



Joan Alba Maldonado  
Ivan Andreu Florensa  
Crèdit 13: Crèdit de síntesi  
30 de Maig del 2008

<b>1.- ANÀLISIS I DISSENY D'APLICACIONS INFORMÀTIQUES</b>	3
<b>1.1.-GLOSSARI DE TERMES</b>	3
<b>1.2.-GUIONS D'USUARI</b>	3
1.2.1.-Activitat descrita per el director:	3
1.2.2.-Entrevistes als usuaris	4
<b>1.3.- LLISTAT DE TASQUES D'USUARI</b>	5
1.3.1.-Llistat de tasques dels agents	5
1.3.2.-Llistat de tasques del inspector	6
1.3.3.-Llistat de tasques dels venedors	6
<b>1.4.-CASOS D'US</b>	8
<b>1.5.-MODEL CONCEPTUAL</b>	15
<b>1.6.-DIAGRAMES DE SEQÜENCIA</b>	16
<b>1.5.-DIAGRAMA DE CLASES I RELACIONS</b>	22
<b>1.6.-ESPECIFICACIO DE REQUERIMENTS</b>	23
<b>1.7.-INTERFICIE D'USUARI</b>	24
<b>2.- DOCUMENTACIÓ DE LA BASE DE DADES</b>	31
2.1.-INTRODUCCIÓ	31
2.2.-DIAGRAMA E-R	31
2.3.-TAULES	32
2.4.-MODEL CONCEPTUAL	33
2.5.-TRANSCRIPCIÓ AL SGBD (mySQL)	38
2.6.-EXPLICACIÓ I ACLARIMENTS	42
<b>3.-DOCUMENTACIÓ DEL SOFTWARE D'ADMINISTRACIÓ DE LA BASE DE DADES</b>	46
3.1.-INTRODUCCIÓ	46
3.2.-FUNCIONAMENT INTERN	46
3.3.-FUNCIONAMENT EXTERN	47
<b>4.-DOCUMENTACIÓ DE LA WEB PÚBLICA</b>	48
<b>5.-DOCUMENTACIÓ DE LA WEB INTERNA DEL PERSONAL</b>	51
5.1.-INTRODUCCIÓ	51
5.2.-FUNCIONAMENT INTERN	51
5.3.-FUNCIONAMENT EXTERN	53
<b>6.-CONSIDERACIONS SOBRE L'APLICACIÓ</b>	54

# **1.- ANÀLISIS I DISSENY D'APLICACIONS INFORMÀTIQUES**

## **1.1.-GLOSSARI DE TERMES**

### **Clients**

**Propietari:** Persona o entitat que posseeix un *local comercial* per llogar.

**Llogater Eventual:** Persona o entitat interessada en llogar un *local comercial*.

**Llogater:** Llogater eventual que ha formalitzat un contracte d'un *local comercial*.

### **Usuaris del sistema**

**Agent:** Treballador que s'encarrega de donar d'introduir un nou *local comercial*.

**Inspector:** Treballador que s'encarrega de visitar el *local comercial* per valorar-lo.

**Venedor:** Treballador que atén als *llogaters* i *llogaters eventuais* i formalitza els *contractes*.

### **Locals comercials**

**Local:** Poden ser *polivalents*, *immobles*, *oficines* i *botigues-magatzem*.

**Polivalents:** *Locals* que poden ser llogats com a *oficines* bé *botigues-magatzem*

**Immobles:** *Locals* que poden ser llogats complets o per zones com a *oficines* o bé *botigues-magatzem*.

**Oficines:** *Locals* independents o perteneixents a un immoble, un *local polivalent* destinat a un ús administratiu, de negocis, etc...

**Botigues-magatzem:** Son *locals* independents o perteneixents a un *immoble*, un *local polivalent* destinat a un ús comercial.

### **Altres**

**Contracte:** Document de vinculació entre un *local comercial*, un *llogater* i un *propietari*.

**Text de restriccions:** Informació relacionada amb les limitacions del *local* o imposter a aquest (per exemple, contaminació acústica limitada).

## **1.2.-GUIONS D'USUARI**

### **1.2.1.-Activitat descrita per el director:**

L'agència lloga locals comercials.

Si el propietari esdevé client, està obligat a acceptar el llogater que li proposi aquesta si compleix les condicions estipulades; llavors l'agència prepara el contracte de lloguer del local entre el propietari i el llogater, que sempre és per un temps determinat. Si al venciment del contracte hi ha renovació l'agència prepara un contracte nou.

L'agència pot rebre peticions de lloguer de locals amb característiques determinades per part d'eventuals llogaters i llavors es busquen locals que les tinguin entre els que l'agència té disponibles, i si s'arriba a un acord es prepara el contracte de lloguer

Els usuaris del programari serien els següents

- **Agents:** introdueixen la informació sobre els locals i els seus propietaris
- **Inspectors:** determinen el preu i els usos possibles dels locals
- **Venedors:** preparen els contractes i atenen les peticions de lloguer

### **1.2.2.-Entrevistes als usuaris**

#### **Entrevista als agents**

Dades d'entrada per un local nou:

- Adreça
- Tipus, superfície
- Text de característiques
- Text de restriccions
- El volum (si es tracta d'una botiga-magatzem)
- Text de característiques polivalents (si és un polivalent)
- Codi que conté la zona, el tipus i un número correlatiu creat pel sistema.

En cas de ser un no client (propietari) cal donar:

- La raó social o el nom i cognoms (segons sigui una empresa o un particular)
- Adreça
- Telèfon
- NIF del propietari.

Un cop afegit el queda pendent d'inspecció.

#### **Entrevista al inspector**

El inspector buscarà el local, i llavors comprova tota la informació i pot modificar-la, llevat del codi del local i el text de restriccions.

Introdueix el preu i selecciona d'una llista els usos possibles del local.

Un cop això el local s'afegirà al llistat de locals admesos per alquilar

#### **Entrevista als venedors**

El busca el local i li surten totes les dades sobre el local introduïdes fins al moment. Si tot surt bé, introdueix el NIF del llogater, si el llogater existeix, surten totes les seves dades si no pot donar-lo d'alta en aquell moment amb les següents dades

- NIF
- Nom i cognoms (o la raó social, si és una empresa)
- Telèfon
- Adreça

Llavors, si continua la tasca, crea un contracte de lloguer introduint-ne les dades, que són la data d'inici i la data de fi de la vigència del contracte.

Si confirma la tasca s'imprimeixen les dades del contracte, que són aquestes:

- del propietari, els cognoms i nom o bé la raó social, el NIF i l'adreça;
- del local, l'adreça;
- del llogater, els cognoms i nom o bé la raó social, el NIF i l'adreça;
- les dates d'inici i de fi de la vigència.

### ***1.3.- LLISTAT DE TASQUES D'USUARI***

#### **1.3.1-Llistat de tasques dels agents**

##### **AltaLocal**

**Meta:** Donar d'alta un nou local.

**Descripció:** L'agent donarà d'alta un nou local introduint les dades següents: adreça, tipus (com a recomanació o exigència del propietari), superfícies, text de característiques, text de restriccions, el volum (en cas de botiga-magatzem o polivalent), i el text de característiques polivalent (en cas de polivalent). En cas de ser un immoble, s'introduiran primer la quantitat d'oficines, botigues-magatzem i/o polivalents i després pregunta les dades de cadascun d'aquests locals.

##### **ModificarLocal**

**Meta:** Modificació de la informació d'un local.

**Descripció:** El inspector podrà modificar les dades d'un local llevat del codi del local, el preu i l'ús del local. En cas de ser un immoble, es podran modificar o donar de baixa els locals que el componen.

##### **BaixaLocal**

**Meta:** Donar de baixa un local.

**Descripció:** Donar de baixa un local sense eliminar-lo del sistema (si es immoble, es donaran de baixa automàticament tots els locals que el componen).

##### **CercaLocal**

**Meta:** Trobar locals amb criteris de cerca i veure la seva informació.

**Descripció:** Buscar un local amb uns criteris de cerca definits prèviament. Els criteris seran: tipus del local, zona o zones, superfície mínima desitjada, període de disponibilitat (data d'inici i data de fi, opcional), preu (opcional).

##### **VeureInformacioLocal**

**Meta:** Veure tota la informació d'un local.

**Descripció:** L'agent per alguna raó necessita obtenir la informació d'un local (encara que no necessiti modificar-la). En cas de ser un immoble, es veuran les dades d'aquest i els locals que el componen.

##### **AltaPropietari**

**Meta:** Donar d'alta un nou propietari.

**Descripció:** Es donarà d'alta un nou propietari i, si no existeix com a client en el sistema, introduirà les dades següents: raó social o nom i cognoms, NIF, adreça, telèfon. En el cas d'una empresa, es poden ficar més d'una adreça i d'un telèfon.

##### **ModificarPropietari**

**Meta:** Modificació de la informació d'un propietari

**Descripció:** Es podran modificar totes les dades d'un propietari. En el cas d'una empresa, es poden modificar totes les seves adreces i telèfons.

**BaixaPropietari**

**Meta:** Donar de baixa un propietari.

**Descripció:** Donar de baixa un propietari sense eliminar-lo del sistema (podrà continuar sent llogater donat el cas).

**CercarClients**

**Meta:** Trobar clients amb criteris de cerca i veure la seva informació o operar amb ells.

**Descripció:** Buscar un client amb uns criteris de cerca definits prèviament. Els criteris seran: NIF (opcional), zona (opcional), si es propietari (opcional), si es llogater (opcional), nom i cognoms (opcional). Es mostraran els clients i algunes dades d'aquests. Podrà escollir donar d'alta com a propietari al client, en cas de que no ho estigui ja, o donar-lo de baixa si ho està. També podrà modificar les seves dades. Deixarà accedir a una fitxa del client amb dades ampliades (locals de la seva propietat, locals llogats, etc...).

**1.3.2.-Llistat de tasques del inspector****VeureLocalsPendants**

**Meta:** Es mostraran tots els locals pendants de inspecció.

**Descripció:** El inspector veurà un llistat dels locals donats de alta per l'agent.

**VeureInformacioLocal**

**Meta:** Veure tota la informació d'un local.

**Descripció:** El inspector per alguna raó necessita obtenir la informació d'un local (encara que no necessiti modificar-la). En cas de ser un immoble, es veuran les dades d'aquest i els locals que el componen.

**ModificarLocal**

**Meta:** Modificació de la informació d'un local.

**Descripció:** El inspector podrà modificar les dades d'un local llevat del codi del local i el text de restriccions, i introduir el preu i l'ús del local. En cas de ser un immoble, es podran modificar els locals que el componen.

**1.3.3.-Llistat de tasques dels venedors****AltaLlogater/AltaLlogaterEventual**

**Meta:** Donar d'alta un llogater eventual.

**Descripció:** Es donarà d'alta un nou llogater eventual, si no existeix com a client, introduirà les dades següents: raó social o nom i cognoms, NIF, adreça, telèfon i tipus de local, zona o zones, i superfície mínima desitjada. Si ja existeix com a client, solament introduirà: tipus de local, zona o zones i superfície mínima desitjada. Si el llogater ja existeix s'introduiran només les ultimes 3 dades.

**ModificarLlogater/ModificarLlogaterEventual**

**Meta:** Modificació de la informació d'un llogater.

**Descripció:** Es podran modificar totes les dades d'un llogater.

### **BaixaLlogater/BaixaLlogaterEventual**

**Meta:** Donar de baixa un llogater.

**Descripció:** Donar de baixa un llogater sense eliminar-lo del sistema (podrà continuar sent propietari donat el cas).

### **LlistarClients**

**Meta:** Mostrar una llista de tots els clients per a obtenir informació o operar amb ells.

**Descripció:** El venedor accedeix a una llista on es mostren els clients i algunes dades d'aquests. Podrà escollir donar d'alta com a llogater al client, en cas de que no ho estigui ja, o donar-lo de baixa si ho està. També podrà modificar les seves dades i cercar els locals de les seves preferències. Deixarà accedir a una fitxa del client amb dades ampliades (locals de la seva propietat, locals llogats, etc...).

### **CercarClients**

**Meta:** Trobar clients amb criteris de cerca i veure la seva informació o operar amb ells.

**Descripció:** Buscar un client amb uns criteris de cerca definits prèviament. Els criteris seran: NIF (opcional), zona (opcional), si es propietari (opcional), si es llogater (opcional), nom i cognoms (opcional). Es mostraran els clients i algunes dades d'aquests. Podrà escollir donar d'alta com a llogater al client, en cas de que no ho estigui ja, o donar-lo de baixa si ho està. També podrà modificar les seves dades i cercar els locals de les seves preferències. Deixarà accedir a una fitxa del client amb dades ampliades (locals de la seva propietat, locals llogats, etc...).

### **VeureInformacioLocal**

**Meta:** Veure tota la informació d'un local.

**Descripció:** El venedor per alguna raó necessita obtenir la informació d'un local (encara que no necessiti modificar-la).

### **CercaLocalPreferenciesLlogater**

**Meta:** Buscar un local convenient per un llogater eventual i veure la seva informació.

**Descripció:** Es buscarà un local mitjançant les preferències introduïdes en la creació d'un llogater eventual, podent ocultar prèviament les dades desitjades. Les opcions per a ocultar seran: raó social o nom i cognoms, NIF, adreça, telèfon.

### **CercaLocal**

**Meta:** Trobar locals amb criteris de cerca i veure la seva informació.

**Descripció:** Buscar un local amb uns criteris de cerca definits prèviament, podent ocultar prèviament les dades desitjades. Els criteris seran: tipus del local, zona o zones, superfície mínima desitjada, període de disponibilitat (data d'inici i data de fi, opcional), preu (opcional), característiques (paraula o paraules escrites que coincideixin, opcional). Les opcions per a ocultar seran: raó social o nom i cognoms, NIF, adreça, telèfon.

### **ImprimirLlistaLocals**

**Meta:** Imprimir la llista de locals prèviament cercats.

**Descripció:** S'imprimirà el llistat de locals que el venedor haurà buscat prèviament.

**CrearContracte**

**Meta:** Efectuar el lloguer d'un local.

**Descripció:** Formalitzar un contracte. Les dades a introduir seran: NIF del llogater, codi del local, data d'inici i data de fi de la vigència del contracte. Al finalitzar el contracte serà imprès.

**BaixaContracte**

**Meta:** Donar de baixa un contracte.

**Descripció:** Donar de baixa un contracte sense eliminar-lo del sistema.

**ModificarContracte**

**Meta:** Modificació d'un contracte

**Descripció:** Modificar un contracte. Les dades a modificar serà la data de fi de la vigència del contracte. Al finalitzar el contracte serà imprès.

### ***1.4.-CASOS D'US***

**Cd'u 1. Alta Local**

**Funcionalitat:** Donar d'alta un nou local.

**Funció per l'usuari:** Pertany a la tasca AltaLocal (Agent).

**Paper dins el treball de l'usuari:** Bàsic.

**Actors:** Agent.

**Cd'u relacionats:** Cap.

**Precondició:** El propietari ha d'estar donat d'alta prèviament mitjançant el cas d'us AltaClient

**Postcondició:** Es donarà d'alta a la base de dades un nou local

**Procés normal principal:**

1. Es mostra un formulari per introduir les dades del local: adreça, tipus (com a recomanació o exigència del propietari), superfícies, text de característiques, text de restriccions, el volum (en cas de botiga-magatzem o polivalent), i el text de característiques polivalent (en cas de polivalent). No deixarà introduir un immoble si ja estem introduint locals dins d'un.
2. L'agent omple el formulari amb les dades donades pel propietari i l'envia.
  - 2.1. En cas d'un immoble:
    - 2.1.1. El sistema genera un codi per el immoble que conté la zona, el tipus i un numero correlatiu.
    - 2.1.2. S'introdueix en la base de dades el nou immoble.
    - 2.1.3. El sistema pregunta el nombre de botigues-magatzem, d'oficines i/o polivalents que té el immoble.
    - 2.1.4. L'usuari introdueix les dades i les envia.
    - 2.1.5. El sistema torna al punt 1 tantes vegades com locals l'agent hagi volgut introduir i s'introduiran dins d'aquest immoble.
3. Es mostra un formulari per introduir el NIF del propietari
  - 3.1. El sistema genera un codi per el local que conté la zona, el tipus i un numero correlatiu.
  - 3.2. S'introdueix en la base de dades el nou local.
  - 3.3. El sistema mostrarà una pantalla d'avis informant de com ha anat l'operació

**Alternatives de procés i excepcions:**

\*a. L'agent cancel·la el procés:

\*a1. El sistema acaba el cd'u.

3a. Si el propietari no existeix, es criada al cas d'us AltaClient.

2.1.2a i 3.2a. El local o immoble ja pot existir a la base de dades

2.1.2a1 i 3.2a1. Es mostrarà una advertència i retornarà al punt immediatament anterior.

## **Cd'u 2. Modificar Local**

**Funcionalitat:** Modificar les dades un local.

**Funció per l'usuari:** Pertany a la tasca ModificarLocal (Agent i Inspector)

**Paper dins el treball de l'usuari:** Esporàdic (Agent). Bàsic (Inspector)

**Actors:** Agent i Inspector

**Cd'u relacionats:** Cd'u Cercar Locals, Cd'u Baixa Local, Cd'u Crear Contracte

**Precondició:** El local ha d'estar donat d'alta a la base de dades

**Postcondició:** Es modificaran les dades del local per introduir-les de nou a la base de dades

### **Procés normal principal:**

1. En cas de ser l'agent, haurà buscat el local a modificar a través del cd'u Cercar Local o VeureInformacioLocal:

1.1. Es mostrarà un formulari amb tots els camps llevat del codi del local, el preu i l'ús del local.

2. En cas de ser un inspector haurà accedit mitjançant el cd'u Llistar Locals Pendants o VeureInformacioLocal:

2.1. Es mostrarà un formulari amb tots els camps llevat del codi del local. El formulari també contindrà l'opció d'aprovar el local com a vàlid.

3. L'agent o el inspector omplen el formulari

4. S'introdueixen els canvis a la base de dades.

### **Alternatives de procés i excepcions:**

\*a. L'agent cancel·la el procés:

\*a1. El sistema acaba el cd'u.

## **Cd'u 3. Baixa Local**

**Funcionalitat:** Donar de baixa un local de les dades un local.

**Funció per l'usuari:** Pertany a la tasca BaixaLocal preteneixen a les tasques de l'agent

**Paper dins el treball de l'usuari:** Esporàdic

**Actors:** Agent

**Cd'u relacionats:** Cd'u Modificar Local, Cd'u Llistar Locals Pendants, Cd'u Cercar Locals

**Precondició:** El local ha d'estar donat d'alta a la base de dades

**Postcondició:** Es donarà de baixa un local sense eliminar-lo posteriorment de la base de dades

### **Procés normal principal:**

1. L'agent buscarà el local a modificar a través del cd'u Cercar Local

2. Apareixerà tota la informació del local en pantalla

3. Es seleccionarà l'opció d'eliminar el local

4. Apareixerà en pantalla una opció de donar de baixa o no

5. Si es confirma, es donarà de baixa de la base de dades el local (no s'eliminarà)

**Alternatives de procés i excepcions:**

\*a. L'agent cancel·la el procés:

\*a1. El sistema acaba el cd'u.

**Cd'u 4. Cercar Local**

**Funcionalitat:** Mostrar un llistat de locals

**Funció per l'usuari:** Pertany a les tasques CercaLocal (Agent i Venedor), CercaLocalPreferenciesLlogater (Venedor).

**Paper dins el treball de l'usuari:** Esporàdic

**Actors:** Agent i Venedor

**Cd'u relacionats:** Cd'u Modificar Local, Cd'u Baixa Local, Cd'u Crear Contracte

**Precondició:** El local ha d'estar donat d'alta a la base de dades.

**Postcondicio:** Es donarà de baixa un local sense eliminar-lo posteriorment de la base de dades.

**Procés normal principal:**

1. Si s'arriba des de veure locals que coincideixin amb les preferències d'un llogater (solament venedor):

1.1. Es seleccionerà tot un llistat de locals amb les preferències d'aquell client.

2. Si s'arriba des de Veure Informació Client:

2.1. Si el client es un propietari, es seleccionaran els locals que posseeix.

2.2. Si es llogater, es seleccionerà els locals que te contractats.

3. En cas contrari:

3.1. Si no s'ha definit mostrar tots (el qual es tracta com a un criteri de cerca també), es mostrarà un formulari amb els possibles filtres que es poden aplicar a la cerca del local i amb opcions de ocultar les dades raó social o nom i cognoms, NIF, adreça, telèfon.

4. Es llistaran tots els locals que resultin de la cerca amb l'opció de mostrar els camps ocultats. El venedor tindrà l'opció d'imprimir el llistat mostrat per pantalla. L'agent tindrà l'opció de donar de baixa el local (mitjançant el cd'u Baixa Local), modificar local (mitjançant el cd'u Modificar Local). Ambdós podran veure la informació ampliada mitjançant el cd'u Veure Informació Local.

**Alternatives de procés i excepcions:**

1.1a. La cerca de locals no ha donat cap resultat.

2.1a. La cerca de locals no ha donat cap resultat.

3.1a. La cerca de locals no ha donat cap resultat.

**Cd'u 5. Llistar Locals Pendants**

**Funcionalitat:** Mostrar el llistat dels locals pendants d'inspecció

**Funció per l'usuari:** Pertany a la tasca VeureLocalsPendants del inspector

**Paper dins el treball de l'usuari:** Bàsic

**Actors:** Inspector

**Cd'u relacionats:** Cd'u Modificar Local

**Precondició:** L'agent haurà d'haver donat d'alta un local (Cd'u 1 Alta Local).

**Postcondicio:** Cap.

**Procés normal principal:**

1. Es llistaran tots els locals que encara estiguin pendants d'inspecció.

2. El inspector selecciona un local de la llista i podrà modificar-lo mitjançant el Cd'u Modificar Local

**Alternatives de procés i excepcions:**

\*a. L'agent cancel·la el procés:

\*a1. El sistema acaba el cd'u.

1a. El llistat de locals pendents d'inspecció esta buit.

**Cd'u 6. Veure Informació Local**

**Funcionalitat:** Mostrar la informació completa d'un local

**Funció per l'usuari:** Pertany a la tasca VeureInformacioLocal (Agent, Inspector i Venedor).

**Paper dins el treball de l'usuari:** Esporàdic

**Actors:** Agent, Inspector i Venedor

**Cd'u relacionats:** Cd'u Modificar Local, Cd'u Cercar Locals, Cd'u Veure Informació Client

**Precondició:** Com a mínim hi ha d'haver un local donat d'alta a la base de dades. En el cas de ser un inspector, és necessari com a mínim un local pendent d'inspecció. En el cas de venedor, ha d'estar aprovat.

**Postcondicio:** Cap

**Procés normal principal:**

1. L'usuari arriba des del cd'u Cercar Locals o des de Veure Informació:
2. Es mostrarà el local amb els possibles filtres (raó social o nom i cognoms, NIF, adreça, telèfon) que s'hagin pogut aplicar a la cerca del local. En el cas de ser aplicable, es llistarà el propietari i, si té, el llogater i l'usuari podrà veure informació ampliada d'aquests mitjançant un vincle o similar (cridarà a Veure Informació Client).

**Alternatives de procés i excepcions:**

\*a. L'agent cancel·la el procés:

\*a1. El sistema acaba el cd'u.

**Cd'u 7. Alta Client**

**Funcionalitat:** Donar d'alta un nou client a la base de dades

**Funció per l'usuari:** Pertany a les tasques donar AltaPropietari (Agent) i AltaLlogater/AltaLlogaterEventual (Venedor)

**Paper dins el treball de l'usuari:** Basic

**Actors:** Agent i Venedor

**Cd'u relacionats:** Cap

**Precondició:** Cap

**Postcondicio:** Es donarà d'alta a la base de dades un nou client

**Procés normal principal:**

1. Es mostra un formulari per introduir les dades del client: raó social o nom i cognoms, NIF, adreça, telèfon.
  - 1.1 En cas de ser un agent qui doni d'alta un client, es mostraran a més a més els camps: tipus de local, zona o zones i superfície mínima desitjada
2. L'agent o el venedor omple el formulari amb les dades donades per l'agent i l'envia
3. S'introdueixen les dades del client a la base de dades.

**Alternatives de procés i excepcions:**

\*a. L'agent cancel·la el procés:

\*a1. El sistema acaba el cd'u.

2a. El client ja pot existir a la base de dades.

2a1. En cas de ser venedor i que el client no existeixi com a llogater.

- 2a1a. Es mostrarà un avis, i s'introduiran les preferències: tipus de local, zona o zones i superfície mínima desitjada, la resta de dades no es canviaran.
- 2a2. En cas de ser el venedor i ja estar donat d'alta com a llogater, o en el cas de ser el agent i ja estar en el sistema com a propietari:
- 2a2a. Es mostrarà un avis, i es tornarà al punt 1.

## **Cd'u 8. Modificar Client**

**Funcionalitat:** Modificar les dades d'un client

**Funció per l'usuari:** Pertany a les tasques donar ModificarPropietari (Agent) i ModificarLlogater/ModificarEventual (Venedor)

**Paper dins el treball de l'usuari:** Esporàdic

**Actors:** Agent i Venedor

**Cd'u relacionats:** Cd'u Baixa Client, Cd'u Cercar Client, Cd'u Crear Contracte

**Precondició:** El client a modificar ja haurà d'estar donat d'alta a la base de dades

**Postcondicio:** Es modificaran les dades del client per introduir-les de nou a la base de dades

### **Procés normal principal:**

1. En cas de ser l'agent, haurà buscat el client a modificar a través del cd'u Cercar Clients o VeureInformacioClient:

1.1. Es mostrarà un formulari amb tots els camps.

2. En cas de ser un inspector, haurà accedit mitjançant el cd'u Cercar Clients o VeureInformacioClient:

2.1. Es mostrarà un formulari amb tots els camps.

3. L'agent o el inspector omplen el formulari

4. S'introdueixen els canvis a la base de dades.

### **Alternatives de procés i excepcions:**

\*a. L'agent cancel·la el procés:

\*a1. El sistema acaba el cd'u.

## **Cd'u 9. Baixa Client**

**Funcionalitat:** Donar de baixa un client de la base de dades.

**Funció per l'usuari:** Pertany a les tasques BaixaPropietari (Agent) i BaixaLlogater/BaixaLlogaterEventual (Venedor)

**Paper dins el treball de l'usuari:** Esporàdic

**Actors:** Agent i Venedor

**Cd'u relacionats:** Cd'u Modificar Client, Cd'u Cercar Client

**Precondició:** Que hi hagi almenys un client a la base de dades.

**Postcondicio:** S'haurà esborrat un client de la base de dades.

### **Procés normal principal:**

1. L'agent haurà accedit aquí amb l'opció de donar de baixa a través del cd'u Cercar Clients o Veure Informació Client:

2. Apareixerà en pantalla una opció de donar de baixa o no

3. Si es confirma, es donarà de baixa de la base de dades el client (no s'eliminarà)

### **Alternatives de procés i excepcions:**

\*a. L'agent cancel·la el procés:

\*a1. El sistema acaba el cd'u.

## **Cd'u 10. Cercar Clients**

**Funcionalitat:** Obtenir una llista dels propietaris i/o llogaters

**Funció per l'usuari:** Pertany a les tasques CercaClients (Agent i Venedor)

**Paper dins el treball de l'usuari:** Freqüent

**Actors:** Agent i Venedor

**Cd'u relacionats:** Cd'u Modificar Client, Cd'u Baixa Client, Cd'u Crear Contracte

**Precondició:** Que hi hagi almenys un client a la base de dades.

**Postcondicio:** Cap

### **Procés normal principal:**

1. Si no s'ha definit mostrar tots (el qual es tracta com a un criteri de cerca també), es mostrarà un formulari amb els possibles filtres que es poden aplicar a la cerca del client.
2. En cas de que no s'hagi definit mostrar tots, l'usuari omple el formulari.
3. Es llistaran tots els clients que resultin de la cerca. Si no ha hagut cerca, es mostren tots.
4. En cas de ser un agent qui llisti:
  - 4.1. Es mostrarà l'opció de donar d'alta com a propietari (si no ho és ja), modificar o donar-lo de baixa (si ja es propietari) o veure la seva informació ampliada mitjançant el cd'u Veure Informació Client.
5. En cas de ser un venedor qui llisti:
  - 5.1. Es mostrarà l'opció de donar d'alta com a llogater (si no ho es ja), modificar o donar-lo de baixa (si ja es llogater) o veure la seva informació ampliada mitjançant el cd'u Veure Informació Client. També tindrà l'opció de cercar els locals amb les preferències del llogater mitjançant el cd'u Cerca Locals. Podrà veure la informació ampliada mitjançant el cd'u Veure Informació Client.

### **Alternatives de procés i excepcions:**

\*a. L'agent cancel·la el procés:

\*a1. El sistema acaba el cd'u.

## **Cd'u 11. Veure Informació Client**

**Funcionalitat:** Accedir a una fitxa del client amb dades ampliades

**Funció per l'usuari:** Pertany a les tasques LlistarClients (Agent i Venedor)

**Paper dins el treball de l'usuari:** Esporàdic

**Actors:** Agent i Venedor

**Cd'u relacionats:** Cd'u Cercar Client, Cd'u Veure Informacio Local

**Precondició:** Que hi hagi almenys un client a la base de dades.

**Postcondicio:** Cap.

### **Procés normal principal:**

1. L'agent o el venedor arribaran a aquí mitjançant el cd'u Cercar Clients o Veure Informació Local. El sistema mostrarà informació ampliada del client (locals propis, si es propietari, locals llogats si es llogater...). En cas de llistar-se locals, el usuari podrà veure informació ampliada d'aquestos mitjançant un vincle o similar (cridarà a Veure Informació Local).

### **Alternatives de procés i excepcions:**

\*a. L'agent cancel·la el procés:

\*a1. El sistema acaba el cd'u.

## **Cd'u 12. Crear Contracte Locals**

**Funcionalitat:** Formalitzar un contracte amb possibilitat d'imprimir .

**Funció per l'usuari:** Pertany a la tasca CrearContracte (Venedor)

**Paper dins el treball de l'usuari:** Frequent.

**Actors:** Venedor

**Cd'u relacionats:** Cd'u Veure Informació Local, Cd'u Cercar Local, Cd'u Cercar Clients, Cd'u Veure Informació Client

**Precondició:** Com a mínim ha d'haver-hi un propietari, un local i un llogater.

**Postcondicio:** Es formalitzarà un contracte.

**Procés normal principal:**

1. El venedor arriba aquí havent seleccionat un llogater o un local:
2. Haurà d'omplir les dades que manquen i son necessàries mitjançant un formulari: local o llogater (depenent si s'ha entrar ficant el llogater o un local, respectivament), data d'inici i data de fi.
3. S'introduirà el contracte en la base de dades. També es mostrarà la informació d'imprimir el contracte.
4. Si així ho desitja, el venedor podrà imprimir el contracte (molt freqüent).

**Alternatives de procés i excepcions:**

\*a. L'agent cancel·la el procés:

\*a1. El sistema acaba el cd'u.

### **Cd'u 13. Modificar Contracte**

**Funcionalitat:** Prorrogar un contracte

**Funció per l'usuari:** Pertany a les tasques ModificarContracte (Venedor)

**Paper dins el treball de l'usuari:** Esporàdic

**Actors:** Venedor

**Cd'u relacionats:** Cap

**Precondició:** Ha d'haver com a mínim un contracte a la base de dades i el llogater i el local han de continuar donats d'alta.

**Postcondicio:** El contracte s'haurà prorrogat

**Procés normal principal:**

1. Es mostrarà un formulari permetin canviar la data de fi de contracte.
2. L'usuari introdueix una data.
3. El sistema modificarà la informació del contracte a la base de dades. També es mostrarà la informació d'imprimir el contracte.
4. Si així ho desitja, el venedor podrà imprimir el contracte (molt freqüent).

**Alternatives de procés i excepcions:**

\*a. L'agent cancel·la el procés:

\*a1. El sistema acaba el cd'u.

### **Cd'u 14. Baixa Contracte**

**Funcionalitat:** Donar de baixa un contracte de la base de dades (sense eliminar-lo).

**Funció per l'usuari:** Pertany a la tasca BaixaContracte (Venedor)

**Paper dins el treball de l'usuari:** Esporàdic

**Actors:** Venedor

**Cd'u relacionats:** Cap

**Precondició:** Ha d'haver com a mínim un contracte a la base de dades.

**Postcondicio:** El contracte es donarà de baixa de la base de dades (no s'eliminarà)

**Procés normal principal:**

1. Apareixerà en pantalla una opció de donar de baixa o no.

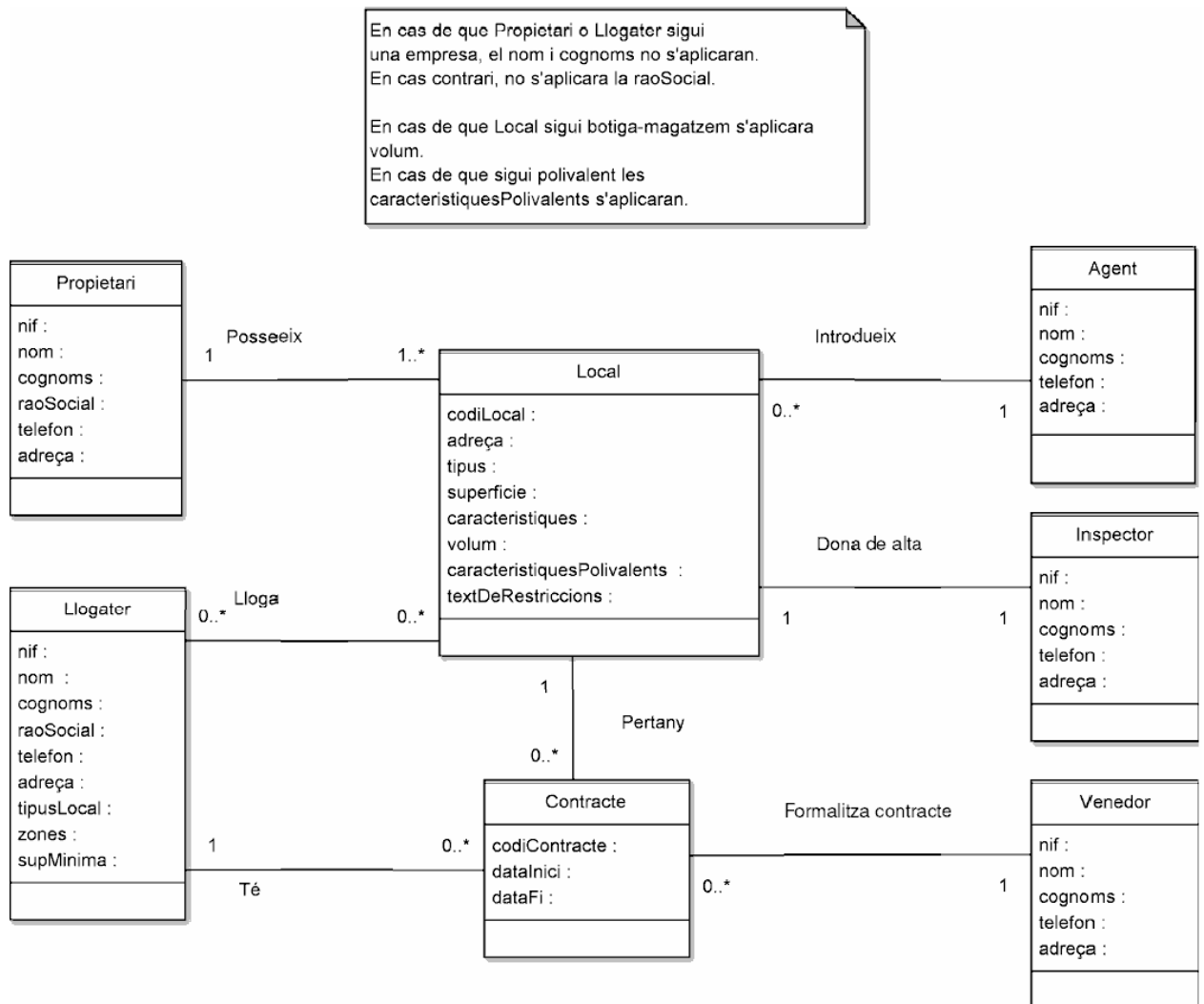
2. Si es confirma, el sistema donarà de baixa el contracte de la base de dades (no s'eliminarà).

#### Alternatives de procés i excepcions:

\*a. L'agent cancel·la el procés:

\*a1. El sistema acaba el cd'u.

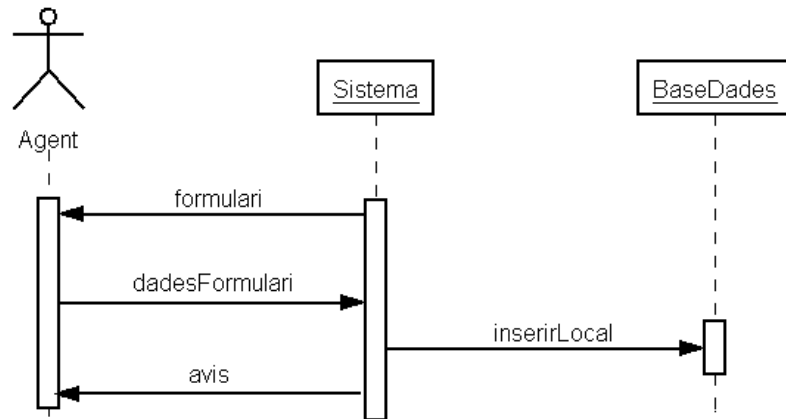
### 1.5.-MODEL CONCEPTUAL



## 1.6.-DIAGRAMES DE SEQUÈNCIA

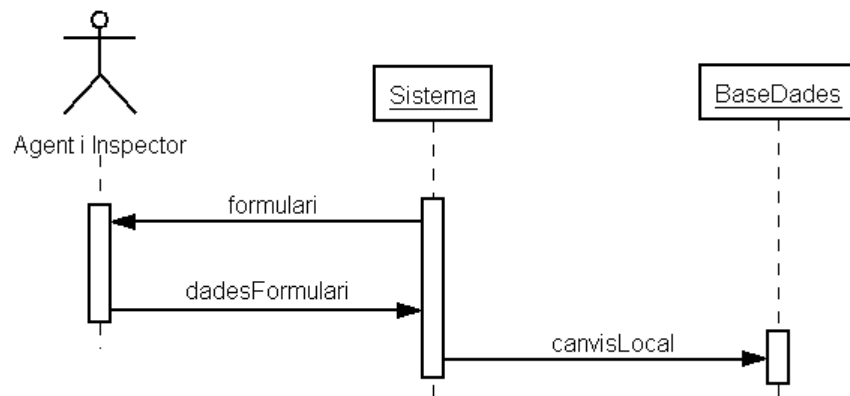
### Cd'u 1. Alta Local

#### Cd'u Alta Local



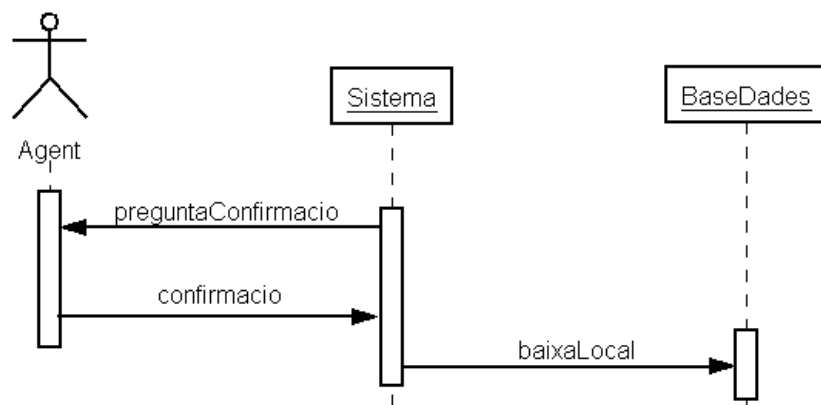
### Cd'u 2. Modificar Local

#### Cd'u Modificar Local



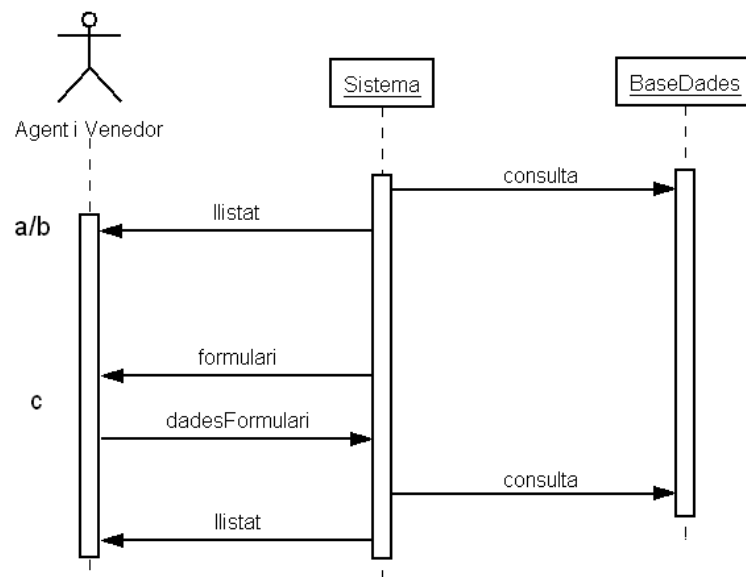
### Cd'u 3. Baixa Local

#### Cd'u Baixa Local



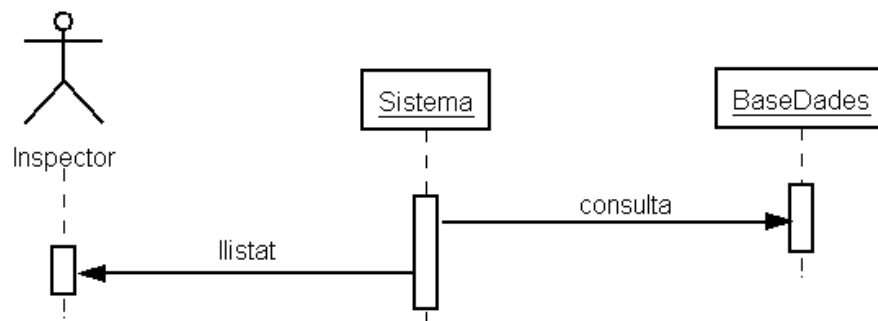
#### Cd'u 4. Cercar Locals

### Cd'u Cercar Locals



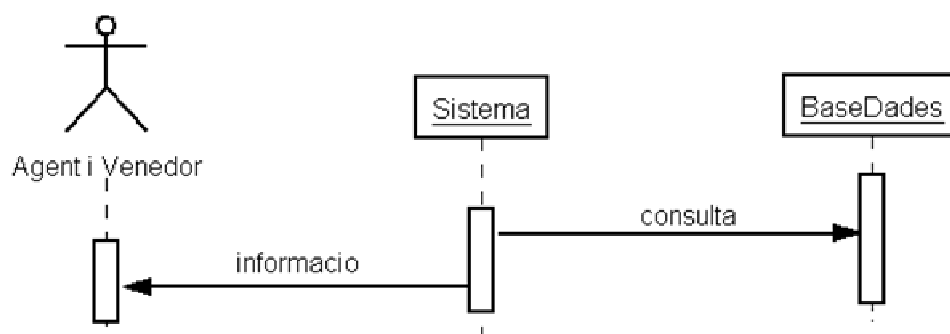
#### Cd'u 5. Llistar Locals Pendants

### Cd'u Llistar Locals Pendants



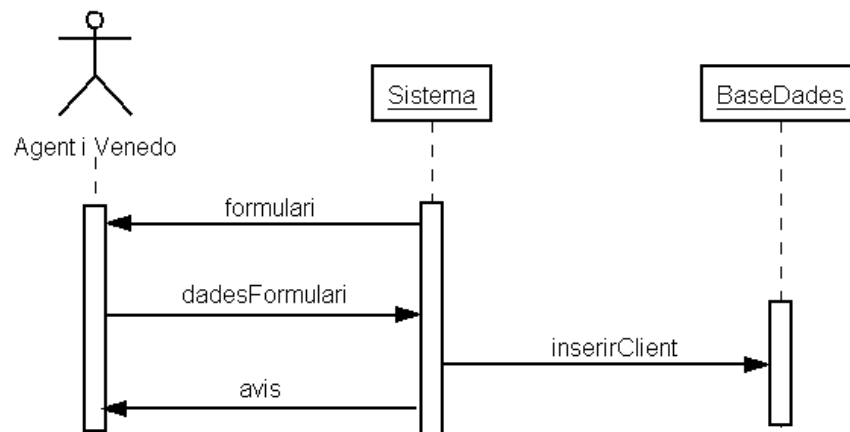
#### Cd'u 6. Veure Informació Local

### Cd'u Veure Informacio Local



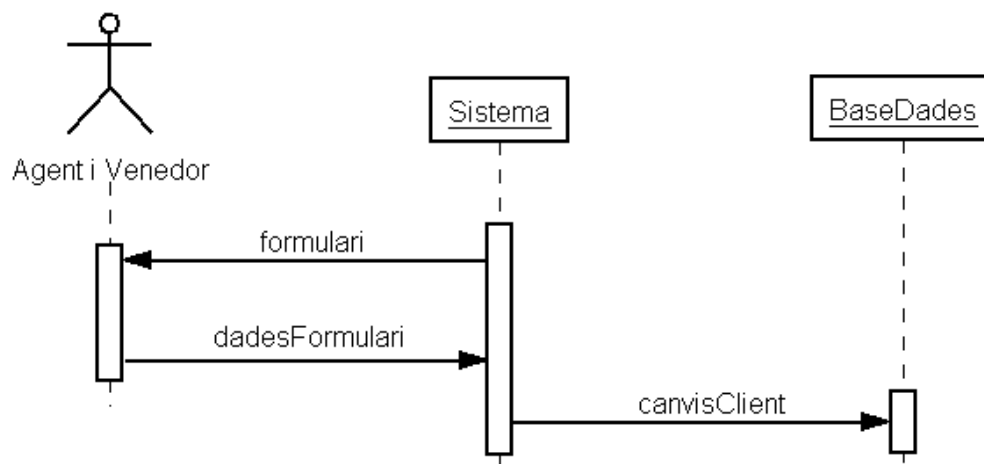
## Cd'u 7. Alta Client

### Cd'u Alta Client



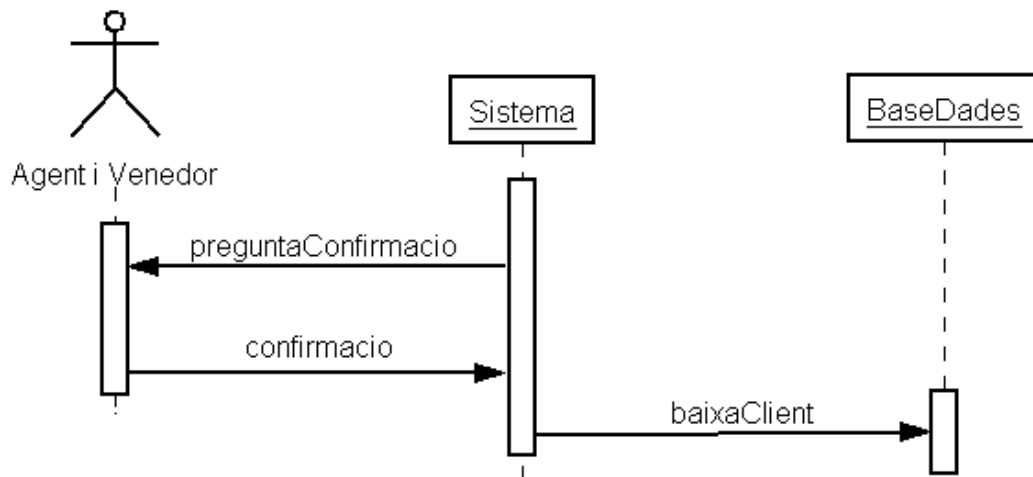
## Cd'u 8. Modificar Client

### Cd'u Modificar Client



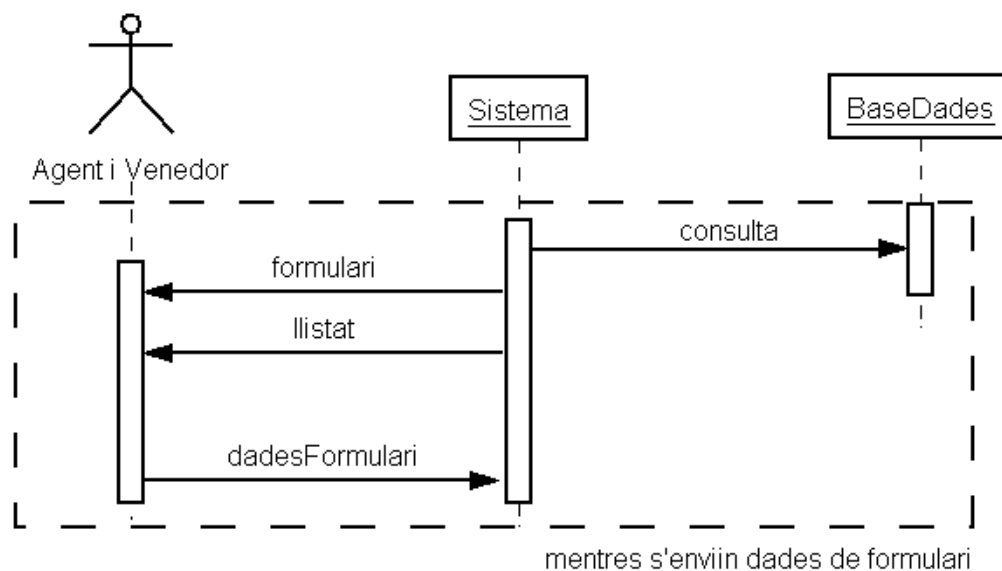
## Cd'u 9. Baixa Client

# Cd'u Baixa Client



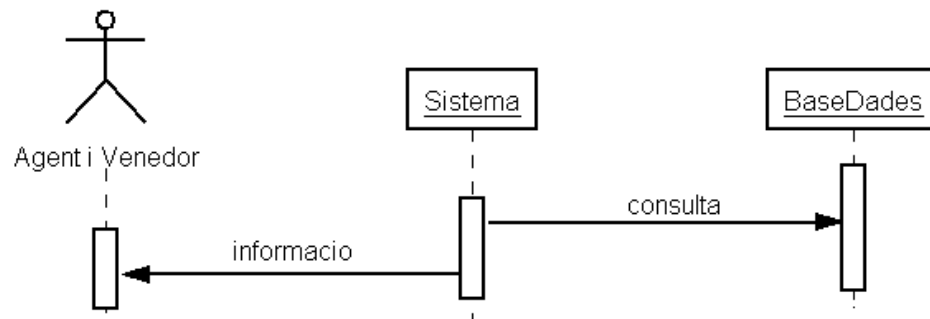
## Cd'u 10. Cercar Clients

# Cd'u Cercar Clients



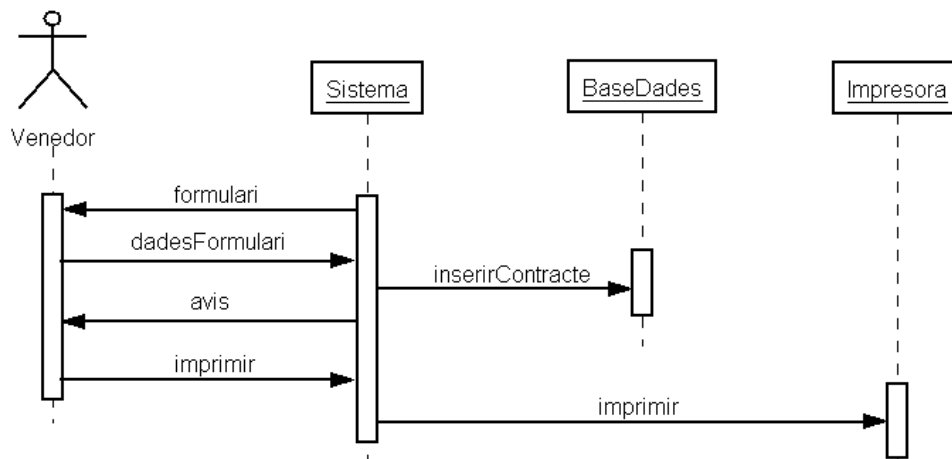
## Cd'u 11. Veure Informacio Client

# Cd'u Veure Informacio Client



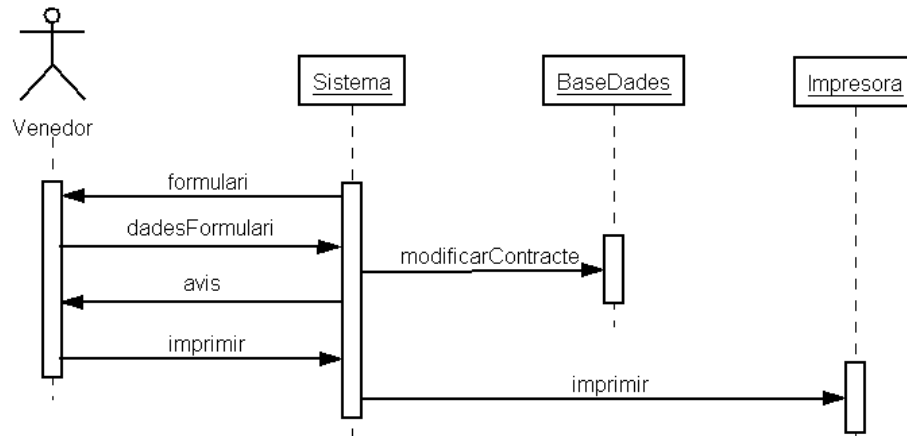
## Cd'u 12. Crear Contracte

# Cd'u Crear Contracte



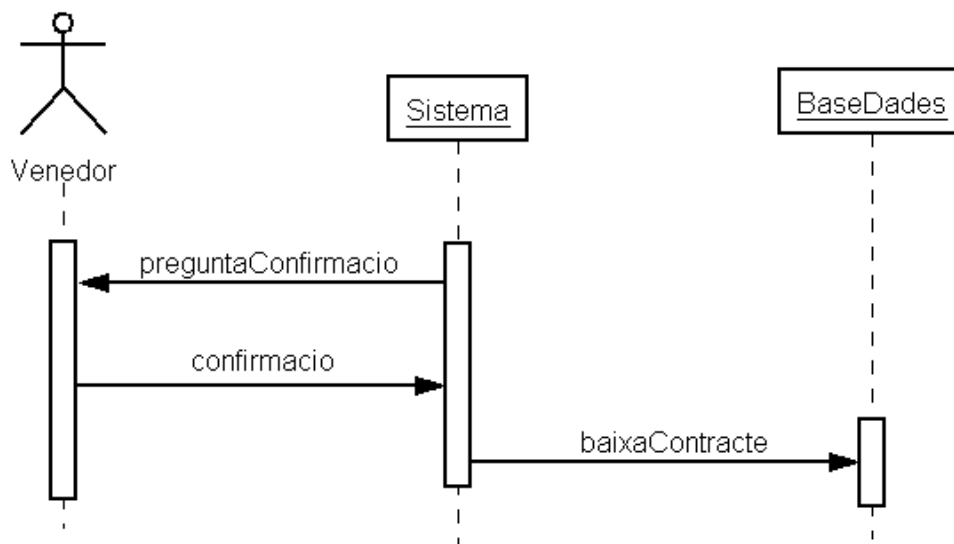
### Cd'u 13. Modificar Contracte

## Cd'u Modificar Contracte

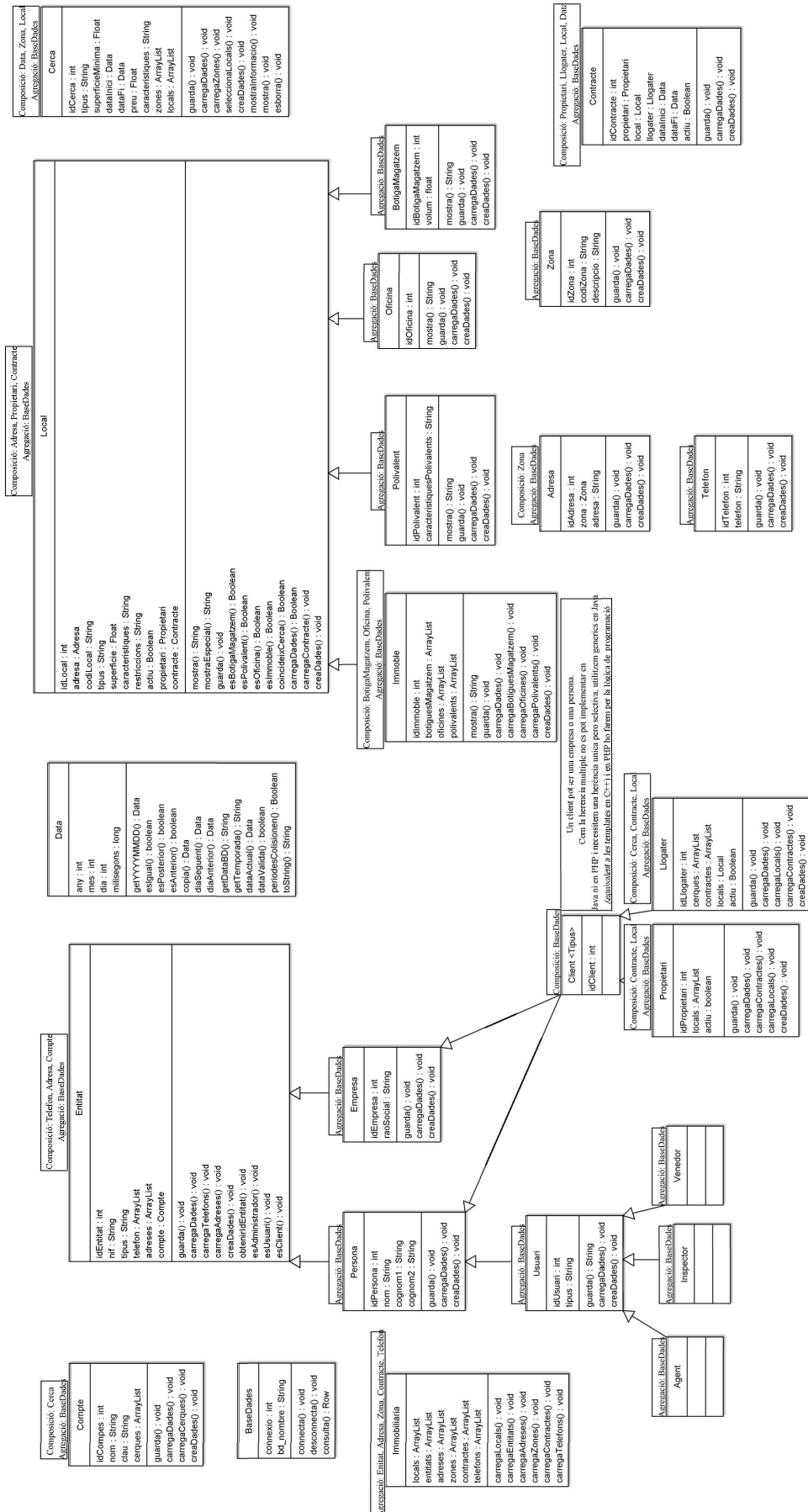


### Cd'u 14. Baixa Contracte

## Cd'u Baixa Contracte



### 1.5.-DIAGRAMA DE CLASES I RELACIONS



## ***1.6.-ESPECIFICACIO DE REQUERIMENTS***

El client, una immobiliària, en aquest demana un sistema que precisa uns requeriments totalment viables i legals, la qual cosa facilitarà la tasca d'anàlisis i d'implementació.

Per tant, podem deduir de forma clara i segura que, havent vist ja els requeriments del professorat, les eines que anem a utilitzar per a la implementació del software son les següents:

Una web externa d'accés públic: els llenguatges i tecnologies seran els més utilitzats i conformats per el consorci de la *World Wide Web (w3c)* i ECMA, és a dir, *HTML*, *CSS* i *JavaScript*. Per la part del servidor utilitzarem PHP, llenguatge ampliament conegut i reconegut i a més lliure i gratuït, lo qual redunda en molta informació i suport disponible gràcies a la comunitat d'Internet. Aquest llenguatge serà l'encarregat de comunicar-se amb el sistema gestor de base de dades.

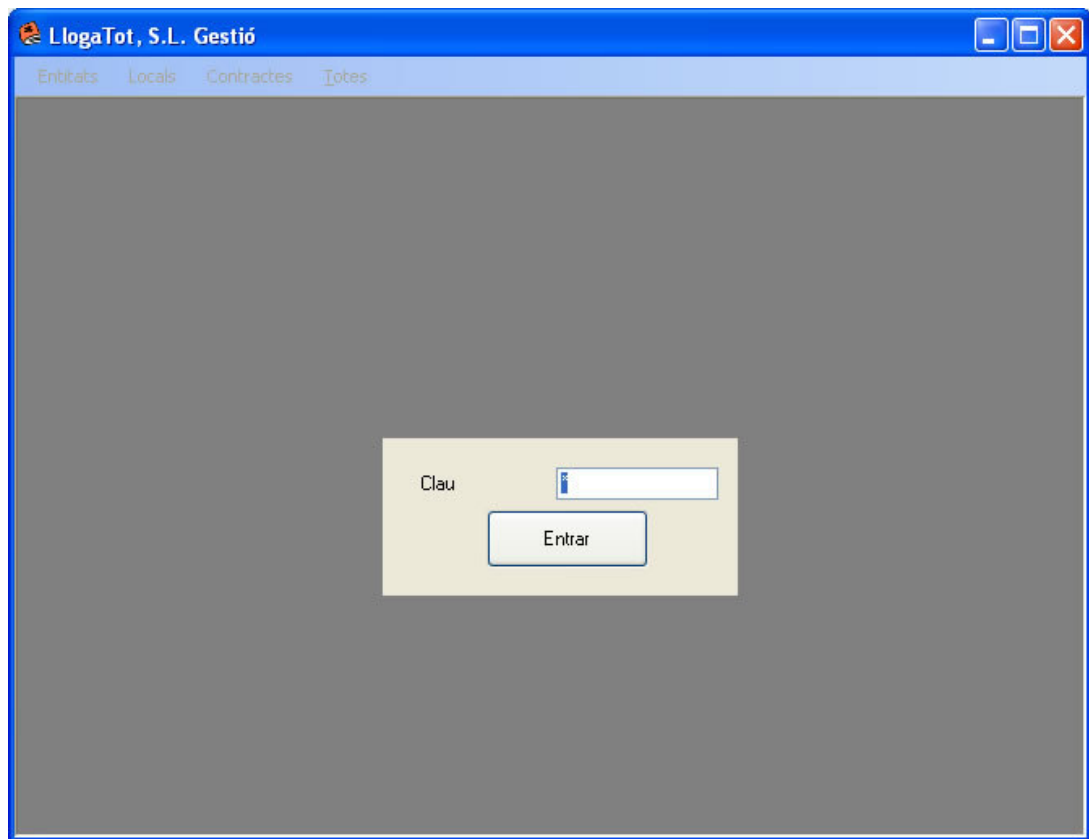
El sistema de base de dades utilitzat serà *mySQL*, sistema també lliure i gratuït i amb una gran acceptació en *la comunitat*. El motor de base de dades que farem servir en *mySQL* serà el motor *InnoDB*. La decisió d'aquest motor ve condicionada per la necessitat d'incloure taules relacionals, amb claus foranies i amb sistema de restriccions i accions en cascada.

Per facilitar la visualització de l'estat de la base de dades i administrar-la farem servir un programa desenvolupat internament amb el llenguatge *C#* (plataforma *.NET 2.0*) utilitzant *Microsoft Visual C# 2005 Express Edition*. Aquest programa tindrà un control d'accés per tal de permetre l'entrada tan sols de l'administrador.

Per a l'administració del sistema, amb un cert nivell d'abstracció i per als treballadors que no vulguin fer simplement una administració de la base de dades, farem servir una web interna amb clau d'accés. Cada figura del personal podrà entrar en aquesta web interna i realitzar les accions que el seu rol assumit dins l'empresa li permet. La web serà realitzada mitjançant *Java (Java SE Development Kit 6 Update 3, utilitzant JavaServer Page)*, utilitzant *NetBeans IDE 5.5.1* i amb la plataforma *Jakarta Struts* que facilita el desenvolupament amb el patró de disseny MVC (Model, Vista, Controlador).

Per la elaboració d'aquesta documentació hem fet servir *Microsoft Office Word 2003* i *ArgoUML*.

## 1.7.-INTERFICIE D'USUARI

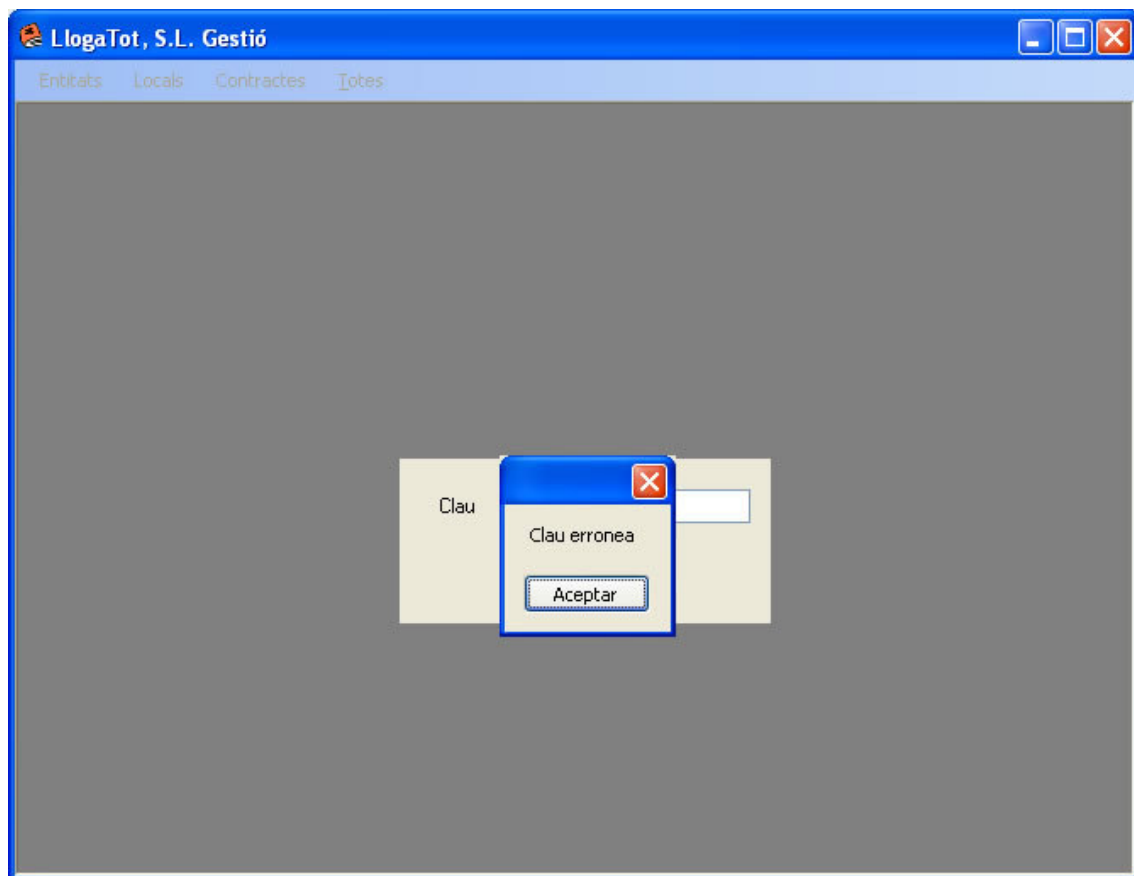


**Finestra:** Login

**Tipus:** Filla

**Modal:** Si

**Descripció:** Demana la paraula clan de l'administrador per permetre l'accés al programa

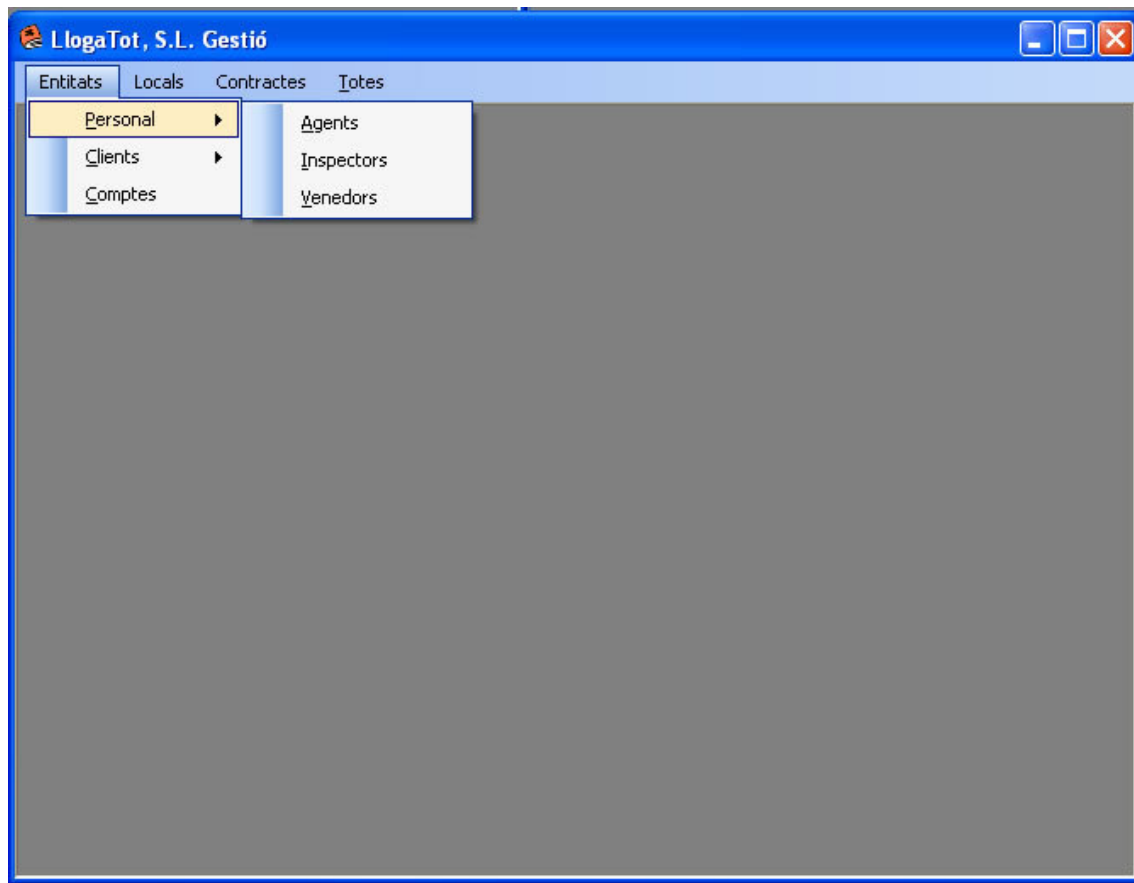


**Finestra:** Login Erroni

**Tipus:** Filla

**Modal:** Si

**Descripció:** La paraula introduïda no es correcta

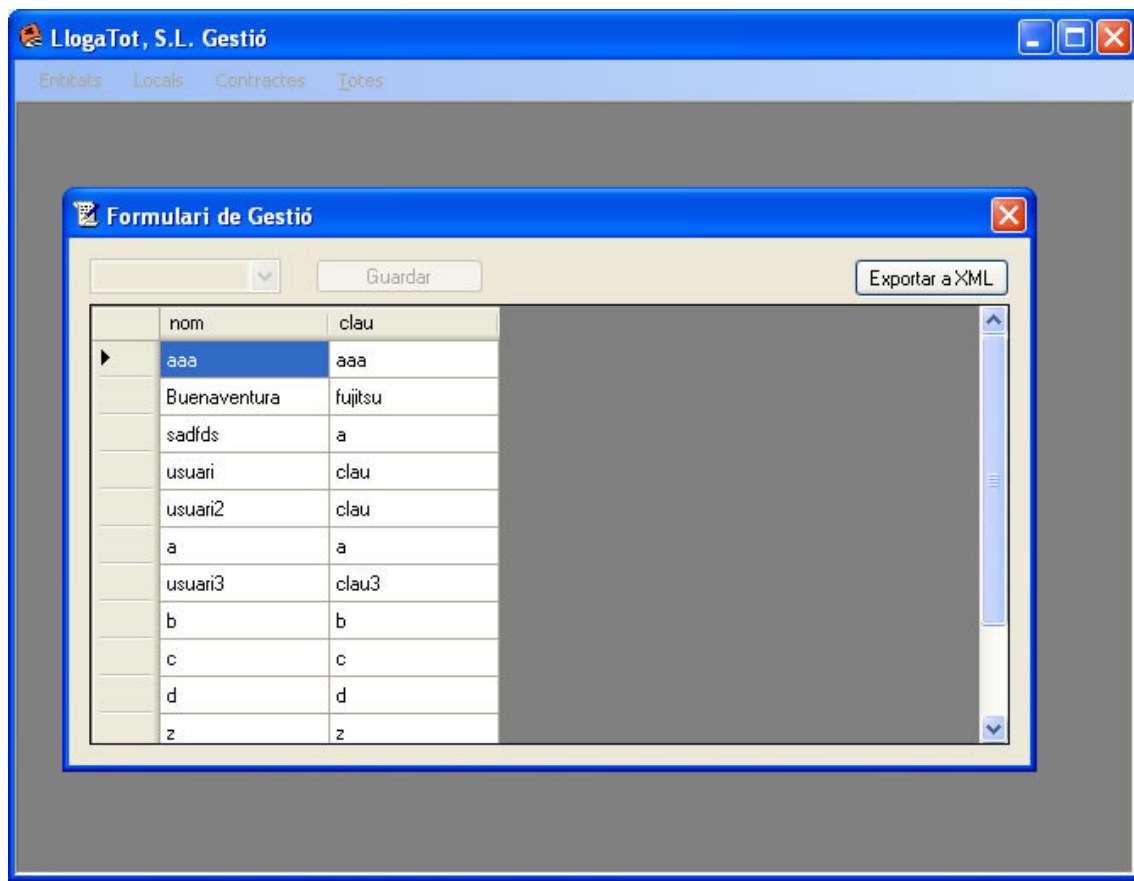


**Finestra:** Principal

**Tipus:** Pare (MDI)

**Modal:** No

**Descripció:** L'usuari administrador pot escollir la opció al menú

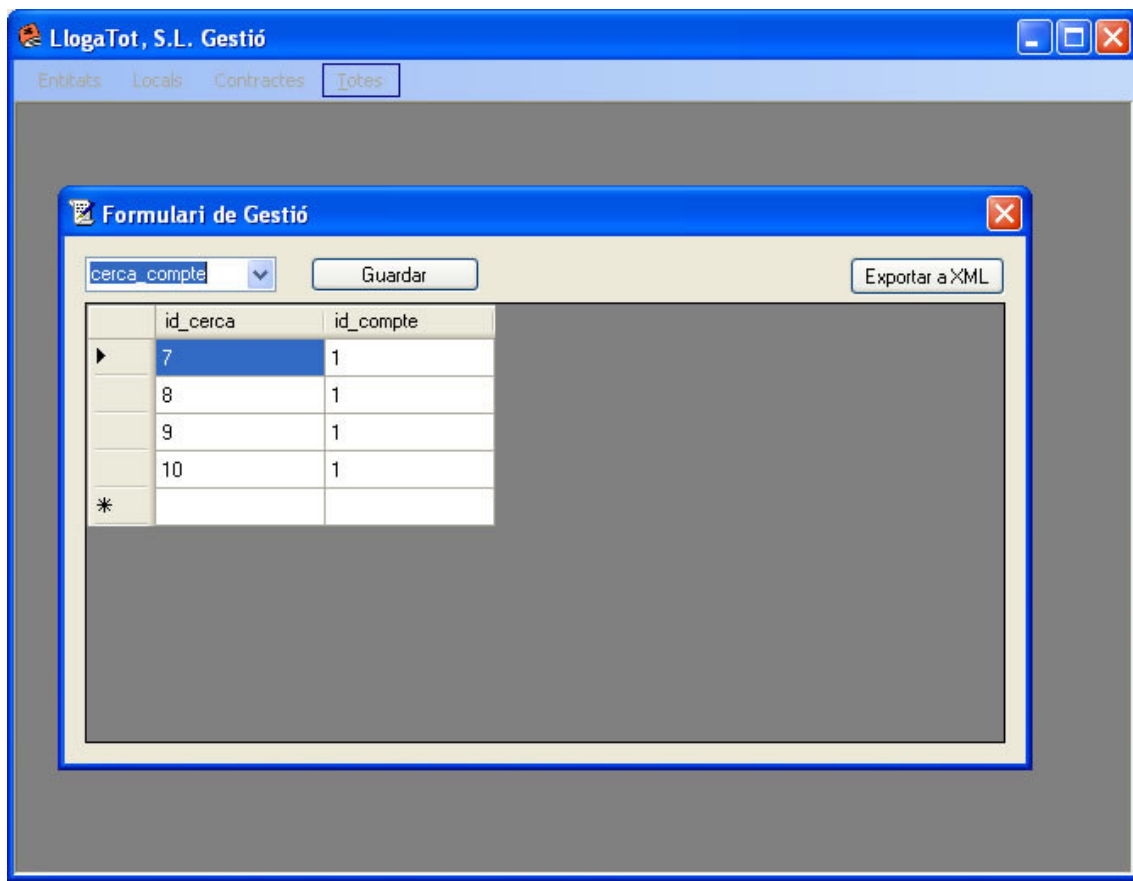


**Finestra:** Formulari de Gestió

**Tipus:** Filla

**Modal:** Si

**Descripció:** L'administrador pot visualitzar el contingut de la base de dades d'una opció específica.

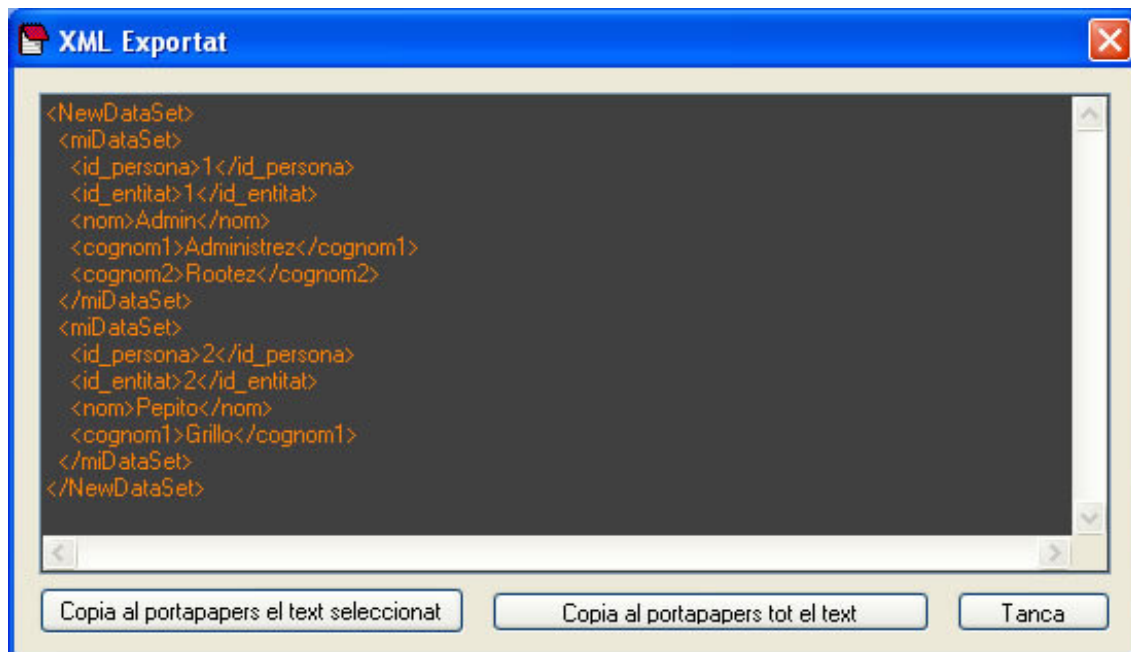


**Finestra:** Formulari de Gestió

**Tipus:** Filla

**Modal:** Si

**Descripció:** L'administrador pot visualitzar i administrar el contingut de la base de dades d'una taula especifica escollida.



**Finestra:** XML Exportat

**Tipus:** Independent

**Modal:** No

**Descripció:** Exporta la taula seleccionada en format XML per poder-la inserir en qualsevol web



**Descripció:** Pantalla principal de la web amb cap usuari validat.



**Descripció:** Pantalla amb un usuari normal validat.



**Descripció:** Pantalla amb un usuari treballador validat (no administrador).



**Descripció:** Pantalla amb un usuari administrador validat.

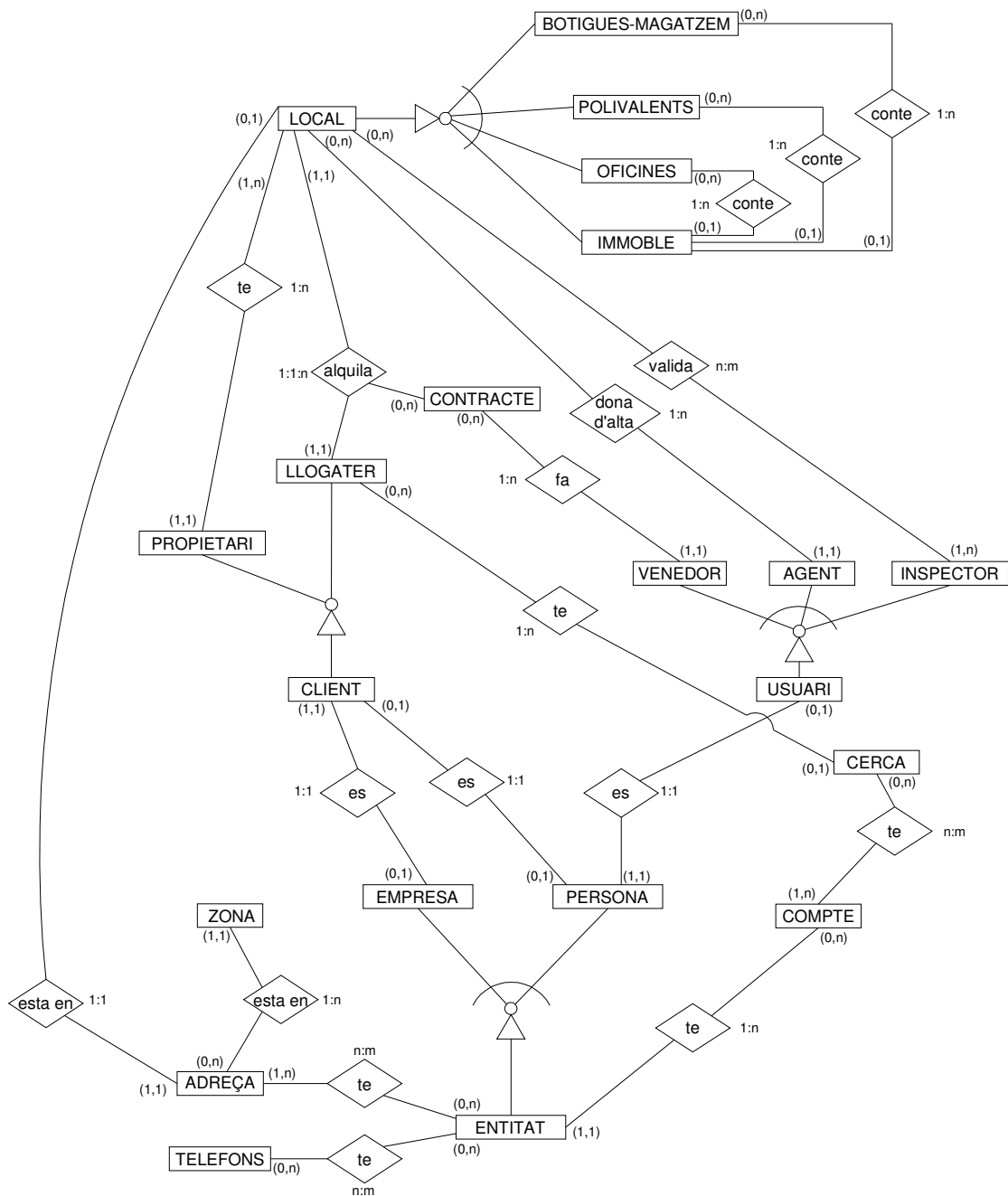
## 2.- DOCUMENTACIÓ DE LA BASE DE DADES

### 2.1.-INTRODUCCIÓ

A continuació s'inclou tota la documentació relativa a la base de dades.

Com a recomanació, aconsellem que es llegeixi primer l'últim apartat, anomenat *Explicació i aclariments*, abans de començar a fer una avaluació de la base de dades. En aquest apartat es podran trobar molts aclariments que justifiquen el perquè de totes les decisions preses en els apartats anteriors.

### 2.2.-DIAGRAMA E-R



### 2.3.-TAULES

LOCALS(id\_local, id\_propietari, id\_adresa, codi\_local, tipus, superficie, caracteristiques, restriccions, preu, actiu)

BOTIGUES\_MAGATZEM(id\_botiga\_magatzem, id\_local, volum)

OFICINES(id\_oficina, id\_local)

POLIVALENTS(id\_polivalent, id\_local, caracteristiques\_polivalent)

IMMOBLES(id\_inmoble, id\_local)

IMMOBLE\_BOTIGA\_MAGATZEM(id\_inmoble, id\_botiga\_magatzem)

IMMOBLE\_OFICINA(id\_inmoble, id\_oficina)

IMMOBLE\_POLIVALENT(id\_inmoble, id\_polivalent)

ZONES(id\_zona, codi\_zona, descripcio)

ENTITATS(id\_entitat, nif, tipus)

PERSONES(id\_persona, id\_entitat, nom, cognom1, cognom2)

EMPRESSES(id\_empresa, id\_entitat, rao\_social)

CLIENTS(id\_client, id\_entitat)

USUARIS(id\_usuari, id\_persona, tipus)

PROPIETARIS(id\_propietari, id\_client, actiu)

LLOGATERS(id\_llogater, id\_client, id\_preferencies, actiu)

TELEFONS(id\_telefon, telefon)

TELEFON\_ENTITAT(id\_telefon, id\_entitat)

ADRESES(id\_adresa, id\_zona, adresa)

ADRESA\_ENTITAT(id\_adresa, id\_entitat)

COMPTE(id\_compte, id\_entitat, nom, clau)

CERQUES(id\_cerca, superficie\_minima, data\_inici, data\_fi, preu, caracteristiques, tipus)

CERCA\_COMPTE(id\_cerca, id\_compte)

CERCA\_ZONA(id\_cerca, id\_zona)

CONTRACTES(id\_contracte, id\_propietari, id\_local, id\_llogater, data\_inici, data\_fi, actiu)

## **2.4.-MODEL CONCEPTUAL**

### LOCALS

```
(  
    id_local INT AUTO_INCREMENT PRIMARY KEY,  
    id_propietari INT NOT NULL,  
    id_adresa INT UNIQUE NOT NULL,  
    codi_local VARCHAR (10) UNIQUE NOT NULL,  
    tipus ENUM ('botiga_magatzem', 'oficina', 'polivalent', 'immoble') NOT NULL,  
    superficie DECIMAL (10, 2),  
    caracteristiques TINYTEXT,  
    restriccions TINYTEXT,  
    preu DECIMAL (10, 2),  
    actiu BOOLEAN,  
    FOREIGN KEY (id_propietari) REFERENCES propietaris (id_propietari) ON  
DELETE CASCADE ON UPDATE CASCADE,  
    FOREIGN KEY (id_adresa) REFERENCES adreeses (id_adresa) ON DELETE  
CASCADE ON UPDATE CASCADE  
);
```

### BOTIGUES\_MAGATZEM

```
(  
    id_botiga_magatzem INT AUTO_INCREMENT PRIMARY KEY,  
    id_local INT NOT NULL,  
    volum DECIMAL (10, 2),  
    FOREIGN KEY (id_local) REFERENCES locals (id_local) ON DELETE  
CASCADE ON UPDATE CASCADE  
);
```

### OFICINES

```
(  
    id_oficina INT AUTO_INCREMENT PRIMARY KEY,  
    id_local INT NOT NULL,  
    FOREIGN KEY (id_local) REFERENCES locals (id_local) ON DELETE  
CASCADE ON UPDATE CASCADE  
);
```

### POLIVALENTS

```
(  
    id_polivalent INT AUTO_INCREMENT PRIMARY KEY,  
    id_local INT NOT NULL,  
    caracteristiques_polivalent TINYTEXT,  
    FOREIGN KEY (id_local) REFERENCES locals (id_local) ON DELETE  
CASCADE ON UPDATE CASCADE  
);
```

## IMMOBLES

```
(  
    id_inmoble INT AUTO_INCREMENT PRIMARY KEY,  
    id_local INT NOT NULL,  
    FOREIGN KEY (id_local) REFERENCES locals (id_local) ON DELETE  
    CASCADE ON UPDATE CASCADE  
);
```

## IMMOBLE\_BOTIGA\_MAGATZEM

```
(  
    id_inmoble INT NOT NULL,  
    id_botiga_magatzem INT NOT NULL,  
    FOREIGN KEY (id_inmoble) REFERENCES immobles (id_inmoble) ON  
    DELETE CASCADE ON UPDATE CASCADE,  
    FOREIGN KEY (id_botiga_magatzem) REFERENCES botigues_magatzem  
    (id_botiga_magatzem) ON DELETE CASCADE ON UPDATE CASCADE,  
    PRIMARY KEY (id_inmoble, id_botiga_magatzem) ON DELETE CASCADE  
    ON UPDATE CASCADE  
);
```

## IMMOBLE\_OFICINA

```
(  
    id_inmoble INT NOT NULL,  
    id_oficina INT NOT NULL,  
    FOREIGN KEY (id_inmoble) REFERENCES immobles (id_inmoble) ON  
    DELETE CASCADE ON UPDATE CASCADE,  
    FOREIGN KEY (id_oficina) REFERENCES oficines (id_oficina) ON DELETE  
    CASCADE ON UPDATE CASCADE,  
    PRIMARY KEY (id_inmoble, id_oficina)  
);
```

## IMMOBLE\_POLIVALENT

```
(  
    id_inmoble INT NOT NULL,  
    id_polivalent INT NOT NULL,  
    FOREIGN KEY (id_inmoble) REFERENCES immobles (id_inmoble) ON  
    DELETE CASCADE ON UPDATE CASCADE,  
    FOREIGN KEY (id_polivalent) REFERENCES polivalents (id_polivalent) ON  
    DELETE CASCADE ON UPDATE CASCADE,  
    PRIMARY KEY (id_inmoble, id_polivalent)  
);
```

## ZONES

```
(  
    id_zona INT AUTO_INCREMENT PRIMARY KEY,  
    codi_zona VARCHAR (4) UNIQUE NOT NULL,  
    descripcio VARCHAR (255) DEFAULT ''  
);
```

## ENTITATS

```
(  
    id_entitat INT AUTO_INCREMENT PRIMARY KEY,  
    nif VARCHAR (10) UNIQUE NOT NULL,  
    tipus ENUM ('persona', 'empresa') NOT NULL  
);
```

## PERSONES

```
(  
    id_persona INT AUTO_INCREMENT PRIMARY KEY,  
    id_entitat INT NOT NULL,  
    nom VARCHAR (125) NOT NULL,  
    cognom1 VARCHAR (125) NOT NULL,  
    cognom2 VARCHAR (125),  
    FOREIGN KEY (id_entitat) REFERENCES entitats (id_entitat) ON DELETE  
    CASCADE ON UPDATE CASCADE  
);
```

## EMPRESSES

```
(  
    id_empresa INT AUTO_INCREMENT PRIMARY KEY,  
    id_entitat INT NOT NULL,  
    rao_social VARCHAR (125) UNIQUE NOT NULL,  
    FOREIGN KEY (id_entitat) REFERENCES entitats (id_entitat) ON DELETE  
    CASCADE ON UPDATE CASCADE  
);
```

## CLIENTS

```
(  
    id_client INT AUTO_INCREMENT PRIMARY KEY,  
    id_entitat INT NOT NULL,  
    FOREIGN KEY (id_entitat) REFERENCES entitats (id_entitat) ON DELETE  
    CASCADE ON UPDATE CASCADE  
);
```

## USUARIS

```
(  
    id_usuari INT AUTO_INCREMENT PRIMARY KEY,  
    id_persona INT NOT NULL,  
    tipus ENUM ('agent', 'inspector', 'venedor') NOT NULL,  
    FOREIGN KEY (id_persona) REFERENCES persones (id_persona) ON DELETE  
    CASCADE ON UPDATE CASCADE  
);
```

## PROPIETARIS

```
(  
    id_propietari INT AUTO_INCREMENT PRIMARY KEY,  
    id_client INT NOT NULL,  
    actiu BOOLEAN,
```

```

        FOREIGN KEY (id_client) REFERENCES clients (id_client) ON DELETE
        CASCADE ON UPDATE CASCADE
    );

```

#### LLOGATERS

```

(
    id_llogater INT AUTO_INCREMENT PRIMARY KEY,
    id_client INT NOT NULL,
    id_preferencies INT NOT NULL,
    actiu BOOLEAN,
    FOREIGN KEY (id_client) REFERENCES clients (id_client) ON DELETE
    CASCADE ON UPDATE CASCADE,
    FOREIGN KEY (id_preferencies) REFERENCES cerques (id_cerca) ON
    DELETE CASCADE ON UPDATE CASCADE
);

```

#### TELEFONS

```

(
    id_telefon INT AUTO_INCREMENT PRIMARY KEY,
    telefon VARCHAR (25) UNIQUE NOT NULL
);

```

#### TELEFON\_ENTITAT

```

(
    id_telefon INT NOT NULL,
    id_entitat INT NOT NULL,
    FOREIGN KEY (id_telefon) REFERENCES telefons (id_telefon) ON DELETE
    CASCADE ON UPDATE CASCADE,
    FOREIGN KEY (id_entitat) REFERENCES entitats (id_entitat) ON DELETE
    CASCADE ON UPDATE CASCADE,
    PRIMARY KEY (id_telefon, id_entitat)
);

```

#### ADRESES

```

(
    id_adresa INT AUTO_INCREMENT PRIMARY KEY,
    id_zona INT NOT NULL,
    adresa VARCHAR (255) NOT NULL,
    FOREIGN KEY (id_zona) REFERENCES zones (id_zona) ON DELETE
    CASCADE ON UPDATE CASCADE
);

```

#### ADRESA\_ENTITAT

```

(
    id_adresa INT NOT NULL,
    id_entitat INT NOT NULL,
    FOREIGN KEY (id_adresa) REFERENCES adrees (id_adresa) ON DELETE
    CASCADE ON UPDATE CASCADE,
    FOREIGN KEY (id_entitat) REFERENCES entitats (id_entitat) ON DELETE
    CASCADE ON UPDATE CASCADE,

```

```

        PRIMARY KEY (id_adresa, id_entitat)
    );

COMPTES
(
    id_compte INT AUTO_INCREMENT PRIMARY KEY,
    id_entitat INT NOT NULL,
    nom VARCHAR (20) UNIQUE NOT NULL,
    clau VARCHAR (20) NOT NULL,
    FOREIGN KEY (id_entitat) REFERENCES entitats (id_entitat) ON DELETE
    CASCADE ON UPDATE CASCADE
);

CERQUES
(
    id_cerca INT AUTO_INCREMENT PRIMARY KEY,
    superficie_minima DECIMAL (10, 2) NOT NULL,
    data_inici DATE,
    data_fi DATE,
    preu DECIMAL (10, 2),
    caracteristiques TINYTEXT,
    tipus ENUM ('tots', 'botiga_magatzem', 'oficina', 'polivalent', 'immoble') NOT
    NULL
);

CERCA_COMPTE
(
    id_cerca INT NOT NULL,
    id_compte INT NOT NULL,
    FOREIGN KEY (id_cerca) REFERENCES cerques (id_cerca) ON DELETE
    CASCADE ON UPDATE CASCADE,
    FOREIGN KEY (id_compte) REFERENCES comptes (id_compte) ON DELETE
    CASCADE ON UPDATE CASCADE,
    PRIMARY KEY (id_cerca, id_compte)
);

CERCA_ZONA
(
    id_cerca INT NOT NULL,
    id_zona INT NOT NULL,
    FOREIGN KEY (id_cerca) REFERENCES cerques (id_cerca) ON DELETE
    CASCADE ON UPDATE CASCADE,
    FOREIGN KEY (id_zona) REFERENCES zones (id_zona) ON DELETE
    CASCADE ON UPDATE CASCADE,
    PRIMARY KEY (id_cerca, id_zona)
);

CONTRACTES
(
    id_contracte INT AUTO_INCREMENT PRIMARY KEY,

```

```

    id_propietari INT NOT NULL,
    id_local INT NOT NULL,
    id_llogater INT NOT NULL,
    data_inici DATE NOT NULL,
    data_fi DATE,
    actiu BOOLEAN,
    FOREIGN KEY (id_propietari) REFERENCES propietaris (id_propietari) ON
DELETE CASCADE ON UPDATE CASCADE,
    FOREIGN KEY (id_local) REFERENCES locals (id_local) ON DELETE
CASCADE ON UPDATE CASCADE,
    FOREIGN KEY (id_llogater) REFERENCES llogater (id_llogater) ON DELETE
CASCADE ON UPDATE CASCADE
);

```

## 2.5.-TRANSCRIPCIÓ AL SGBD (mySQL)

```

CREATE TABLE IF NOT EXISTS LOCALS
(
    id_local INT AUTO_INCREMENT PRIMARY KEY,
    id_propietari INT NOT NULL,
    id_adresa INT UNIQUE NOT NULL,
    codi_local VARCHAR (10) UNIQUE NOT NULL,
    tipus ENUM ('botiga_magatzem', 'oficina', 'polivalent', 'immoble') NOT NULL,
    superficie DECIMAL (10, 2),
    caracteristiques TINYTEXT,
    restriccions TINYTEXT,
    preu DECIMAL (10, 2),
    actiu BOOLEAN,
    FOREIGN KEY (id_propietari) REFERENCES propietaris (id_propietari) ON DELETE
CASCADE ON UPDATE CASCADE,
    FOREIGN KEY (id_adresa) REFERENCES adresses (id_adresa) ON DELETE CASCADE ON
UPDATE CASCADE
) TYPE = INNODB;

CREATE TABLE IF NOT EXISTS BOTIGUES_MAGATZEM
(
    id_botiga_magatzem INT AUTO_INCREMENT PRIMARY KEY,
    id_local INT NOT NULL,
    volum DECIMAL (10, 2),
    FOREIGN KEY (id_local) REFERENCES locals (id_local) ON DELETE CASCADE ON UPDATE
CASCADE
) TYPE = INNODB;

CREATE TABLE IF NOT EXISTS OFICINES
(
    id_oficina INT AUTO_INCREMENT PRIMARY KEY,
    id_local INT NOT NULL,
    FOREIGN KEY (id_local) REFERENCES locals (id_local) ON DELETE CASCADE ON UPDATE
CASCADE
) TYPE = INNODB;

CREATE TABLE IF NOT EXISTS POLIVALENTS
(
    id_polivalent INT AUTO_INCREMENT PRIMARY KEY,
    id_local INT NOT NULL,
    caracteristiques_polivalent TINYTEXT,
    FOREIGN KEY (id_local) REFERENCES locals (id_local) ON DELETE CASCADE ON UPDATE
CASCADE
) TYPE = INNODB;

CREATE TABLE IF NOT EXISTS IMMOBLES

```

```

(
id_immoble INT AUTO_INCREMENT PRIMARY KEY,
id_local INT NOT NULL,
FOREIGN KEY (id_local) REFERENCES locals (id_local) ON DELETE CASCADE ON UPDATE
CASCADE
) TYPE = INNODB;

CREATE TABLE IF NOT EXISTS IMMOBLE_BOTIGA_MAGATZEM
(
id_immoble INT NOT NULL,
id_botiga_magatzem INT NOT NULL,
FOREIGN KEY (id_immoble) REFERENCES immobles (id_immoble) ON DELETE CASCADE ON
UPDATE CASCADE,
FOREIGN KEY (id_botiga_magatzem) REFERENCES botigues_magatzem
(id_botiga_magatzem) ON DELETE CASCADE ON UPDATE CASCADE,
PRIMARY KEY (id_immoble, id_botiga_magatzem)
) TYPE = INNODB;

CREATE TABLE IF NOT EXISTS IMMOBLE_OFICINA
(
id_immoble INT NOT NULL,
id_oficina INT NOT NULL,
FOREIGN KEY (id_immoble) REFERENCES immobles (id_immoble) ON DELETE CASCADE ON
UPDATE CASCADE,
FOREIGN KEY (id_oficina) REFERENCES oficines (id_oficina) ON DELETE CASCADE ON
UPDATE CASCADE,
PRIMARY KEY (id_immoble, id_oficina)
) TYPE = INNODB;

CREATE TABLE IF NOT EXISTS IMMOBLE_POLIVALENT
(
id_immoble INT NOT NULL,
id_polivalent INT NOT NULL,
FOREIGN KEY (id_immoble) REFERENCES immobles (id_immoble) ON DELETE CASCADE ON
UPDATE CASCADE,
FOREIGN KEY (id_polivalent) REFERENCES polivalents (id_polivalent) ON DELETE
CASCADE ON UPDATE CASCADE,
PRIMARY KEY (id_immoble, id_polivalent)
) TYPE = INNODB;

CREATE TABLE IF NOT EXISTS ZONES
(
id_zona INT AUTO_INCREMENT PRIMARY KEY,
codi_zona VARCHAR (4) UNIQUE NOT NULL,
descripcio VARCHAR (255) DEFAULT ''
) TYPE = INNODB;

CREATE TABLE IF NOT EXISTS ENTITATS
(
id_entitat INT AUTO_INCREMENT PRIMARY KEY,
nif VARCHAR (10) UNIQUE NOT NULL,
tipus ENUM ('persona', 'empresa') NOT NULL
) TYPE = INNODB;

CREATE TABLE IF NOT EXISTS PERSONES
(
id_persona INT AUTO_INCREMENT PRIMARY KEY,
id_entitat INT NOT NULL,
nom VARCHAR (125) NOT NULL,
cognom1 VARCHAR (125) NOT NULL,
cognom2 VARCHAR (125),
FOREIGN KEY (id_entitat) REFERENCES entitats (id_entitat) ON DELETE CASCADE ON
UPDATE CASCADE
) TYPE = INNODB;

CREATE TABLE IF NOT EXISTS EMPRESES
(

```

```

id_empresa INT AUTO_INCREMENT PRIMARY KEY,
id_entitat INT NOT NULL,
rao_social VARCHAR (125) UNIQUE NOT NULL,
FOREIGN KEY (id_entitat) REFERENCES entitats (id_entitat) ON DELETE CASCADE ON
UPDATE CASCADE
) TYPE = INNODB;

CREATE TABLE IF NOT EXISTS CLIENTS
(
id_client INT AUTO_INCREMENT PRIMARY KEY,
id_entitat INT NOT NULL,
FOREIGN KEY (id_entitat) REFERENCES entitats (id_entitat) ON DELETE CASCADE ON
UPDATE CASCADE
) TYPE = INNODB;

CREATE TABLE IF NOT EXISTS USUARIS
(
id_usuari INT AUTO_INCREMENT PRIMARY KEY,
id_persona INT NOT NULL,
tipus ENUM ('agent', 'inspector', 'venedor') NOT NULL,
FOREIGN KEY (id_persona) REFERENCES persones (id_persona) ON DELETE CASCADE ON
UPDATE CASCADE
) TYPE = INNODB;

CREATE TABLE IF NOT EXISTS PROPIETARIS
(
id_propietari INT AUTO_INCREMENT PRIMARY KEY,
id_client INT NOT NULL,
actiu BOOLEAN,
FOREIGN KEY (id_client) REFERENCES clients (id_client) ON DELETE CASCADE ON
UPDATE CASCADE
) TYPE = INNODB;

CREATE TABLE IF NOT EXISTS LLOGATERS
(
id_llogater INT AUTO_INCREMENT PRIMARY KEY,
id_client INT NOT NULL,
id_preferencies INT NOT NULL,
actiu BOOLEAN,
FOREIGN KEY (id_client) REFERENCES clients (id_client) ON DELETE CASCADE ON
UPDATE CASCADE,
FOREIGN KEY (id_preferencies) REFERENCES cerques (id_cerca) ON DELETE CASCADE ON
UPDATE CASCADE
) TYPE = INNODB;

CREATE TABLE IF NOT EXISTS TELEFONS
(
id_telefon INT AUTO_INCREMENT PRIMARY KEY,
telefon VARCHAR (25) UNIQUE NOT NULL
) TYPE = INNODB;

CREATE TABLE IF NOT EXISTS TELEFON_ENTITAT
(
id_telefon INT NOT NULL,
id_entitat INT NOT NULL,
FOREIGN KEY (id_telefon) REFERENCES telefons (id_telefon) ON DELETE CASCADE ON
UPDATE CASCADE,
FOREIGN KEY (id_entitat) REFERENCES entitats (id_entitat) ON DELETE CASCADE ON
UPDATE CASCADE,
PRIMARY KEY (id_telefon, id_entitat)
) TYPE = INNODB;

CREATE TABLE IF NOT EXISTS ADRESES
(
id_adresa INT AUTO_INCREMENT PRIMARY KEY,
id_zona INT NOT NULL,
adresa VARCHAR (255) NOT NULL,

```

```

FOREIGN KEY (id_zona) REFERENCES zones (id_zona) ON DELETE CASCADE ON UPDATE
CASCADE
) TYPE = INNODB;

CREATE TABLE IF NOT EXISTS ADRESA_ENTITAT
(
id_adresa INT NOT NULL,
id_entitat INT NOT NULL,
FOREIGN KEY (id_adresa) REFERENCES adresses (id_adresa) ON DELETE CASCADE ON
UPDATE CASCADE,
FOREIGN KEY (id_entitat) REFERENCES entitats (id_entitat) ON DELETE CASCADE ON
UPDATE CASCADE,
PRIMARY KEY (id_adresa, id_entitat)
) TYPE = INNODB;

CREATE TABLE IF NOT EXISTS COMPTE
(
id_compte INT AUTO_INCREMENT PRIMARY KEY,
id_entitat INT NOT NULL,
nom VARCHAR (20) UNIQUE NOT NULL,
clau VARCHAR (20) NOT NULL,
FOREIGN KEY (id_entitat) REFERENCES entitats (id_entitat) ON DELETE CASCADE ON
UPDATE CASCADE
) TYPE = INNODB;

CREATE TABLE IF NOT EXISTS CERQUES
(
id_cerca INT AUTO_INCREMENT PRIMARY KEY,
superficie_minima DECIMAL (10, 2) NOT NULL,
data_inici DATE,
data_fi DATE,
preu DECIMAL (10, 2),
caracteristiques TINYTEXT,
tipus ENUM ('tots', 'botiga_magatzem', 'oficina', 'polivalent', 'immoble') NOT
NULL
) TYPE = INNODB;

CREATE TABLE IF NOT EXISTS CERCA_COMPTE
(
id_cerca INT NOT NULL,
id_compte INT NOT NULL,
FOREIGN KEY (id_cerca) REFERENCES cerques (id_cerca) ON DELETE CASCADE ON UPDATE
CASCADE,
FOREIGN KEY (id_compte) REFERENCES comptes (id_compte) ON DELETE CASCADE ON
UPDATE CASCADE,
PRIMARY KEY (id_cerca, id_compte)
) TYPE = INNODB;

CREATE TABLE IF NOT EXISTS CERCA_ZONA
(
id_cerca INT NOT NULL,
id_zona INT NOT NULL,
FOREIGN KEY (id_cerca) REFERENCES cerques (id_cerca) ON DELETE CASCADE ON UPDATE
CASCADE,
FOREIGN KEY (id_zona) REFERENCES zones (id_zona) ON DELETE CASCADE ON UPDATE
CASCADE,
PRIMARY KEY (id_cerca, id_zona)
) TYPE = INNODB;

CREATE TABLE IF NOT EXISTS CONTRACTES
(
id_contracte INT AUTO_INCREMENT PRIMARY KEY,
id_propietari INT NOT NULL,
id_local INT NOT NULL,
id_llogater INT NOT NULL,
data_inici DATE NOT NULL,
data_fi DATE,
actiu BOOLEAN,

```

```

FOREIGN KEY (id_propietari) REFERENCES propietaris (id_propietari) ON DELETE
CASCADE ON UPDATE CASCADE,
FOREIGN KEY (id_local) REFERENCES locals (id_local) ON DELETE CASCADE ON UPDATE
CASCADE,
FOREIGN KEY (id_llogater) REFERENCES llogater (id_llogater) ON DELETE CASCADE ON
UPDATE CASCADE
) TYPE = INNODB;

```

## 2.6.-EXPLICACIÓ I ACLARIMENTS

A simple vista, la base de dades pot semblar molt extensa però la justificació es que hem cregut que era important que aquesta mantingués la consistència i la integritat de les dades, encara que el software que la utilitzés també ho faci. Així, si aquest últim falla o es crea un de nou amb algun possible error, la base de dades hauria de ser capaç d'evitar en tot lo possible qualsevol incongruència.

Hem cregut convenient utilitzar claus foranies, la qual cosa ens ha obligat a utilitzar el motor InnoDB de MySQL (no present en totes les versions, per desgràcia, però si en les més noves si s'activa), i utilitzar bastant l'especialització. També hem fet us de les accions en cascada, per mantenir la coherència, per estalviar feina al software i per automatitzar tasques que, degut al nombre de taules, haurien estat més complicades.

També hem intentat que la base de dades sigui fàcilment *escalable*, creant-la pensant en possibles modificacions que es podrien exigir en un futur. Com no totes les modificacions futures es poden esbrinar, hem intentat ajustar-nos a les més previsibles i possibles. Evidentment, la previsió de futurs canvis és sempre molt limitada per raons obvies i hem escollit moltes vegades lo que nosaltres creiem més raonable. En un cas de veritat aquestes decisions dependrien de tot un estudi.

Això també ajuda a millorar la *escalabilitat* del software que es basa en la base de dades.

Hem intentat que les entitats de la base de dades siguin prou convenients per a poder facilitar el seu transport als objectes que utilitza el software que manega amb aquesta. Així, moltes vegades hem pogut passar taules a objectes i fins i tot passar especialitzacions a herències o interfícies d'una forma àgil i lògica. Moltes relacions existents en la base de dades s'han traduït en relacions (d'associació o d'agregació, per exemple) en el software final.

Cal recalcar que si la base de dades de tipus MySQL no suporta InnoDB les taules es crearan sense cap error però les claus foranies no funcionaran (tampoc les accions en cascada), perdent així gran part de la funcionalitat i la raó de l'existència de moltes de les taules i camps. Per sort, les últimes versions de MySQL ja suporten aquest motor de fa temps i en la majoria ja ve activat (però si no, activar-lo no suposa molt treball). Es possible que el programari no funcioni bé sense motor InnoDB (sobretot al fer modificacions o esborrar, degut a la manca de suport d'accions en cascada).

Per falta de temps no hem pogut implementar restriccions (*constraint*) que hagueren millorat una mica més el comportament de la base de dades (aquestes solament les fem de moment per software). Però hem intentat que el disseny de la base de dades estigui pensat per facilitar la implementació en un futur d'aquestes restriccions, d'una manera fàcil i sense tenir que fer canvis ni a les taules ni als seus camps.

A continuació passem a comentar les taules que formen la nostra base de dades i la raó d'algunes coses que sense una explicació podrien semblar no tenir una justificació:

**LOCALS:** guarda els locals, siguin del tipus que siguin (botiga-magatzem, oficina, polivalent o immoble). Aquesta taula té tots els camps que tenen en comú tots els locals.

**BOTIGUES\_MAGATZEM:** te com a “pare” la taula LOCALS i guarda els camps específics per als locals de tipus botiga-magatzem. La raó de separar aquesta taula de LOCALS es perquè te camps que no tenen en comú altres locals i, gràcies a les claus foranies, per evitar en lo possible la inserció d'un immoble dintre d'un altra immoble.

**OFICINES:** te com a “pare” la taula LOCALS i guarda els camps específics per als locals de tipus oficina. La raó de separar aquesta taula de LOCALS es perquè podria tenir camps que no tenen en comú altres locals i, gràcies a les claus foranies, per evitar en lo possible la inserció d'un immoble dintre d'un altra immoble.

**POLIVALENTS:** te com a “pare” la taula LOCALS i guarda els camps específics per als locals de tipus polivalent. La raó de separar aquesta taula de LOCALS es perquè te camps que no tenen en comú altres locals i, gràcies a les claus foranies, per evitar en lo possible la inserció d'un immoble dintre d'un altra immoble.

**IMMOBLES:** te com a “pare” la taula LOCALS i guarda els camps específics per als locals de tipus immoble. La raó de separar aquesta taula de LOCALS es perquè te camps que no tenen en comú altres locals i, gràcies a les claus foranies, per evitar en lo possible la inserció d'un immoble dintre d'un altra immoble.

**IMMOBLE\_BOTIGA\_MAGATZEM:** guarda quines botigues-magatzem hi ha en un immoble concret, si en té. Utilitza una clau forana apuntant cap a IMMOBLES i altra cap a BOTIGUES\_MAGATZEM per tal d'evitar que no es fiquin locals d'altra tipus.

**IMMOBLE\_OFICINA:** guarda quines oficines hi ha en un immoble concret, si en té. Utilitza una clau forana apuntant cap a IMMOBLES i altra cap a OFICINES per tal d'evitar que no es fiquin locals d'altra tipus.

**IMMOBLE\_POLIVALENT:** guarda quins polivalents hi ha en un immoble concret, si en té. Utilitza una clau forana apuntant cap a IMMOBLES i altra cap a POLIVALENTS per tal d'evitar que no es fiquin locals d'altra tipus.

**ZONES:** guarda informació de cada zona.

**ENTITATS:** guarda informació de cada “entitat”, que pot ser una persona o una empresa. Aquestes entitats poden ser clients, usuaris (ens referim com a usuaris als treballadors), les dues coses o cap.

**PERSONES:** té com a “pare” la taula ENTITATS i guarda els camps específics per a les entitats que són persona. La raó de ficar dos camps, un per cognom, es per realitzar una separació coherent i necessària per si en un futur necessitem separar els cognoms o realitzar operacions amb un d'ells o amb els dos (a més, cal no oblidar l'existència de cognoms compostos que, si es tractessin els dos en un sol camp, dificultarien la identificació de cadascun).

**EMPRESSES:** té com a “pare” la taula ENTITATS i guarda els camps específics per a les entitats que són empresa.

**CLIENTS:** té com a “pare” la taula ENTITATS i guarda els camps específics per a les entitats que son un client. Aquestes podran ser persones o empreses (definida a partir d’un camp en la taula ENTITATS i també per la seva aparició en PERSONES o EMPRESSES).

**USUARIS:** té com a “pare” la taula PERSONES i guarda els camps específics per a les persones que son un usuari (ens referim a usuari com als usuaris reals del primer sistema analitzat, es a dir, als treballadors). Aquí es defineix quin tipus de treball fa cada usuari (agent, inspector o venedor), permetent també la possibilitat de ficar un treballador amb més d’un càrrec (hem cregut que podria donar-se el cas).

**PROPIETARIS:** té com a “pare” la taula CLIENTS i guarda els camps específics per a cada client que es propietari. Aquesta separació ens ajuda a evitar la introducció per error d’un client que no es propietari en un camp que no pot anar altre tipus de client. Aquesta taula té un camp que fa referència a CERQUES que en realitat es tractada com les seves preferències, degut a la similitud entre preferència i cerca. Així, també, les preferències poden ser tan amplies com ho es una cerca.

**LLOGATERS:** té com a “pare” la taula CLIENTS i guarda els camps específics per a cada client que es llogater (eventual o no). Aquesta separació ens ajuda a evitar la introducció per error d’un client que no es llogater en un camp que no pot anar altre tipus de client.

**TELEFONS:** guarda els telèfons. Hem cregut convenient la separació degut a la existència d’empreses (sobretot en vivers o locals compartits) o persones que poden compartir el número telefònic.

**TELEFON\_ENTITAT:** guarda quins telèfons té cada entitat, permetent tenir cada entitat més d’un telèfon. El camp que fa referència a TELEFONS no es únic per permetre que es pugui ficar el mateix número a més d’una entitat (per evitar això, no més caldria marcar com a únic aquest camp).

**ADRESES:** guarda les adreces. Com una adreça pot ser la mateixa en dues zones diferents (si són de dos pobles diferents, per exemple), tenim un camp que associa aquesta adreça a una zona de la taula ZONES. Això també ens ajuda a saber en quina zona es troba aquesta adreça.

**ADRESA\_ENTITAT:** guarda quines adreces té cada entitat, permetent tenir cada entitat més d’una adreça. Permetem que dues o més entitats puguin compartir la mateixa direcció (per casos com el esmentats abans, com el d’un pis compartit).

**COMPTE:** guarda les comptes, amb un nom (únic) i una clau, per l’accés al software i las relaciona amb la taula ENTITATS. Aquesta separació ajuda que puguin existir entitats sense compte, però mai una compte sense entitat. També evita que una entitat tingui cerques, ja que les cerques solament són de les entitats amb comptes.

**CERQUES:** guarda les cerques.

**CERCA\_COMPTE:** guarda les relacions entre un compte i una cerca. Aquesta separació es per donar la possibilitat de que una compta pugui utilitzar la mateixa cerca, per optimització del software (aquesta característica no la utilitzem, però hem cregut correcte deixar la base de dades amb aquesta capacitat per un futur). Ficar la cerca en comptes hauria impossibilitat associar més d'una cerca al mateix compte, i ficar la compte en la cerca hauria impossibilitat associar més d'un compte en la mateixa cerca.

**CERCA\_ZONA:** guarda la zona o zones que té associada una cerca.

**CONTRACTES:** guarda els contractes. La referència al propietari no era necessària ja que es podria arribar a aquest mitjançant la referència al immoble llogat, però la hem deixat així perquè ho especifica l'anàlisi i perquè en un futur es podria contemplar la possibilitat de que un local pogués tenir més d'un propietari a l'hora (intermediaris, per exemple).

### ***3.-DOCUMENTACIÓ DEL SOFTWARE D'ADMINISTRACIÓ DE LA BASE DE DADES***

#### ***3.1.-INTRODUCCIÓ***

Aquest apartat correspon al crèdit 8 del cicle.

A continuació s'inclou tota la documentació relativa al software d'administració de la base de dades.

Aquest software està desenvolupat en el llenguatge C# de la plataforma .NET, utilitzant Microsoft Visual C# 2005 Express Edition.

L'objectiu del software és facilitar, a la figura de l'administrador, la gestió de la base de dades que utilitza tot el sistema del projecte de la immobiliària. No està permesa la entrada a una altra compte que no sigui la d'administrador.

L'administrador ha de ser una persona o persones (amb el mateixa compte, ja que solament han de saber la contrasenya per entrar) amb la competència necessària i amb coneixements amplis sobre el software que ens ocupa i sobre la base de dades, ja que estarà manipulant informació molt important d'una aplicació que depèn molt de la base de dades. Així doncs, ha de conèixer la base de dades i el seu funcionament d'una forma profunda.

#### ***3.2.-FUNCIONAMENT INTERN***

##### **Control propi d'identificació:**

Aquest control ens permet, afegit al nostre projecte, que solament pugui entrar la figura de l'administrador i impedir la entrada a qualsevol altra entitat. És un control independent que pot ser integrat en qualsevol projecte que ho precisi i s'encarrega de demanar una clau.

##### **Projecte principal:**

Formularis del projecte principal:

**FormPrincipal:** és el formulari MDI del projecte. Conté un menú superior que està activat o desactivat segons convingui (per exemple, al no estar encara identificat el menú estarà desactivat). Quan s'inicia el programa, aquest formulari i una connexió a la base de dades s'inicia i quan s'acaba el programa el formulari i la connexió es tanquen.

El menú, quan està actiu, té opcions que criden al *FormGestio* (explicat més endavant) de forma específica (per veure algun aspecte en concret) o de forma general (deixant administrar tota la base de dades).

**FormLogin:** és el formulari que utilitza el control propi d'identificació que hem esmentat abans. Permet l'entrada a l'administrador de la web, que sempre es fa amb el nom *admin* de la taula *comptes*. En aquest cas,

hem decidit ficar la contrasenya ja ficada per facilitar al docent la prova del programa (en un entorn real, això s'hauria de treure).

**FormGestio:** és el formulari amb *datagrid* que permet administrar la base de dades. Cal recalcar que si es vol administrar la taula de comptes, el sistema ja no mostra la compte del *admin* a fi de que no es pugui modificar i estalviar possibles problemes derivats. Aquest formulari ha pogut ser cridat des d'una opció per veure una cosa concreta (llavors farà un *select* específic segons la opció, que podrà contenir més d'una taula alhora) o per administrar totes les taules en general (llavors deixarà escollir la taula que es vol administrar). Permet exportar els resultats a XML, per ser emmagatzemats, interpretats o processats en un altre programa extern.

**FormXML:** el formulari es crida des de *FormGestio* i mostra el XML que s'ha generat a partir de la informació que teníem en el *datagrid*. Aquest formulari ens deixarà copiar al *portapapers* tot el text o solament el que tinguem seleccionat. Cal recalcar que aquest es l'únic formulari que no es "fill" del MDI principal (*FormPrincipal*), per raons de facilitat cap a l'usuari ja que així podrà veure més d'una d'aquestes finestres i alternar de manera fàcil des d'aquestes amb un programa extern de forma àgil. Al acabar el programa el programa (al tancar el formulari *FormPrincipal*), totes les finestres *FormXML*, encara que no son filles, es tanquen també.

### 3.3.-FUNCIONAMENT EXTERN

L'ús del software ha estat pensat per ser senzill de cara a l'usuari administrador. Encara que es innecessari, expliquem breument el seu funcionament en diferents passos:

- L'usuari administrador inicia el programa i se li demana la clau d'administrador (és la compte amb nom *admin* de la taula *comptes*). Cal recordar que per facilitar la prova del programa, hem deixat la contrasenya ja ficada (en la realitat, l'hauríem d'esborrar).
- L'usuari introdueix correctament la clau d'administrador i accedeix al sistema.
- L'usuari escolleix una opció específica del sistema, que li permetrà veure la informació de la base de dades relativa a un aspecte concret (abans de fer una modificació, és necessari conèixer bé quin es l'estat de la base de dades) o bé podrà administrar directament la base de dades amb la opció *Totes*.
- Si la opció escollida es específica, podrà veure la informació d'un aspecte en concret però no podrà modificar-la. Aquest aspecte es merament informatiu de cara a l'administrador. Si la opció es *Totes*, podrà administrar taules de la base de dades fent canvis i guardant-los quan estigui segur (si ho està).
- Totes les opcions (específiques o la general) permeten exportar les dades que està veient en el *datagrid* a codi XML. Si l'usuari escolleix l'opció de exportar a XML, accedirà a una finestra que podrà veure el codi i copiar-lo al *portapapers* per utilitzar-lo en qualsevol cosa que ell consideri pertinent.
- Quan l'usuari administrador ha acabat, solament ha de tancar el formulari principal per terminar.

## 4.-DOCUMENTACIÓ DE LA WEB PÚBLICA

Aquest apartat correspon al crèdit 9 del cicle.

La web està pensada per ser modificada i ampliada fàcilment. Coses com afegir un nou apartat o modificar l'aspecte de tots els apartats (o altres modificacions) poden ser fetes sense cap mena de problema i d'una forma molt simple.

Hem intentat fer la web separada en capes, utilitzant el patró de disseny MVC (Model, Vista, Controlador). Fins i tot, aquestes capes les hem pogut dividir en alguns casos en subcapes.

El disseny de la web també està pensat per ser modificat en tot moment de manera àgil, tenint solament que modificar el CSS i, si cal, les imatges. Per els gràfics hem fet servir *Adobe Fireworks*.

També hem tingut en ment l'accessibilitat, intentant crear una web fàcil d'utilitzar per a persones amb discapacitats visuals o d'altre tipus. Això es gràcies al disseny simple i directe, a la utilització de HTML validat per estar acord en lo màxim amb la w3c, no impeding la modificació del seu aspecte (tipus de lletres, ampliació de fonts, colors...), etc.

La *usabilitat* del lloc web la hem tingut en compte també en tot moment, fent una web simple i directa, intentant fer-la lo més funcional possible i sense complicacions innecessàries per part del visitant.

Aquesta web es veu correctament i funciona a la perfecció en tots els navegadors moderns i fins i tot en els més antics. Ha estat provada en *Mozilla Firefox*, *Internet Explorer*, *Opera* i *Safari*. Podem dir, per tant, que és una web amb alta capacitat *cross-browser*.

La web també permet la implementació, de manera molt fàcil, de *meta-tags* i altres tècniques per millorar la seva presència en cercadors i motors d'indexació. És una capacitat que no hem fet servir però que la hem tingut en compte per poder fer-la realitat si en un futur és precís.

Hem intentat reestructurar la web ordenant i separant els seus arxius en diferents apartats, segons hem cregut convenient.

Per la part de la *vista d'usuari* (HTML i CSS):

**CSS:** hem ficat en un directori (*css*) els estils en CSS (hi ha més d'un arxiu ja que l'usuari registrat podrà escollir qui l'agrada més en l'apartat de *Personalitzar* de la web). Tot l'aspecte de la web depèn del CSS i de les imatges (en el directori *img*).

**HTML:** per al HTML hem fet una abstracció, fent que totes les pàgines utilitzin el mateix arxiu font (que conté el HTML). Aquest arxiu font alhora fa servir altres arxius font per separar-lo en el registre/identificació (*login*), el menú (que són tres), cos del document i peu. Això ha simplificat la lògica de l'aplicació i el seu desenvolupament, així com la minimització de riscos i la facilitació per crear nou contingut o modificar el ja existent. Molt del HTML es crea dinàmicament, gràcies a PHP.

Per la part de *programació del costat del client* (JavaScript):

**JavaScript:** Els arxius que utilitzen JavaScript s'han separat segons el seu funcionament, classificant-los en: funcions generals, estils (modifica la visualització i comportament de la web en temps d'execució) i funcions relatives als formularis de la web. També hem utilitzat software extern en JavaScript per crear un selector de data (*Date Time Picker*, creat per TengYong Ng) i *Google Maps* (que utilitza AJAX).

**DOM:** hem utilitzat aquest model mitjançant JavaScript per accedir als elements que conformen el HTML.

Per la part de la *programació del costat del servidor* (PHP i mySQL):

**PHP:** el PHP s'encarrega de la part *controladora*, on tenim tota la lògica de l'aplicació web en la part del servidor. El codi PHP està ficat en un directori apart (*php*).

**mySQL:** el mySQL s'encarrega de la part *model* (dades) i funciona gràcies al objecte controlador de la base de dades creat en PHP (més tard comentarem més sobre això).

No hem tingut temps per aplicar altres tecnologies com ara *AJAX* (fent servir un objecte de JavaScript), però hem fet la web pensant en que sigui possible la seva aplicació de manera simple. La web no hauria de sofrir casi canvis i els pocs canvis, tal com hem pensat i construït la web, no suposarien gaire esforç.

A continuació comentem l'estructura de directoris i què és cadascun dels arxius:

```
css/ 'Directori amb tot lo relatiu als CSS
    estils.css 'Estil que s'aplica per defecte
    estils2.css 'Estil que es pot escollir (podríem ficar més)
img/ 'Directori amb tot lo relatiu a les imatges
    logo.png 'Logotip de la web
intranet/ 'Directori amb tot lo relatiu a la gestió d'usuaris (en realitat conté lo que hauria d'anar a la intranet)
    administrador/ 'Apartat de gestió de la base de dades (relatiu al crèdit 8)
        LlogaTotGestio.zip 'Programa de gestió de la base de dades, fet en C# (demana contrasenya)
    usuarios/ 'Apartat del programa per als treballadors (relatiu al crèdit 6)
        *. * 'Programa fet en JSP (demana contrasenya).
js/ 'Directori amb tot lo relatiu als arxius de JavaScript
    datetimepick/ 'Directori amb el Date Time Picker (selector de dates)
        *. * 'Arxius del Date Time Picker
    general.js 'Arxiu amb funcions generals de JavaScript
    estils.js 'Arxiu amb funcions en JavaScript relatives al disseny i estils visuals
    formularis.js 'Arxiu amb funcions en JavaScript relatives als formularis (validacions, etc)
php/ 'Directori amb tot lo relatiu al PHP
    classes/ 'Directori amb les classes fetes en PHP (els objectes i mètodes necessaris)
        bs.php 'Objecte controlador de la base de dades
        contractes.php 'Conté la classe que representa un contracte (Contracte)
        situacio.php 'Conté les classes relatives a la situació d'una entitat o d'un local, per adreces i zones (Adresa i
Zona)
        locals.php 'Conté les classes relatives als locals (Local, BotigaMagatzem, Oficina, Polivalent i Inmòbil).
        general.php 'Conté les classes generals, que hem preferit classificar-les separades i juntes (Immobiliària i Data)
        entitats.php 'Conté les classes relatives a les entitats (Entitat, Persona, Empresa, Client, Usuari, Propietari,
Llogater, Compte, Cerca, i Telefon)
    inc/ 'Directori que conté els arxius en PHP que en realitat son contingut de la web (generen HTML, CSS i JavaScript).
        contingut/ 'Directori que conté el contingut específic del cos d'una categoria del menú de la web
            index.php 'Contingut del cos del apartat d'inici
            noticies.php 'Contingut del cos del apartat de notícies
            personalitzar.php 'Contingut del cos del apartat de personalitzar
            ubicacio.php 'Contingut del cos del apartat d'ubicació
            registrarse.php 'Contingut del cos del apartat de registrar-se
            ultimes_cerques.php 'Contingut del cos del apartat d'últimes cerques
            cercar.php 'Contingut del cos del apartat de cercar locals
            html.php 'Conté l'estructura general en HTML per a tots els apartats de la web i les accions en PHP que
processen tots.
```

menu.php 'Conté el menú per a tots els visitants i agrega el menú pertinent segons estigui validat l'usuari o no i la seva categoria  
 menu\_registrat.php 'Conté el menú per a usuaris registrats  
 menu\_usuari.php 'Conté el menú per a usuaris treballadors  
 menu\_administrador.php 'Conté el menú per als usuaris administradors (en el nostre sistema solament hi ha un administrador)  
 cos.php 'Conté el contingut del cos que es comú en tots els apartats de la web  
 peu.php 'Conté el contingut del peu que es comú en tots els apartats de la web  
 login.php 'Conté el contingut de la identificació dels usuaris, i la seva lògica.  
 general.php 'Conté les funcions generals en PHP, necessàries per a diferents objectes i altres funcions o mètodes.  
 configuracio.php 'Conté la configuració de la web (nom de la empresa, configuració de la base de dades, títols dels apartats, etc)  
 index.php 'Apartat d'inici  
 noticies.php 'Apartat de notícies  
 personalitzar.php 'Apartat de personalitzar l'aspecte de la web  
 ubicacio.php 'Apartat d'ubicació de la empresa  
 registrarse.php 'Apartat de registrar-se  
 ultimes\_cerques.php 'Apartat de veure les últimes cerques  
 cercar.php 'Apartat de cercar locals

El mapa del web es aquest (amb totes les opcions del menú, es a dir, tal com ho veu un administrador):

**Registre** (es recarrega a si mateixa): permet registrar-se com a usuari  
**Identificació** (es recarrega en la pàgina actual): permet identificar com a usuari  
**Inici**: pàgina web principal, de "benvinguda"  
**Notícies**: notícies de la web i de la empresa  
**Cerques locals** (es recarrega a si mateixa): permet fer cerques per trobar locals  
**Ubicació**: mostra la situació de la empresa (utilitza *Google Maps*)  
**Últimes cerques** (si es pitja en una cerca per veure-la, va cap a Cerques locals, si no es recarrega a si mateixa): permet veure les últimes deu cerques de l'usuari registrat.  
**Personalitzar web**: permet canviar l'estil CSS de la web, modificant el seu aspecte.  
**Administrar**: permet als treballadors administrar la empresa (relatiu al crèdit 6)  
**Gestionar base de dades**: permet al administrador gestionar la base de dades (relatiu al crèdit 8)

Com a exemple de la seva facilitat de desenvolupament, esmentem a continuació com afegir una nova secció:

- 1) Es crea un arxiu amb el nom desitjat en l'arrel de la web, i es fica al següent contingut:
 

```
<?php
    require_once "php/inc/html.php";
?>
```
- 2) Es crea un enllaç cap a aquesta secció en el menú que convingui: en menu.php si es per a tots, en menu\_registrat.php si solament es per a usuaris registrats, en menu\_usuari.php si es per a usuaris treballadors i en menu\_administrador si es per a administradors solament.
- 3) Es crea un arxiu amb el mateix nom que en el punt 1 però en el directori php/inc/

Solament fa falta omplir amb el contingut desitjat l'arxiu del punt 3. Conèixer les funcions i objectes que ja té la web ens pot estalviar molta feina si volem programar coses que ja estan fetes.

## ***5.-DOCUMENTACIÓ DE LA WEB INTERNA DEL PERSONAL***

### ***5.1.-INTRODUCCIÓ***

Aquest apartat correspon al crèdit 6 del cicle.

A continuació s'inclou tota la documentació relativa al software d'administració de locals i clients (tant propietaris com llogaters o ambdós)

Aquest software està desenvolupat en el llenguatge Java de la plataforma de Sun Microsystems, utilitzant l'IDE NetBeans 5.5.1, utilitzant els patrons de programació Model-Vista-Controlador, amb el framework Struts 1.6.

L'objectiu del software és facilitar una aplicació web per una apresa de lloguer de locals, el software permetrà gestionar els locals, clients (propietaris i llogaters) i els contractes. En aquesta aplicació només està permesa l'entrada dels treballadors de l'empresa.

### ***5.2.-FUNCIONAMENT INTERN***

#### **JavaServer Pages**

Contindran tots els formularis necessaris per omplir els beans per introduir locals, clients o donar d'alta nous contractes.

Totes les pàgines mantenen una estructura similar, mostrant a l'esquerra en tot moment la pàgina login.jsp de manera que surti sempre un menú amb el qual es pugui accedir a qualsevol de les pàgines que es té accés.

**altaLocal.jsp:** Mostra el formulari per introduir qualsevol local, de qualsevol tipus, la resta de formulari està compost per l'adreça, zona, preu, característiques, restriccions, volum (s'introduirà a la bd en cas de que es seleccioni botiga-magatzem) i característiques polivalents (en cas de que es seleccioni un local polivalent)

**afegirLocals.jsp:** Mostrarà el formulari per introduir locals de tipus oficina, botiga-magatzem o polivalent dins del immoble que es seleccionarà al formulari, les demes opcions a omplir són el tipus, la planta i pis que ocupa del immoble (que s'afegiran darrera de la direcció del immoble), preu, característiques, restriccions, volum (s'introduirà a la bd en cas de que es seleccioni botiga-magatzem) i característiques polivalents (en cas de que es seleccioni un local polivalent)

**altaPropietari.jsp:** Mostra el formulari per introduir llogaters, el formulari està compost pels camps de nom i cognoms o raó social (només s'acceptarà o nom i cognoms o raó social), adreça, zona, telèfon, i el NIF

**altaLlogater.jsp:** El formulari que mostrarà serà idèntic al de altaPropietari.jsp amb la única addició de tres camps més: zona desitjada, tipus de local desitjat i superfície mínima desitjada.

**index.jsp:** Sera la pagina d'inici de l'aplicació, per al moment no mostrarà res,

**login.jsp:** Aquesta pagina es mostrarà en totes les pagines jsp restants, consisteix en un simple formulari de login amb que demana nom d'usuari i clau.

El formulari esta lligat al action *actions.Login* i al bean *entitats.compte*, la pàgina controla si existeix un compte a la sessió amb el nom i clau introduïts, si n'és el cas, mostra el menú del compte cridant a un mètode del bean *creaMenu()* el qual retornarà un string, que serà els enllaços a les pàgines dels formularis.

## Action

Les accions que procedeixen als enviaments dels formularis. Tots els actions contenen un bloc try (a on s'agafa la connexió i es fan els inserts i/o updates necessaris) amb el seu catch i el finally (indispensable ja que es on es tanca la connexió perquè no estigui ocupant un socket) un cop acabat fa el *findForward* cap al formulari d'on procedeix.

El llistat de actions (dins del package *actions*) son els següents: AfegirAllImmoble, AltaLlogater, AltaLocal, AltaPropietari, Login

## ActionForm Beans

Els beans s'ompliran amb les dades del seu formulari associat. Cada bean té un atribut comú, el d'error el qual mostra informació resultant del action (tant si ha anat sense problemes com s'hi hi ha hagut alguna excepció).

La majoria dels beans contenen un mètode *comprovaDadesFormulari* el qual retorna un boolean, la funció del mètode és comprovar que l'usuari que interactua amb el sistema no s'ha deixat cap camp buit obligatori del formulari, la crida del mètode es fa a través el seu action corresponent.

Els actions queden distribuïts en 3 packages:

- contractes: Contracte
- entitats: Cerca, Client, Compte, Empresa, Entitat, Llogater, Persona, Propietari, Telefon, Usuari
- locals: BotigaMagatzem, Immobile, Local, Oficina, Polivalent
- situacio: Adresa, Zona

També queda implícit que en cada bean hi han els sets i gets públics corresponents als seus atributs.

A continuació descriurem els beans que hem usat fins al moment:

**Propietari:** Conté la informació necessària per omplir totes les taules necessàries per tal de donar d'alta un propietari, s'omplirà a través del formulari *altaPropietari*, i l'utilitzarà l'action *action.altaPropietari*.

**Llogater:** Conté la informació necessària per donar d'alta un nou llogater, cal remarcar que tant aquest bean com l'anterior esmentat (Propietari) hereten de *entitat.Client*, per lo qual algunes de les dades del formulari es carreguen al bean pare.

Aquest bean s'omplirà a traves del formulari *altaLlogater*, i l'utilitzara l'action *actions.altaLlogater*

**Local:** Conté la informació necessària per omplir totes les taules necessàries per tal de donar d'alta un local, s'omplirà a través del formulari `altaLocal`, i l'utilitzarà l'accion `action.AltaLocal`

**Compte:** Conté la informació del usuari que entra en el sistema a través del login, aquest bean quedarà guardat a la sessió a diferencia de la resta del beans.  
El bean s'omplirà a través del formulari de la pagina login, i l'utilitzarà l'accion `action.login`

## Servlets

Tots els servlets de l'aplicació tenen com a missió omplir `ArrayList` que posteriorment serviran per omplir els options dels formularis amb les opcions que estan a la base de dades.

**CarregaImmables:** Carrega tots els immobles, la crida al servlet es fa cada cop que s'accedeix al formulari `afegirLocals.jsp`.

**CarregaLocalsPendants:** Carrega tots els locals que encara estan pendants d'inspecció, la crida al servlet es fa cada cop que s'accedeix al formulari `localsPendants.jsp`.

**CarregaZones:** Carrega totes les zones de la base de dades per tal que al afegir una adreça, poder-la associar a una zona, la crida al servlet es farà solament un cop al accedir al sistema.

**CarregaPropietaris:** Carrega tots el propietaris actius per tal de a la hora de donar d'alta un local, poder associar-lo a un local, la crida al servlet es fa cada cop que s'accedeix al formulari `altaLocal.jsp`

## 5.3.-FUNCIONAMENT EXTERN

L'ús de l'aplicació web es bastant senzill i simple.

Només entrar a la web, apareixerà un login a la part esquerra el qual serà obligatori per poder accedir a qualsevol de les pagines web.

Un cop s'ha accedit al sistema, dependrà de quin tipus d'usuari ha accedit apareixerà un menú completament diferent els quals descriurem a continuació:

Per als agents: Alta local, Afegir locals a un immoble, Alta propietari.

Per als inspectors: Confirmar local.

Per als venedors: Crear contracte, Alta llogater.

Cada formulari esta fet perquè retorni a ell mateix mostrant un missatge de com ha anat el procés.

## 6.-CONSIDERACIONS SOBRE L'APLICACIÓ

Es necessari tenir documentats certs aspectes que fan referència al software desenvolupat, per tal de recalcar les restriccions de l'aplicació, ja que es precis -o pot ser-ho en un moment donat- tenir-les en compte.

El software final està desenvolupat en quatre aspectes ben diferencies, cadascun amb les seves característiques:

**La part de la web pública:** està desenvolupada utilitzant, entre altres tecnologies, el llenguatge de programació *PHP*. Això fa necessari el tenir allotjada la web en una maquina amb un servidor que suporti aquest llenguatge. Aquesta maquina ha de ser visible mitjançant Internet per poder ser accedida des de fora. Per part del client, no es necessari el suport de *JavaScript* ni de *CSS*, però facilitarà i millorarà l'experiència d'usuari.

**La part de la web privada:** està desenvolupada utilitzant *JSP*, per tant serà necessari una maquina amb un servidor que suporti aquesta tecnologia. Aquesta maquina no necessitarà tenir entrades activades des d'Internet, i si ho te es recomanable restringir l'accés amb protocols i mesures addicionals de seguretat a part dels utilitzats pel nostre software. Es recomenable també utilitzar també el mateix software servidor utilitzat en el seu desenvolupament, que correspon a *Jakarta Struts 1.2.9* com *Apache Bundled Tomcat 5.5.17*, a fi de minimitzar i tenir documentats els possibles contratemps.

**La part d'administració:** en aquest cas es tracta d'un programa natiu i no d'una web. Aquest programa ja té mesures de seguretat però seria preferible restringir el seu accés al mínim de persones possibles i d'una forma regulada ja que existeix la possibilitat de trencar la seva seguretat utilitzant tècniques com *força bruta*, etc. que podrien comprometre la seguretat de l'aplicació. Aquest programa solament funcionarà en *Microsoft Windows XP* o compatible i amb *.Net Framwork 2.0* o posterior. També existeix la possibilitat d'utilitzar software d'emulació com ara *Wine*, d'emulació/virtualització com *QEmu* o de virtualització com *VMWare*, *Parallels*, etc. De qualsevol forma, emfatitzem que no es recomanable el seu ús ja que podria derivar, de forma més pronunciada que en un entorn natiu, en un comportament del software que no estigués previst.

## ***7.-CONCLUSIONS PERSONALS SOBRE EL PROCÉS DESENVOLUPAT***

### **Juan Alba Maldonado:**

Aquest crèdit de síntesi ha estat molt complet en quant a tecnologies i requeriments. Personalment crec que aquesta plenitud i complexitat han derivat en una dificultat que no estava acord, tenint en compte la nostra inexperiència, amb el temps disponible.

Si bé crec que podria ser factible la realització de tot el projecte amb tots els seus requisits, i segurament ampliar-lo més, fins i tot en un espai de temps igual o més reduït, també crec que alguns errors que hem comés ja des del principi degut a la falta d'experiència (no de formació) ens han repercutit durant tota la realització posterior.

Aquests errors i conseqüències, ja que ens ha estat impossible buscar una solució totalment viable i sense repercussions, són millor abordar-los des del punt de vista optimista i considerar-los com una oportunitat d'aprendre d'ells.

Crec que tenim els coneixements necessaris per assolir tot el que ens demanen i l'únic que necessitem és experiència. Aquest crèdit de síntesi ha millorat i ampliat la nostra experiència i ens ha brindat l'oportunitat d'aprendre dels nostres errors. I ara, de cara a la vida laboral, podrem fer front a altres problemes amb una mica menys de dificultat.

Com l'objectiu del personal docent és la docència i no la penalització, crec que seria correcte valorar el nostre treball i el de tots els nostres companys d'una perspectiva que valorés més l'assoliment d'experiència i actitud en vers als fets derivats que el resultat obtingut. Més tenint en compte que, segons la meva opinió, tots els alumnes estem ja qualificats per dur a terme els objectius requerits. En quant a l'experiència que ens pot faltar, mai hem d'oblidar que no es pot ensenyar.

### **Ivan Andreu Florensa:**

El crèdit de síntesi ens ha ajudat a aplicar els coneixements adquirits durant aquests dos anys en un sol projecte el qual comentaré per parts dividides de manera coherent.

Pel que fa l'anàlisi es una cosa que requereix bastanta experiència, tot i això ens varem centrar tots dos en desenvolupar l'anàlisi debatent quines eren les millors solucions per el projecte, tot i que crec que no hi vam dedicar massa temps, després vam veure que el temps pel desenvolupament sens va quedar curt per tota l'aplicació.

La part de la base de dades ha estat bastant complexa, la base de dades resultant pot semblar molt gran per un simple projecte i programar sobre ella pot tornar-se una utopia, però mante una lògica i consistència que la fa capaç d'evitar errors que puguin passar per alt el sistema.

La programació del programa LlogaTotGestió (l'encarregat de poder editar les taules) ha estat el més curt (però no per això simple) dels tres apartats de programació, quan ja teníem la base programada vam deixar els apartats d'XML i de fer DataGrids pare-fill (això últim no demanat en els requeriments) per quan acabéssim la resta de programari (que no s'ha acabat donant el cas), tot i això hem aconseguit una aplicació bastant complexa amb un control GUI d'accés, formularis MDI que permeten veure taules indexades (sense poder editar-les) i la possibilitat de veure i/o editar qualsevol de totes les taules de la base de dades.

La part de programació PHP ha estat amb la que l'hi hem dedicat més temps, hem aconseguit una estructuració de la web molt bona el qual ajuda que afegir-hi més seccions sigui cosa d'uns minuts, cosa que en php depèn com pot ser bastant complicat.

Pel que fa la part de Java, es la que menys temps l'hi hem pogut dedicar, es nota bastant tot i que el que hem fet funciona a la perfecció no hem arribat a acabar-ho ni organitzar-ho del tot bé per la falta de temps