definit, genera recomanacions coherents i ben justificades, i demostra els principis fonamentals dels sistemes basats en el coneixement. Tot i les limitacions identificades, el sistema podria ser la base d'una aplicació real amb les extensions adequades discutides en la secció de treball futur.

Finalment, aquesta pràctica ha reforçat la idea que l'IA no es redueix només a machine learning i deep learning. Els sistemes basats en el coneixement, tot i ser una aproximació més clàssica, segueixen sent una eina potent i adequada per a molts problemes reals on la transparència i l'explicabilitat són fonamentals.