

ÍNDEX

ÍNDEX.....	1
1 INTRODUCCIÓ I OBJECTIUS	6
1.1 Introducció	6
1.2 Propòsit	6
1.3 Context i justificació	6
1.4 Objectius	7
2 ÀMBIT I ABAST DEL PROJECTE.....	8
2.1 Àmbit	8
2.2 Abast.....	8
2.3 Eines utilitzades.....	8
2.3.1 Llenguatges de programació	9
2.3.2 Entorn de desenvolupament.....	11
2.3.3 Altres eines de software utilitzades	12
2.4 Requeriments del Sistema	13
2.5 Estudis i Decisions	13
2.5.1 Què és Joomla!?.....	13
2.5.2 Decisions.....	14
2.5.3 Patró MVC i Extensió de classes.....	15
3 ESTUDI DE VIABILITAT	17
4 METODOLOGIA.....	18
4.1 Procés Unificat de Rational (RUP)	18
4.2 eXtreme Programming (XP)	18

5	ANÀLISI I DISSENY DEL SISTEMA.....	20
5.1	Joomla!.....	20
5.1.1	Estructura de desenvolupament.....	20
5.1.2	Funcionament.....	21
5.2	Component “com-property”.....	22
5.2.1	Funcionalitats.....	22
5.2.2	Estructura de directoris i fitxers.....	24
5.3	Mòdul de Clients.....	25
5.3.1	Objectius.....	25
5.3.2	Model d’Entitat-Relació.....	26
5.3.2.1	Diagrama Entitat-Relació.....	27
5.3.2.2	Model Relacional.....	27
5.3.3	Descripció general.....	30
5.3.4	Diagrama de casos d’ús usuari Gestor.....	31
5.3.5	Casos d’ús.....	31
5.3.5.1	Alta Client.....	31
5.3.5.2	Baixa Client.....	32
5.3.5.3	Modificació Client.....	32
5.3.5.4	Assignar Usuari a Client.....	33
5.3.5.5	Llistar Clients.....	33
5.3.6	Interfícies d'usuari.....	34
5.3.6.1	Alta de Clients.....	34
5.3.6.2	Baixa de Clients.....	34
5.3.6.3	Modificar Clients.....	35

5.4	Mòdul de Preus	36
5.4.1	Objectius.....	36
5.4.2	Model d'Entitat-Relació.....	36
5.4.2.1	Diagrama Entitat-Relació.....	37
5.4.2.2	Model Relacional.....	37
5.4.3	Descripció general	40
5.4.4	Diagrama de casos d'ús usuari Gestor	41
5.4.5	Casos d'ús.....	41
5.4.5.1	Alta Preu	41
5.4.5.2	Baixa Preu.....	42
5.4.5.3	Modificació Preu	42
5.4.6	Interfícies d'usuari	43
5.4.6.1	Alta Preu	43
5.4.6.2	Baixa Preu.....	43
5.4.6.3	Modificar Preu.....	44
5.5	Mòdul de Facturació	44
5.5.1	Objectius.....	44
5.5.2	Model d'Entitat-Relació.....	44
5.5.2.1	Diagrama Entitat-Relació.....	45
5.5.2.2	Model Relacional.....	45
5.5.3	Descripció general	49
5.5.4	Diagrama de casos d'ús usuari Gestor	49
5.5.5	Casos d'ús.....	50
5.5.5.1	Alta Factura	50

5.5.5.2	Modificar Factura	50
5.5.5.3	Baixa Factura	51
5.5.5.4	Alta Detall Factura.....	51
5.5.5.5	Modificar Detall Factura.....	52
5.5.5.6	Baixa Detall Factura.....	52
5.5.6	Diagrama de casos d'ús usuari Client.....	53
5.5.6.1	Visualitzar Factures	53
5.5.7	Interfícies d'usuari.....	54
5.5.7.1	Alta Factura	54
5.5.7.2	Baixa Factura	54
5.5.7.3	Modificació Factura.....	55
5.5.7.4	Alta, Baixa i Modificació Detall Factura.....	56
5.5.7.5	Visualitzar Factures	56
6	IMPLEMENTACIÓ I PROVES.....	58
6.1	Implementació. Càlcul de l'Import.....	58
6.2	Càlcul	60
6.2.1	Exemple de Càlcul	60
6.3	Generació de Factures amb PDF	61
6.4	Implementació. Directoris.....	62
6.5	Proves.....	66
6.5.1.1	Conclusió de les proves realitzades.....	67
6.6	Accés a l'aplicació.....	67
7	CONCLUSIONS	70
8	TREBALL FUTUR.....	71

9	INSTAL·LACIÓ.....	72
10	BIBLIOGRAFIA / WEBGRAFIA.....	73
11	ÍNDEX DE FIGURES.....	74

1 INTRODUCCIÓ I OBJECTIUS

1.1 Introducció

En aquest Projecte de Final de Carrera d'Enginyeria Tècnica en Informàtica de Gestió dissenyaré i implementaré una aplicació per incorporar noves funcionalitats a la gestió de propietats immobles. En els últims anys, el lloguer d'apartaments i cases ha augmentat el seu volum de reserves a través d'Internet, i és per això, que moltes empreses del sector han dedicat molts esforços en potenciar aquest canal i fer tots els tràmits a través de la xarxa. L'hivern del 2009 l'empresa a qui va dirigida aquest projecte, va decidir fer aquest salt i irrompre a Internet. En aquell moment es van plantejar diferents opcions per a la gestió d'immobles: fer una aplicació totalment nova i a mida, comprar una aplicació ja feta a empreses especialitzades o adaptar una aplicació gratuïta a les necessitats de l'empresa. Analitzats amb detall els tres casos, es va decidir implementar la tercera opció ja que satisfieix les necessitats de l'empresa i amb modificacions o adaptacions aconseguiria el resultat esperat. L'aplicació gratuïta és el "com-property" i la seva posada en marxa va ser en dos mesos.

1.2 Propòsit

El propòsit d'aquest Projecte Final de Carrera és la creació de noves funcionalitats adaptades a un client, una empresa immobiliària, per tal de fer la gestió dels clients i la facturació.

1.3 Context i justificació

El client per qui s'ha realitzat aquest projecte és l'empresa Port la Sardana, la qual es dedica a la venda i lloguer de cases i apartaments a la zona de l'Alt Empordà. Actualment, l'empresa té un portal¹ a Internet desenvolupat amb [Joomla!](http://www.joomla.org) i té instal·lat un component anomenat "com-property" que li permet fer la gestió de les propietats immobles, la reserva i el lloguer dels habitatges. A més a més, té una aplicació realitzada amb Access per la gestió dels clients.

¹ www.portlasardana.com

Davant de la necessitat del client d'informatitzar i agrupar totes les funcionalitats en un mateix sistema, es decideix incorporar la gestió dels clients en aquest component instal·lat i crear una nova funcionalitat per la facturació de l'empresa.

Durant la realització del projecte l'empresa immobiliària ha registrat una altra marca comercial amb el nom d'Empuriarent, a part de l'existent Portlasardana. De manera que es poden trobar aquestes dues marques a l'aplicatiu web.

1.4 Objectius

L'objectiu principal d'aquest Projecte Final de Carrera és desenvolupar l'anàlisi, disseny, implementació i posta en marxa dels següents punts:

- Adaptar el component "com-property" a les necessitats específiques de l'empresa.
- Facilitar i agilitzar l'ús de totes les interfícies web al personal de l'empresa.
- Desenvolupar un mòdul de gestió de clients d'una immobiliària per Joomla!.
- Desenvolupar un mòdul de facturació d'una immobiliària per Joomla!.

Per tal d'assolir aquests objectius ha estat necessari :

- Conèixer el funcionament de la plataforma Joomla!.
- Aprendre els llenguatges PHP i Javascript, amb els que està programada la plataforma.
- Entendre l'estructura del mòdul "com-property" en Joomla!. Els components de Joomla! s'estructuren seguint el patró Model Vista Controlador (MVC), els quals he estudiat amb dedicació per tal d'entendre la programació amb la qual ha estat implementat.
- Conèixer la disposició de les taules de la base de dades, els camps que contenen i les seves relacions.

2 ÀMBIT I ABAST DEL PROJECTE

En aquest apartat es descriurà l'àmbit en què es desenvolupa aquest projecte i quin serà el seu abast. Es descriuen les diferents eines que s'han d'utilitzar per tal de realitzar les millores proposades.

2.1 Àmbit

Aquest projecte s'emmarca dintre del desenvolupament d'aplicacions web realitzades amb Joomla! L'aplicatiu web consisteix en la millora d'un component ja existent "com-property".

2.2 Abast

Donats els objectius del Projecte Final de Carrera, el seu abast és:

- Obtenir els requeriments per part del personal de l'empresa.
- Introduir-se en el framework de programació de Joomla!.
- Fer un estudi d'usabilitat i accessibilitat de la web actual de la immobiliària per tal de millorar l'experiència dels usuaris.
- Analitzar el disseny i el funcionament del mòdul "com-property".
- Ampliar, millorar i adaptar el mòdul "com-property" a les necessitats de l'empresa.
- Agilitzar els processos administratius existents per tal que siguin més ràpids.
- Crear el nou mòdul de gestió de clients per Joomla!.
- Desenvolupar un nou mòdul per Joomla! per tal de fer generar factures i pressupostos.

2.3 Eines utilitzades

Per dur a terme el projecte s'han utilitzat diferents eines. En cada etapa del projecte s'ha utilitzat la més adequada. Aquestes es poden agrupar en:

- *Llenguatges de programació*: aquestes eines s'han utilitzat en l'etapa d'implementació del projecte.
- *Entorns de desenvolupament*: aquests entorns són necessaris per tal d'agilitzar l'etapa d'implementació, ja que fan més còmode la tasca de programar.

- *Altres tipus d'eines:* s'han utilitzat per elaborar la documentació del projecte.

2.3.1 Llenguatges de programació

Els llenguatges de programació que s'usen quan es treballa amb Joomla! són:

- **PHP (*PHP Hypertext Preprocessor*):** és un llenguatge de programació interpretat. Va ser dissenyat originalment per la creació de pàgines web dinàmiques, es pot incrustar dins el codi *HTML*. El seu funcionament és molt senzill s'executa en un servidor web que contingui el PHP, el que fa el servidor web és agafar el codi en PHP com entrada i l'interpreta, i com a resultat n'obtenim pàgines web.

Es pot utilitzar en la majoria dels sistemes operatius del mercat, incloent Linux, moltes varietats de Unix, Microsoft Windows, Mac OS X,..., igual que suporta la majoria de servidors web d'avui dia, incloent Apache, Internet Information Server,...; sense cost ja que és un llenguatge Open Source. D'aquesta manera, si programem amb aquest llenguatge tenim la llibertat de poder escollir el sistema operatiu i servidor web que més ens convingui.

Els principals avantatges d'aquest llenguatge són els següents:

- És un llenguatge multiplataforma.
- Té capacitat de connexió amb la majoria de sistemes de gestió de base de dades que s'utilitzen en l'actualitat.
- Llegir i manipular dades de diverses fonts, incloent dades que poden introduir els usuaris als formularis *HTML*.
- Capacitat d'expandir el seu potencial amb mòduls o extensions.
- Trobem una àmplia documentació d'aquest llenguatge a la web.
- Llenguatge orientat a objectes.



Figura 2.3.1.1. Icona de PHP.

- **JavaScript:** és un llenguatge de programació interpretat, és a dir, que no requereix que sigui compilat. S'utilitza principalment en pàgines web, amb una sintaxis molt semblant a la del llenguatge *Java*. Igual que *Java*, *JavaScript* és un llenguatge orientat a objectes, ja que disposa de les eines mínimes per realitzar l'orientació a objectes com poden ser herència, encapsulació, ... S'utilitza en pàgines web HTML, per realitzar tasques des del client, de tal manera que realitzant validacions des de la part del client, estalviant així "viatges" al servidor. La majoria de navegadors interpreten el codi *JavaScript* integrat dins les pàgines web, tot i que a vegades ens podem trobar amb alguna dificultat.
- **MySQL:** és un sistema de gestió de base de dades relacional (RDBMS) que utilitza el llenguatge SQL (Structured Query Language). MySQL ha esdevingut molt popular gràcies a la seva velocitat en executar consultes i el seu suport de forma nativa per part del llenguatge PHP. Es pot fer ús de MySQL en aplicacions de tota mena, de forma lliure i gratuïta sota les condicions de llicència GPL.



Figura 2.3.1.2. Icona de MySQL.

- **HTML (*Hyper Text Markup Language*):** és un llenguatge de marques molt senzill que s'utilitza per crear els textos i pàgines web. Aquest es basa en marques/etiquetes que defineixen estils per crear els hipertextos. Aquesta definició es deu a que està compost per etiquetes que defineixen l'estructura i el format del document que l'usuari visualitzarà a través de la web. El navegador s'encarrega de llegir les etiquetes i interpretar-les. Pel que fa a la creació d'arxius *HTML*, són fitxers plans de tal manera que amb un editor senzill es poden crear.



Figura 2.3.1.3. Icona de HTML.

2.3.2 Entorn de desenvolupament

Els entorns de desenvolupament que he utilitzat en el desenvolupament d'aquest projecte han estat:

- **Navegador web:** un navegador web és una aplicació software que permet a l'usuari recuperar i visualitzar documents d'hipertext (documents HTML), els quals són necessaris per visualitzar els nous mòduls implementats. Actualment, existeixen diversos navegadors: Internet Explorer, Mozilla Firefox, Opera, Safari, Chrome, etc. Cadascun d'ells té una interpretació pròpia del llenguatge HTML i del JavaScript, és per això que a l'hora de fer les proves de visualització dels nous mòduls s'han triat els dos més utilitzats. En concret, s'han utilitzat els navegadors:
 - Mozilla Firefox 13.0.
 - Google Chrome 20.0.
- **WinSCP:** SSH és un protocol segur de comunicació a través de la xarxa. WinSCP és un conjunt d'eines que van ser dissenyades des d'un inici per oferir una seguretat màxima i permetre l'accés remot als servidors de manera segura utilitzant el protocol SSH. Per realitzar les transferències de fitxers al servidor, s'ha utilitzat aquest software que ens permet connectar-nos al servidor a través de SSH.
- **Notepad++ :** El Notepad++ és un editor de codi font lliure que suporta diversos llenguatges de programació i funciona sota l'entorn MS Windows. Aquest projecte, basat en el component d'edició Scintilla (un component d'edició molt potent), està escrit en C++. Està sota llicència GPL.

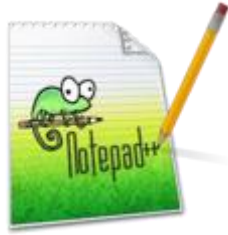


Figura 2.3.2.1. Icona de Notepad.

2.3.3 Altres eines de software utilitzades

Altres eines que s'han utilitzat en el desenvolupament d'aquest projecte són:

- **Poseidon for UML 8.0:** és una eina que permet generar diversos tipus de diagrames UML. S'ha utilitzat per crear la documentació del disseny i l'anàlisi del projecte.



Figura 2.3.3.1. Icona de Poseidon for Uml 8.0.

- **Microsoft Word:** és el processador de textos que està integrat en la suite ofimàtica de Microsoft Office.



Figura 2.3.3.2. Icona de Microsoft Word.

2.4 Requeriments del Sistema

Els requisits per al funcionament de l'aplicació són:

- ✓ Si s'accedeix a l'aplicació com a *client o usuari* és necessari qualsevol navegador amb suport per JavaScript.

- ✓ Si s'accedeix a l'aplicació com a *gestor o administrador* és necessari:
 - Tenir instal·lat Joomla! 1.5 o superior.
 - PHP 4.3.10 o superior, preferiblement 5.x.
 - MySQL 3.23 o superior.
 - El servidor web ha de tenir les funcionalitats xml i zlib.
 - Navegador web.

2.5 Estudis i Decisions

La decisió de programar utilitzant el Joomla! i el seu framework es va prendre al plantejar el projecte.

2.5.1 Què és Joomla!?

Joomla! és un sistema gestor de continguts dinàmics (CMS) que permet crear pàgines web de alta interactivitat, professionalitat i eficiència. L'administració de Joomla! està basada en la gestió online de continguts. És de gestió online perquè totes les accions que realitzen els administradors de Joomla!, ja sigui modificar, agregar o eliminar continguts es realitza exclusivament mitjançant un navegador web connectat a Internet. Això converteix a Joomla! en una eina de difusió de informació, de marketing online, de negocis per Internet, d'administració de projectes, etc.

La interfície administrativa és tan senzilla, que qualsevol persona pot administrar els seus propis continguts web sense la necessitat de tenir coneixements tècnics.

Joomla! està programat amb llenguatge PHP(Hypertext Preprocessor) y SQL (Structure Query Language). Utilitza bases de dades relacionals, més específicament MySQL.

El Joomla! utilitza el seu propi framework.

El funcionament del Joomla! es basa en dos elements principals:

- La base de dades MySQL: on es guarda tota la informació i la major part de configuració del sistema.
- Els scripts PHP: són els que executen les accions de consulta i realitzen modificacions a la base de dades convertint les dades en simples pàgines web interpretables per a navegadors.

També, existeixen uns altres tipus d'arxius que realitzen funcions en el Joomla! com ara: els arxius XML, scripts JavaScript, CSS...

2.5.2 Decisions

D'una banda, com he comentat a la introducció, el Joomla! s'adapta perfectament a les necessitats personals i a les de l'empresa client.

El component "com-property" és un aplicatiu que permet als usuaris i gestors publicar immobles pel seu lloguer o compra. Està preparat per posar en contacte clients i propietaris a través de la xarxa, sota la vigilància d'un administrador. L'aplicatiu disposa d'una interfície gràfica molt acurada, la qual permet que es puguin veure les fotos de cada immoble, el seu calendari de disponibilitat, fer reserves i consultar els preus corresponents.

Decideixo adaptar aquest component perquè puc aprofitar les funcionalitats que ja incorpora i afegir-ne de noves per tenir un producte final complint els requisits establerts.

D'altra banda, treballar amb un framework permet estalviar molt de temps en el desenvolupament d'aplicacions web. Malgrat l'existència d'altres frameworks PHP com ara: el Zend framework, Symphony, CakePHP, Drupal, etc. S'ha utilitzat el framework de Joomla! Al ser un dels més utilitzats i que a la vegada aporta una garantia de seguretat, eficiència i fiabilitat.

Les implicacions que comporta fer ús del framework de Joomla! per al desenvolupament de l'aplicació s'han detallat a l'apartat de "Disseny", tot i que a continuació hi ha una explicació d'algunes de les afectacions del framework.

Per al disseny de l'aplicació s'ha tingut en compte l'extensibilitat de Joomla! i s'ha evitat d'escriure codi innecessari si a canvi es pot estendre les classes del framework. Així s'han fet servir, entre d'altres, les següents classes:

- **JFactory:** és la classe estàtica per accedir als objectes globals i crear nous objectes.
- **JDatabase:** és la classe connectora amb la base de dades. Hi ha diverses subclasses o també conegudes com Drivers que permeten la connexió amb els diferents servidors de Base de Dades tals com MySQL o SQLserver.
- Les classes abstractes base que fa servir la implementació del patró Model Vista Controlador: **JModel, JView i JController.**
- **JDate:** és la classe d'emmagatzematge de dates.
- **JPagination:** és la classe que proveeix d'una interfície comuna per a la paginació del contingut.
- **JUser:** és la classe que controla tota la interacció de l'aplicació amb l'usuari.

La particularitat del framework de Joomla! i concretament les particularitats del desenvolupament seguint el patró de programació “Model Vista Controlador” fa que les entitats de la especificació es corresponguin directament amb classes.

2.5.3 Patró MVC i Extensió de classes

Per aprofitar les funcionalitats que proposa el framework de programació Joomla! tot el desenvolupament es realitza segons el patró de programació Model Vista Controlador. Això comporta que les classes del Model de Negoci es subdivideixen en tres grups principals:

- **Els Models:** són les classes responsables d'accedir a la capa d'emmagatzemament per arxivar, modificar i llegir dades.
- **Les Vistes:** són les classes que reben les dades del Model i les mostren a l'usuari mitjançant Template.
- **Els Controladors:** són les classes que accepten els inputs de l'usuari i s'encarreguen de donar una resposta mitjançant la Vista i/o el Model adient. És a dir, la interacció d'un usuari és rebuda per un Controlador, que demana

unes dades al Model, el qual les passa a una Vista, per tal que es mostrin de forma adient a l'usuari mitjançant un Template.

Les classes abstractes que s'extendran del framework per a l'acompliment del patró són:

- **JModel:**

<http://api.joomla.org/1.5/Joomla-Framework/Application/JModel.html>

- **JView:**

<http://api.joomla.org/1.5/Joomla-Framework/Application/JView.html>

- **JController:**

<http://api.joomla.org/1.5/Joomla-Framework/Application/JController.html>

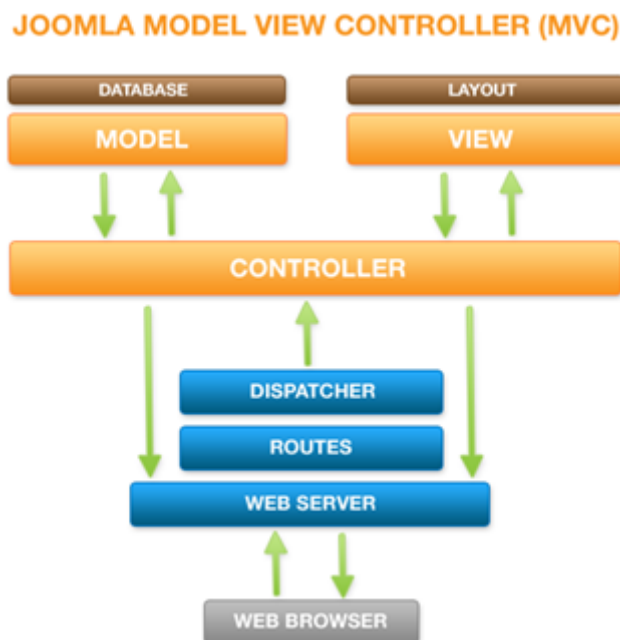


Figura 2.3.6.1 Model Vista Controlador de Joomla!.

3 ESTUDI DE VIABILITAT

Una vegada definits els objectius i analitzats els requeriments, cal fer un estudi de viabilitat del projecte. Per decidir la viabilitat del projecte s'ha tingut en compte:

- El cost d'un servidor web i d'una base de dades MySQL, el cost dels quals és zero. Com que l'aplicació ja funciona en l'actualitat, no és necessari l'adquisició de cap servei addicional.
- Per definir els requeriments amb el client es necessiten 8 hores.
- Pel disseny i la implementació del projecte es calcula que caldran aproximadament 450 hores.

Una vegada analitzats aquests punts amb el client i aprovats per aquest, el projecte s'ha considerat viable i ha començat la seva execució.

4 METODOLOGIA

Una metodologia és un seguit d'etapes i procediments utilitzats durant el procés de desenvolupament d'un sistema informàtic. En el cas d'aquest projecte, s'han utilitzat la combinació de dos metodologies: Procés Unificat de Rational (RUP) i la eXtreme Programming (XP). A continuació, detallaré les característiques principals de cada metodologia que s'han utilitzat en aquest projecte.

4.1 Procés Unificat de Rational (RUP)

És una de les metodologies més esteses i conegudes per la seva àmplia difusió comercial. Es pot estudiar com una metodologia representativa de tipus clàssic. Va ser definit pels creadors de l'UML unificant els mètodes de Jacobson, Booch i Rumbaugh.

Els principis d'actuació claus del projecte referents a aquesta metodologia consisteixen en:

- Basat en components.
- Aquests components al seu torn estan connectats entre si a través d'interfícies.
- Utilitza l'UML com notació bàsica.
- Dirigit per casos d'ús.
- Cicle de vida iteratiu i incremental.
- Molta documentació.
- Proves unitàries

4.2 eXtreme Programming (XP)

Aquest tipus de metodologia pretén que el desenvolupament d'un projecte de programari sigui un desenvolupament àgil, però disciplinat, i aporti solucions senzilles. *XP* té un enfocament adaptatiu, ja que la planificació del projecte progressa a mida que sorgeixen els canvis.

Els principis d'actuació claus del projecte referents a aquesta metodologia consisteixen en:

- Reduir els cicles de desenvolupament.
- Involucrar al client des de l'inici fins al final de cada cicle.
- Centrar-se cada vegada en un problema molt concret en el moment just.

- Solucionar els problemes de manera immediata i no arrossegar-los al llarg de tot el projecte, abans de començar un nou mòdul del projecte.

5 ANÀLISI I DISSENY DEL SISTEMA

5.1 Joomla!

Una vegada realitzada la introducció del Joomla! a l'apartat [Què és el Joomla!?](#) cal entrar en detall de com funciona i com s'estructura per tal de poder fer les millores i aprofitar el codi existent. Després, cal fer el mateix amb el component "com-property", per tal de poder dissenyar els mòduls de Clients, Preus i Facturació.

5.1.1 Estructura de desenvolupament

Des del punt de vista de desenvolupament, es pot entendre també l'estructura de Joomla! conformada per:

- **El sistema o core i continguts:**

Els continguts que hi ha al lloc web s'emmagatzemen en Joomla! de forma independent, fonamentalment en una base de dades, encara que també algunes vegades, en arxius dins de carpetes (per exemple, carpetes d'imatges).

El sistema o core és el conjunt d'aplicacions / arxius responsable del funcionament de Joomla!, per exemple d'enviar les instruccions per visualitzar certs continguts sol·licitats per un usuari en fer clic sobre un enllaç. El nucli dona instruccions a instàncies superiors, que són les que compleixen aquestes instruccions. En certa manera és com si Joomla! estigués organitzat en jerarquies, i la jerarquia més bàsica, la de més baix nivell (la que tracta directament amb el servidor de pàgines web) és el nucli.

- **Les plantilles:**

Són patrons gràfics que utilitzen codi HTML, CSS i JavaScript per mostrar disposicions d'elements com imatges, tipus de lletra i configuracions de color que modifiquen l'aspecte d'un lloc web. Les plantilles creen espais amb format els quals es veuran després en els mòduls, així com els plugins i els components que es van configurar prèviament en l'administració.

- **Les extensions** (mòduls, components, plugins):
 - **Components:** Són paquets complets programats per a complir funcions específiques com crear un fòrum de discussió, una galeria d'imatges o un sistema de formularis de contacte.
 - **Mòduls:** Són paquets pensats per treballar en conjunt amb els articles, generalment aconsegueixen objectius menys ambiciosos que els components però són més flexibles, ja que es poden utilitzar mòduls propis de l'aplicació, usar-ne d'altres desenvolupats per tercers o bé, crear mòduls personalitzats amb HTML, CSS i JavaScript.
 - **Plugins:** Són desenvolupaments a mida, que compleixen funcions bastant pràctiques.

5.1.2 Funcionament

Joomla! implementa l'arquitectura Model Vista Controlador de la següent manera:

- Inicialitza el controlador que registra els esdeveniments (o tasques) i les funcions que els capturaran.
- Executa la tasca inicial. Aquesta tasca inicialitza la vista i el model, els quals són units posteriorment.
- Executa la funció display, que construeix la interfície.
- Un cop s'ha construït la interfície i aquesta s'ha mostrat per pantalla, s'espera a que l'usuari premi un botó que llanci un esdeveniment.
- El llançament d'aquest esdeveniment inicialitza la tasca corresponent, la qual és capturada pel controlador.
- El controlador executa la funció associada a aquesta tasca. Aquesta funció ataca el model i normalment, un cop s'ha finalitzat l'execució de la funció, es recarrega la pàgina amb el mateix controlador o bé, un de nou, tornant a repetir el mateix procés.

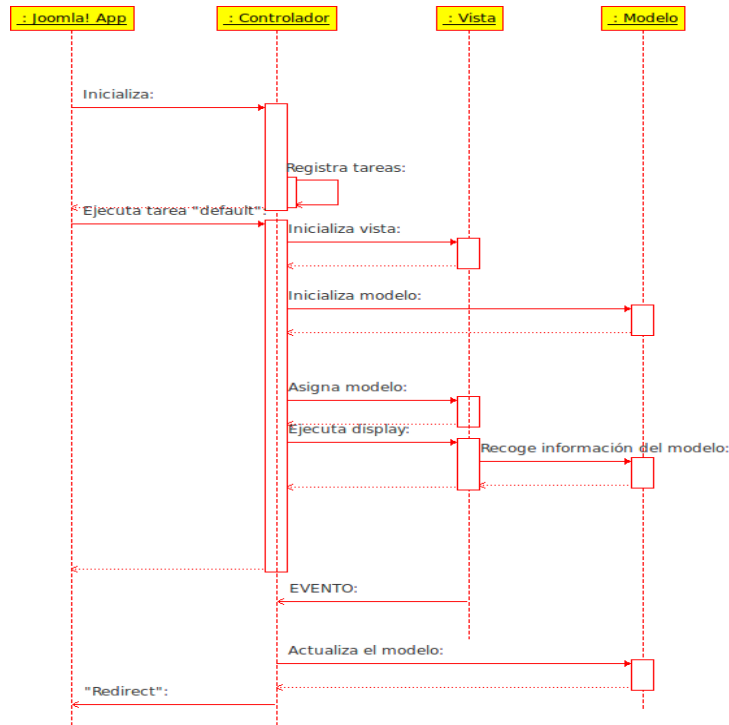


Figura 5.1.2.1 Diagrama seqüència Model Vista Controlador de Joomla!.

5.2 Component “com-property”

El component “com-property” és un component per Joomla! per a la gestió d’immobles. El component està programat i dissenyat per a una immobiliària amb les funcionalitats estàndard perquè es pugui adaptar a cada cas.

Per tal d’afegir les noves funcionalitats, cal fer un estudi previ de com s’estructura aquest component, quines taules utilitza i de com està dissenyat.

Cal afegir que d’aquest component, no hi ha cap tipus de documentació a la seva pàgina web², i per tant, ha estat una feina personal, el fet d’aprendre i entendre el funcionament del component.

5.2.1 Funcionalitats

Aquestes són les funcionalitats que té el component “com-property”:

- Llistats d’immobles per categories i tipus.

² www.com-property.com

- Veure detalls de cases i apartaments.
- Sistema de reserves dels immobles.
- Alta, baixa i modificació d'habitatges.
- Comentaris. Els usuaris poden deixar comentaris de les cases.
- Puntuació dels immobles.
- Contacte entre l'arrendador i l'arrendatari.
- Càlcul de preus.

A petició del client, les funcionalitats de *comentaris* i de *puntuació* dels immobles no han estat habilitades.

Representació gràfica de les taules, els seus camps i les seves relacions.

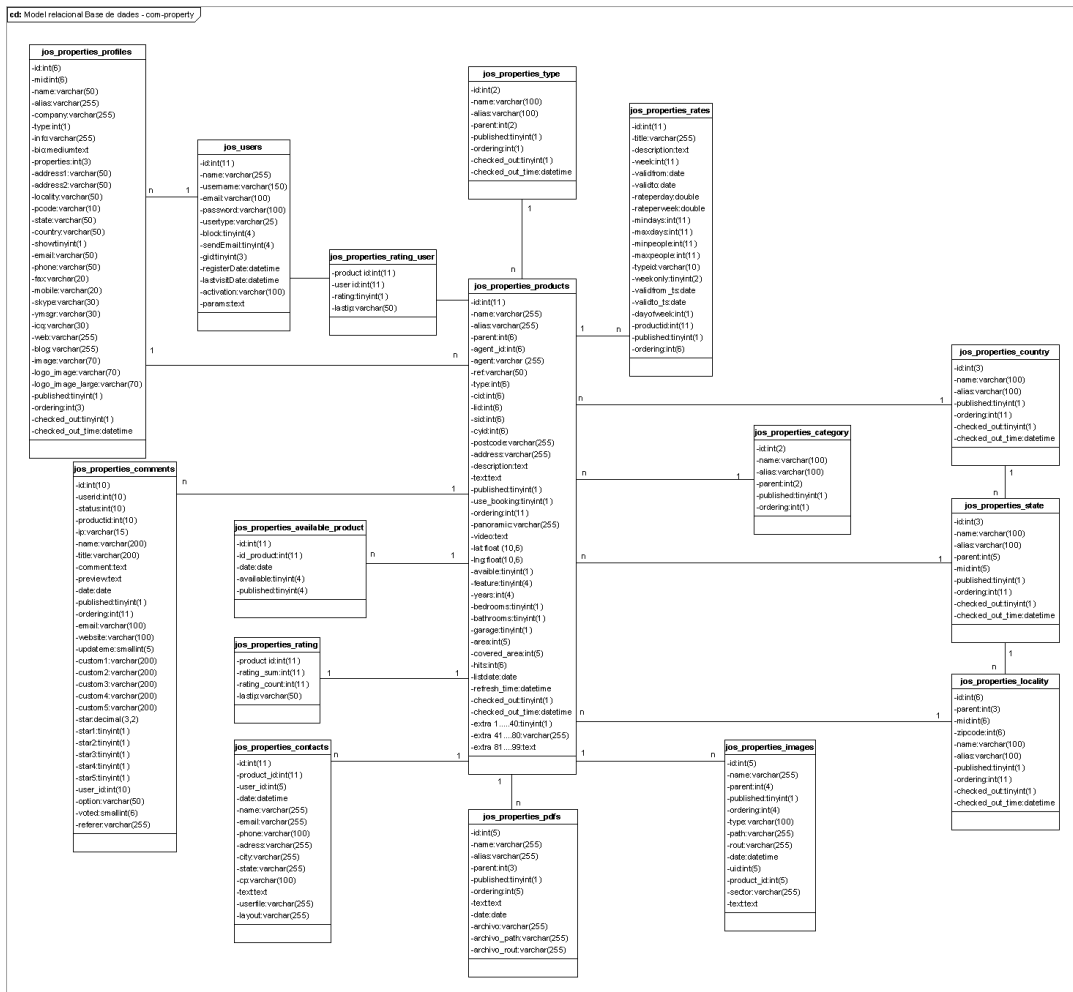


Figura 5.2.2.1 Model Relacional de base de dades del component "com-property".

Com es pot observar al model relacional de base de dades, la taula de productes és la més important, i a partir d'aquesta taula, van sortint les relacions cap a les altres.

5.2.2 Estructura de directoris i fitxers

En l'estructura de directoris i fitxers del component "com-property" cal fer una diferenciació entre el frontend i el backend.

▪ Frontend

Cal accedir a la carpeta /components/com_properties

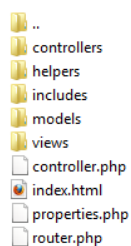


Figura 5.2.3.1 Estructura de Directoris i Fitxers frontend.

▪ Backend

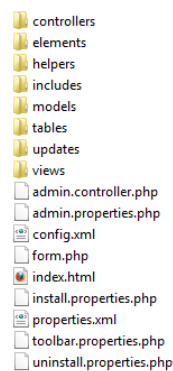


Figura 5.2.3.2 Estructura de Directoris i Fitxers backend.

En els dos casos, l'estructura de carpetes és la mateixa:

- **Carpeta controllers:** Hi ha els arxius PHP encarregats d'escoltar totes les peticions, executar la lògica de l'aplicació i mostrar la vista apropiada per a cada ocasió.
- **Carpeta models:** Hi ha els arxius PHP encarregats de la comunicació cap a la base de dades.
- **Carpeta views:** Hi ha els arxius PHP encarregats de representar tot el que visualitza per la pantalla.

En resum, si es vol fer una ampliació i afegir noves funcionalitats, s'ha de crear nous arxius a les carpetes esmentades anteriorment. A més a més, cal tenir en compte si aquesta nova funcionalitat s'hi podrà accedir des del frontend o del backend.

5.3 Mòdul de Clients

El mòdul de Clients ha de permetre al gestor l'administració dels clients de la seva empresa. En concret, l'alta, la modificació o la baixa d'un client amb tota la informació corresponent a aquest.

5.3.1 Objectius

L'objectiu del Mòdul de Clients és facilitar a l'administrador de finques, una base de dades amb la informació detallada de cada client, fent constar totes les dades i eines necessàries per a la seva gestió.

Per assolir aquest objectiu, s'ha agafat com a punt de partida una base de dades, en format Microsoft Access, de clients que l'administrador de finques disposava, tal com es pot observar a la següent figura.

Alta Clients

Nom

1er Cognom

2on Cognom

Nif Principal:

Nie Secundari:

Nie Secundari:

Passaport Principal:

Passaport Secu

Direcció

Codi postal

Ciutat

Pais

Telf. part.

Telf. mobil

Nº de fax

Correu electrònic

Comentaris

Figura 5.3.1.1 Pantalla d'administració de clients en format Microsoft Access.

5.3.2 Model d'Entitat-Relació

Per tal de realitzar el mòdul de clients s'ha interactuat amb les entitats que el Joomla! té configurades i les del component "com-property". En aquest cas, tenim 3 entitats que estan relacionades entre sí:

- **Usuaris:** Taula jos_users del Joomla! és la taula d'usuaris del sistema. Aquesta taula hi ha emmagatzemada tota la informació referent a l'usuari i permet fer el seu login al frontend de l'aplicació.
- **Perfils:** Taula jos_properties_profiles. És la taula de perfils del mòdul "com-property". Aquesta taula solament hi ha registrats els administradors de les finques, en definitiva, els que gestionen l'alta, la baixa i la modificació de les propietats i tots els seus atributs.
- **Clients.** Taula jos_properties_clients. És la taula del clients de l'aplicació.

5.3.2.1 Diagrama Entitat-Relació

El Diagrama Entitat-Relació entre les entitats “CLIENTS”, “USUARIS” i “PERFILS” és el següent:



Figura 5.3.2.1 Diagrama Entitat – Relació del mòdul de clients.

5.3.2.2 Model Relacional

La definició de les taules és la següent, tenint en compte que els valors remarcats amb negreta són **claus primàries**, els valors amb cursiva són *claus foranes* i els valors que estan entre parèntesi és la taula origen dels valors:

La taula *JOS_PROPERTIES_CLIENTS* guarda la informació dels clients:

jos_properties_clients		
Nom	Descripció	Tipus
id	Codi d'identificació del client	int(11)
nif	Número Identificació	varchar(50)
name	Nom del client	varchar(255)
alias	Sobrenom del client	varchar(50)
email	Correu electrònic del client	varchar(100)
surname_1	Primer cognom client	varchar(255)
surname_2	Segon cognom client	varchar(255)
address	Adreça postal client	varchar(255)
pcode	Codi postal de la població	varchar(10)
locality	Localitat o Població	varchar(255)
state	Estat	varchar(255)
country	País	varchar(255)
phone	Telèfon	varchar(30)
mobile	Telèfon Mòbil	varchar(255)
fax	Fax	varchar(30)
ccc	Nº de compte bancari	varchar(20)
bank	Nom del banc	varchar(255)
published	Indicador de publicat o no publicat	tinyint(1)
ordering	Ordre	int(3)
image	Nom de la imatge relacionada	varchar(70)
<i>mid</i>	Codi d'identificació de l'usuari. (JOS_USERS.ID)	int(3)
<i>gid</i>	Indicador de grup (JOS_GROUPS.ID)	tinyint(3)

La taula *JOS_USERS* guarda la informació referent als usuaris del sistema:

jos_users		
Nom	Descripció	Tipus
id	Identificador únic d'usuari	int(11)
name	Nom de l'usuari	varchar(255)
username	Nom d'usuari	varchar(150)
email	Email	varchar(100)
password	Contrasenya	varchar(100)
usertype	Tipus d'usuari	varchar(25)
block	Indicador de bloqueig	tinyint(4)
sendEmail	Indicador de enviar email	tinyint(4)
<i>gid</i>	Indicador de grup (JOS_GROUPS.ID)	tinyint(3)
registerDate	Data de registre	datetime
lastvisitDate	Data última connexió	datetime
activation	Indicador d'activat	varchar(100)
params	Paràmetres	text

La taula JOS_PROFILES guarda la informació referent als perfils:

jos_properties_profiles		
Nom	Descripció	Tipus
id	Identificador únic de perfil	int(6)
<i>mid</i>	Codi d'identificació de l'usuari. (JOS_USERS.ID)	int(6)
name	Nom del perfil	varchar(50)
alias	Sobrenom del perfil	varchar(255)
company	Nom de la Companyia	varchar(255)
type	Tipus	int(1)
info	Informació del perfil	varchar(255)
bio	Biografia	mediumtext
properties	Número de propietats	int(3)
address1	Adreça postal 1	varchar(50)
address2	Adreça postal 1	varchar(50)
locality	Població / Municipi	varchar(50)
pcode	Codi Postal	varchar(10)
state	Estat	varchar(50)
country	País	varchar(50)
show	Indicador de mostrar	tinyint(1)
email	Correu electrònic	varchar(50)
phone	Telèfon	varchar(20)
fax	Fax	varchar(20)
mobile	Mòbil	varchar(20)
skype	Usuari Skype	varchar(30)
ymsg	Usuari Messenger	varchar(30)
icq	Usuari ICQ	varchar(30)
web	Pàgina web	varchar(255)
blog	Blog personal	varchar(255)
image	Nom imatge	varchar(70)
logo_image	Logo de l'empresa	varchar(70)
logo_image_large	Logo de l'empresa	varchar(70)
published	Indicador de publicació	tinyint(1)
ordering	Ordre	int(3)
checked_out	Indicador hora sortida	tinyint(1)
checked_out_time	Hora sortida	datetime

Representació gràfica de les taules, els seus camps i les seves relacions.

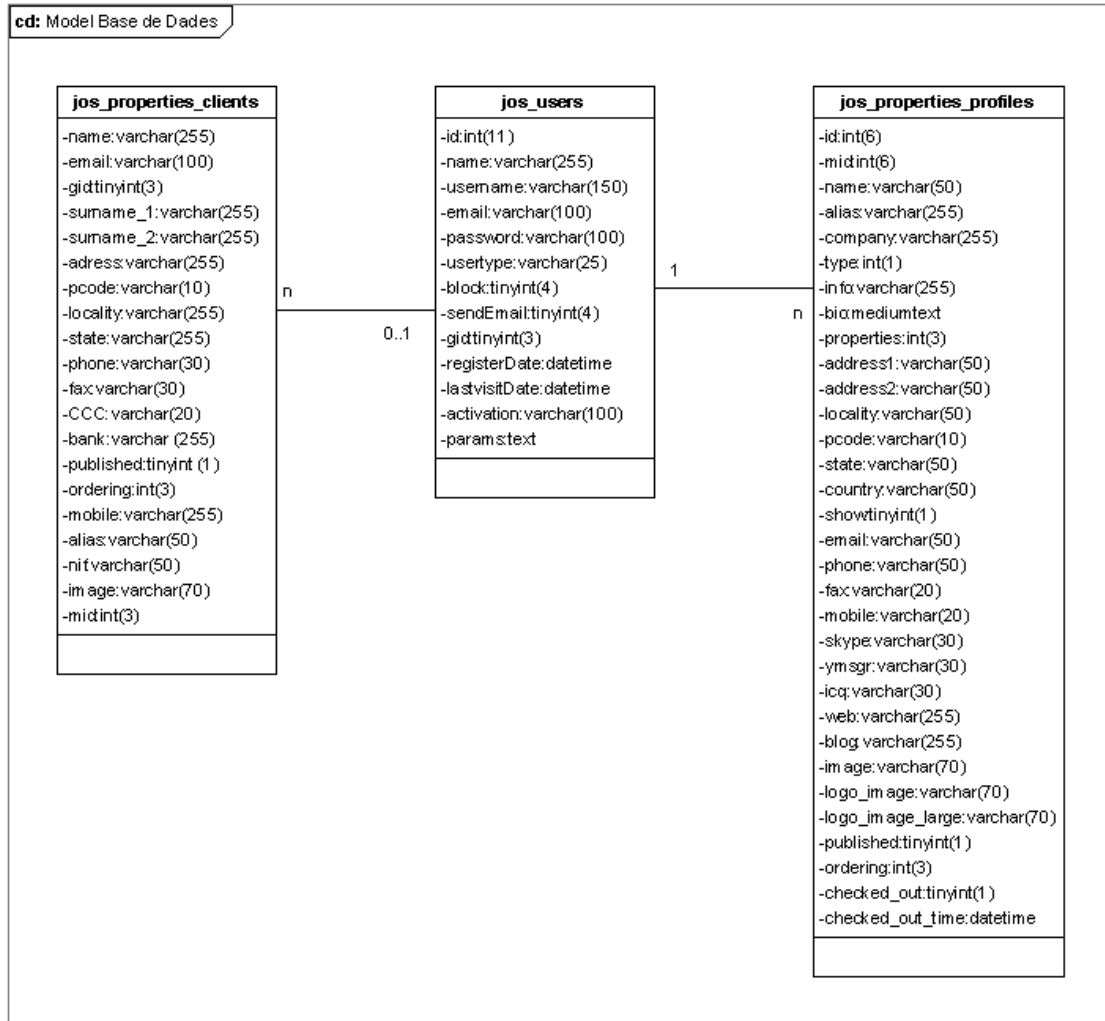


Figura 5.3.2.2 Model Relacional de base de dades – Relació del mòdul de clients.

5.3.3 Descripció general

Aquest mòdul permet:

- Afegir, modificar i eliminar clients
- Assignar usuaris a clients
- Llistar clients

5.3.4 Diagrama de casos d'ús usuari Gestor

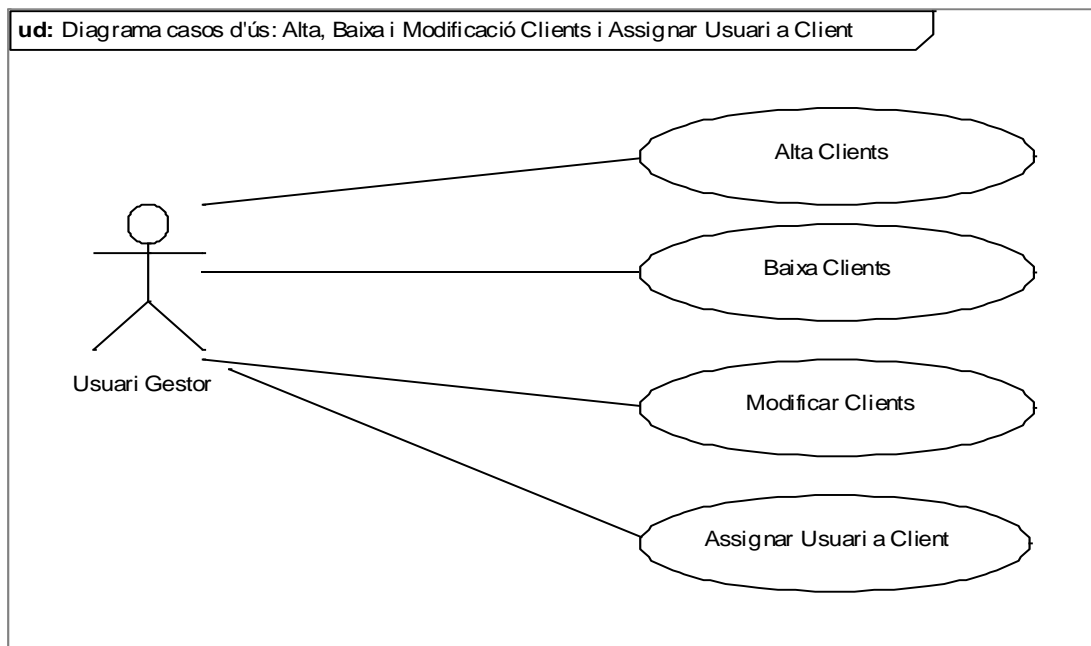


Figura 5.3.4.1 Diagrama casos d'ús – Alta, baixa i modificació de clients i assignar usuari a client.

5.3.5 Casos d'ús

5.3.5.1 Alta Client

Cas d'ús: Alta Client	
Funcionalitat:	Enregistrar un nou client al sistema.
Descripció:	S'introdueixen totes les dades disponibles del client.
Actors:	Usuari Gestor.
Casos d'ús relacionats:	
Precondició:	L'usuari gestor s'ha identificat al sistema. El client no existeix al sistema.
Flux:	Entrar dades nou client. Assignar al client un usuari. Si l'usuari no existeix. Crear un nou usuari. Altrament.

	Assignar al client un usuari. Afegim el client a la llista de clients.
Postcondició:	El client ha de ser enregistrat correctament al sistema.

5.3.5.2 *Baixa Client*

	Cas d'ús: Baixa Client
Funcionalitat:	Eliminar el client del sistema.
Descripció:	Eliminar de la base de dades un client en concret.
Actors:	Usuari Gestor.
Casos d'ús relacionats:	
Precondició:	L'usuari gestor s'ha identificat al sistema. El client existeix al sistema.
Flux:	Entrar dades del client. Seleccionar el client a suprimir. Eliminar el client de la llista de clients.
Postcondició:	El client és eliminat correctament del sistema.

5.3.5.3 *Modificació Client*

	Cas d'ús: Modificació Client
Funcionalitat:	Modificar el client del sistema.
Descripció:	Modificar els atributs del client del sistema.
Actors:	Usuari Gestor.
Casos d'ús relacionats:	
Precondició:	L'usuari gestor s'ha identificat al sistema. El client existeix al sistema.
Flux:	<ol style="list-style-type: none"> 1. Entrar dades del client. 2. Seleccionar el client. 3. Modificar les dades del client.

	4. Guardar el client a la llista de clients.
Postcondició:	El client és modificat correctament del sistema.

5.3.5.4 Assignar Usuari a Client

	Cas d'ús: Assignar Usuari a Client
Funcionalitat:	Assignar un usuari registrat a un client.
Descripció:	Relacionar un usuari a un client. Un client pot o no tenir un usuari assignat.
Actors:	Usuari Gestor.
Casos d'ús relacionats:	
Precondició:	L'usuari gestor s'ha identificat al sistema. El client existeix al sistema. L'usuari existeix al sistema.
Flux:	<ol style="list-style-type: none"> 1. Entrar dades del client 2. Seleccionar el client. 3. Assignar un usuari Si no existeix l'usuari. Afegir l'usuari al sistema. Altrament. Assignar-lo. 4. Guardar el client a la llista de clients.
Postcondició:	El client és assignat a un usuari.

5.3.5.5 Llistar Clients

	Cas d'ús: Assignar Usuari a Client
Funcionalitat:	Llistar els clients en el sistema.
Descripció:	Llistar per pantalla tots els clients enregistrats al sistema. A través del seu id de client s'accedeix al detall.
Actors:	Usuari gestor.

Casos d'ús relacionats:	
Precondició:	L'usuari gestor s'ha identificat al sistema. El client existeix al sistema.
Flux:	<ol style="list-style-type: none"> 1. Seleccionar els clients. 2. Mostrar per pantalla les dades principals.
Postcondició:	Llistat de clients mostrats.

5.3.6 Interfícies d'usuari

5.3.6.1 Alta de Clients

Una vegada s'ha accedit a l'aplicació "com-property", cal entrar a la pestanya "Clients". A continuació, cal clicar sobre "Nou".

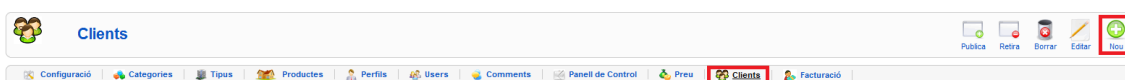


Figura 5.3.6.1.1 Imatge de l'aplicatiu. Nou client.

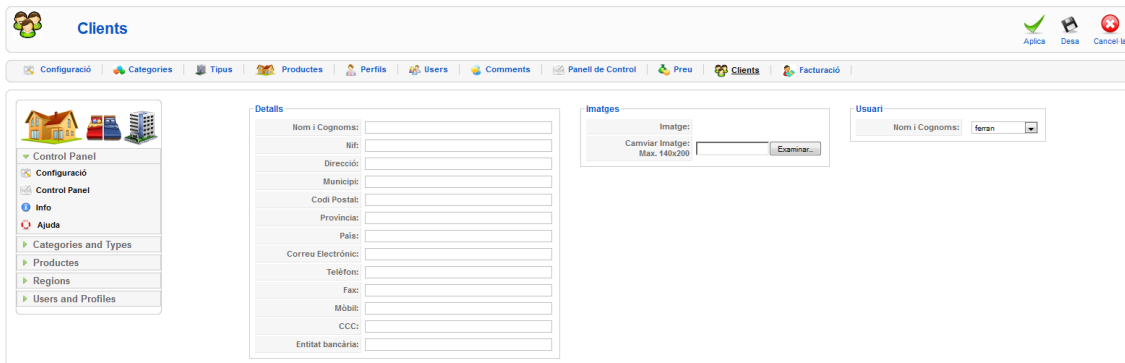


Figura 5.3.6.2.2 Imatge de l'aplicatiu. Entrar dades nou client.

Una vegada s'ha accedit a aquesta pantalla, s'emplenen els atributs corresponents i ja s'hi poden desar els canvis fets.

5.3.6.2 Baixa de Clients

Una vegada s'ha accedit a l'aplicació "com-property", cal entrar a la pestanya "Clients". A continuació, s'ha de seleccionar els diferents clients a eliminar i clicar sobre "Borrar". Els clients s'esborren de la taula de clients.

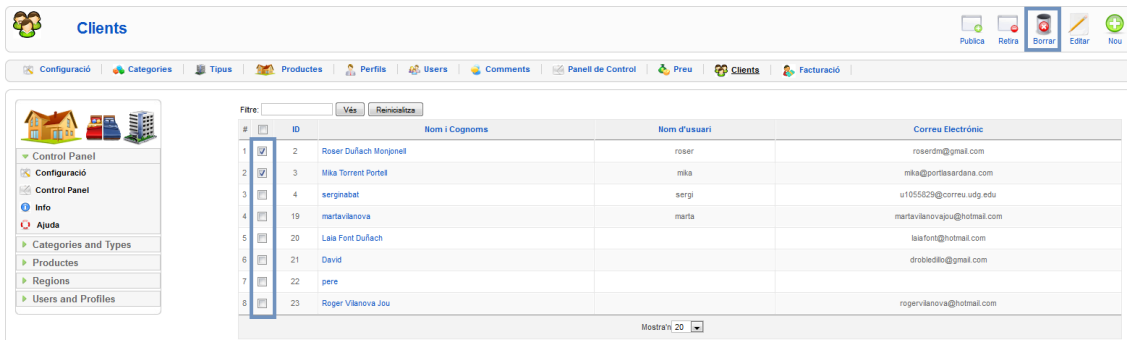


Figura 5.3.6.2.1 Imatge de l'aplicatiu. Esborrar client.

5.3.6.3 Modificar Clients

Una vegada s'ha accedit a l'aplicació "com-property", cal entrar a la pestanya "Clients". Després, s'ha de seleccionar un client de la llista de clients a modificar. La modificació es pot realitzar de dues formes diferents:

- Clicant directament sobre el nom del client per accedir al seu detall
- Marcant el registre amb el "check" i clicar sobre "Editar".

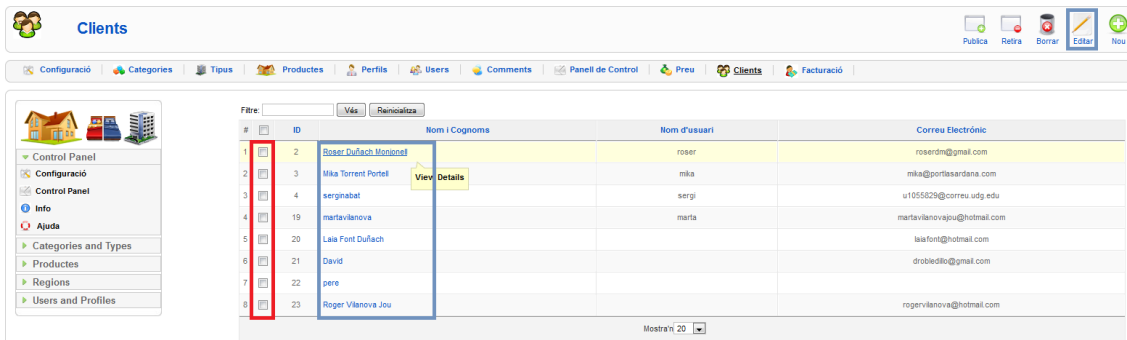


Figura 5.3.6.3.1 Imatge de l'aplicatiu. Seleccionar client a editar.

En el dos casos s'accedeix a la mateixa pantalla per modificar, la qual es mostra a continuació. Una vegada s'hi ha accedit, es poden modificar els atributs desitjats i desar-los.

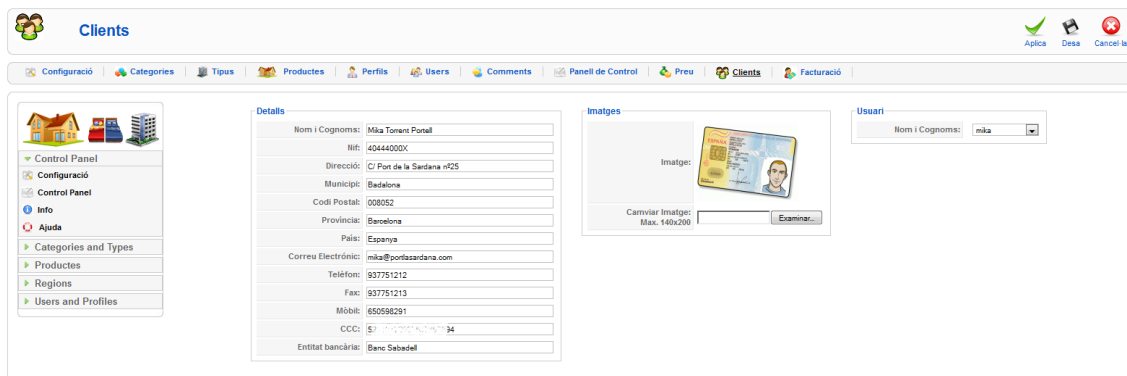


Figura 5.3.6.3.2 Imatge de l'aplicatiu. Client a editar.

5.4 Mòdul de Preus

El mòdul de preus és un mòdul necessari per poder realitzar la facturació dels clients. Per això, s'ha desenvolupat aquest mòdul, el qual consisteix en una taula on s'hi emmagatzema tota la informació dels preus i del tipus de producte.

5.4.1 Objectius

L'objectiu d'aquest mòdul és facilitar i centralitzar tota la informació dels diferents preus de les propietats immobles en una taula a l'administrador de finques.

Un requeriment del client, era que el preu de l'habitatge anés lligat directament amb el tipus d'aquest. En altres paraules, per a cada tipus de producte i depenent de l'època de l'any, el preu varia. En aquest cas, el component "com-property" no estava adaptat per les necessitats del client, és per això que s'ha dissenyat. En concret, el component "com-property" relacionava el preu amb el producte, sense tenir en compte el seu tipus.

5.4.2 Model d'Entitat-Relació

Per tal de realitzar el mòdul de preus s'ha interactuat amb les entitats del component "com-property". En aquest cas, tenim 3 entitats que han d'estar relacionades entre sí.

- **Preus:** Taula jos_properties_price. És la taula on hi ha enregistrat els imports de cada propietat per tipus de producte.
- **Tipus:** Taula jos_properties_type. És la taula de tipus de propietats del mòdul “com-property”.
- **Productes:** Taula jos_properties_products. És la taula on s’hi emmagatzema tots els productes (immobles de l’aplicació). Pertany al mòdul “com-property”.

5.4.2.1 Diagrama Entitat-Relació

El Diagrama Entitat-Relació entre les entitats “PREU”, “TIPUS” i “PRODUCTE” és el següent:



Figura 5.4.2.1.1 Diagrama Entitat-Relació del Mòdul de Clients.

5.4.2.2 Model Relacional

La definició de les taules és la següent, tenint en compte que els valors remarcats amb negreta són **claus primàries**, els valors amb cursiva són *claus foranes* i els valors que estan entre parèntesi és la taula origen dels valors.

La taula JOS_PROPERTIES_PRICE guarda la informació dels preus d’un tipus entre dues dates.

jos_properties_price		
Nom	Descripció	Tipus
id	Identificador únic de preu	int(11)
description	Descripció del preu	text
import	Import	decimal(15,2)
validfrom	Data inici	date
validto	Data fi	date
rateperday	Preu per dia	decimal(15,3)
rateperweek	Preu per setmana	decimal(15,3)
published	Indicador de publicació	tinyint(1)
ordering	Indicador d'ordre	int(6)
<i>cid</i>	Tipus d'habitatge. (JOS_PROPERTIES_TYPE.ID)	int(2)

La taula JOS_PROPERTIES_TYPE guarda la informació dels tipus d'habitatge o immobles.

joseproperties_type		
Nom	Descripció	Tipus
id	Identificador únic de Tipus	int(2)
name	Nom del tipus	varchar(100)
alias	Sobrenom del tipus	varchar(100)
<i>parent</i>	Identificador del tipus pare (JOS_PROPERTIES_TYPE.ID)	int(2)
published	Indicador de publicació	tinyint(1)
ordering	Ordre	int(1)
checked_out	Indicador de hora de sortida	tinyint(1)
checked_out_time	Hora de sortida	datetime

La taula JOS_PROPERTIES_PRODUCTS guarda la informació referent als productes. És la taula més important del sistema.

joseproperties_products		
Nom	Descripció	Tipus
id	Identificació únic de Productes	int(11)
name	Nom del producte	varchar(255)
alias	Sobrenom del producte	varchar(255)
<i>parent</i>	Producte pare. (JOS_PROPERTIES_PRODUCTS.ID)	int(6)
<i>agent_id</i>	Nº agent assignat. (JOS_PROPERTIES_PROFILES.ID)	int(6)
agent	Nom de l'agent assignat	varchar(255)
ref	Referència	varchar(50)
<i>type</i>	Identificador de tipus. (JOS_PROPERTIES_TYPE.ID)	int(6)
<i>cid</i>	Identificador de categoria. (JOS_PROPERTIES_CATEGORY.ID)	int(6)
<i>lid</i>	Identificador de Localitat. (JOS_PROPERTIES_LOCALITY.ID)	int(6)
<i>sid</i>	Identificador de Estat. (JOS_PROPERTIES_STATE.ID)	int(6)
<i>cyid</i>	Identificador de País. (JOS_PROPERTIES_COUNTRY.ID)	int(6)
postcode	Codi Postal	varchar(255)
address	Adreça postal	varchar(255)

description	Descripció	text
text	Auxiliar de descripció	text
price	Preu	decimal(15,2)
published	Indicador publicat	tinyint(1)
use_booking	Indicador de ús de reserves	tinyint(1)
ordering	Ordre	int(11)
panoramic	Imatge panoràmica	varchar(255)
video	Video	text
lat	Latitud	float(10,6)
lng	Longitud	float(10,6)
available	Indicador disponible	tinyint(1)
featured	Destacat	tinyint(4)
years	Nº d'anys	int(4)
bedrooms	Nº d'habitacions	tinyint(1)
bathrooms	Nº de Banys	tinyint(1)
garage	Garatge	tinyint(1)
area	Area total	int(5)
covered_area	Area total coberta	int(5)
hits	Nº de vots	int(6)
listdate	Dia entrada a la llista	date
refresh_time	Data última actualització	datetime
checked_out	Indicador hora sortida	tinyint(1)
checked_out_time	Hora sortida	datetime
extra1...extra40	Indicadors extrems	tinyint(1)
extra41...extra80	Camps auxiliars	varchar(255)
extra81...extra90	Camps auxiliars	text

Representació gràfica de les taules, els seus camps i les seves relacions.

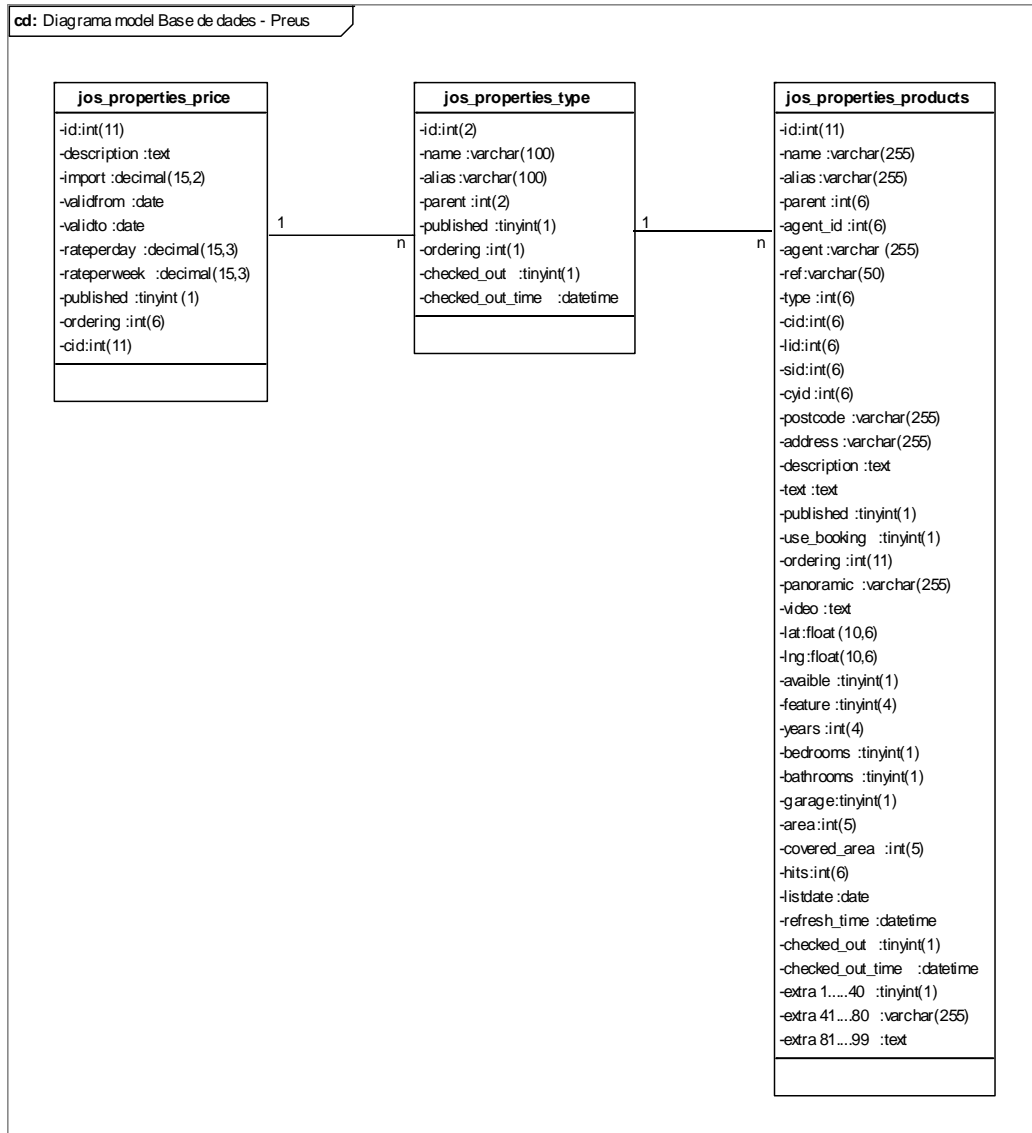


Figura 5.3.2.2 Model Relacional de base de dades – Relació del mòdul de preus.

5.4.3 Descripció general

Aquest mòdul permetrà:

- Afegir, modificar i eliminar preus

5.4.4 Diagrama de casos d'ús usuari Gestor

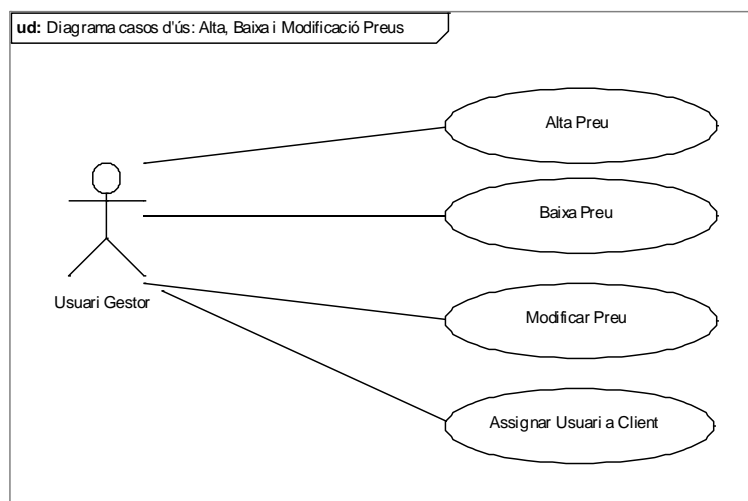


Figura 5.3.4.1 Diagrama de casos d'ús – Alta, baixa i modificació de preus.

5.4.5 Casos d'ús

5.4.5.1 Alta Preu

	Cas d'ús: Alta Preu
Funcionalitat:	Afegir un preu a un tipus d'habitatge.
Descripció:	S'introdueixen totes les dades per tal de guardar un preu, depenent del tipus d'habitatge i de dues dates seleccionades.
Actors:	Usuari Gestor.
Casos d'ús relacionats:	
Precondició:	L'usuari gestor s'ha identificat al sistema. La data d'inici ha d'ésser més petita que la data de fi.
Flux:	<ol style="list-style-type: none"> 1. Entrar dades. 2. Assignar un preu a una tipologia de l'habitatge. <ol style="list-style-type: none"> 2.1. Si no existeix la tipologia. <ol style="list-style-type: none"> 2.1.1 Afegir la tipologia. 3 Afegir el preu a la llista de preus.
Postcondició	Preu ha estat enregistrat correctament al sistema.

5.4.5.2 *Baixa Preu*

	Cas d'ús: Baixa Preu
Funcionalitat:	Eliminar el preu del sistema.
Descripció:	Eliminar de la base de dades el registre corresponent.
Actors:	Usuari Gestor.
Casos d'ús relacionats:	
Precondició:	L'usuari gestor s'ha identificat al sistema. La data d'inici ha d'ésser més petita que la data de fi.
Flux:	<ol style="list-style-type: none">1. Seleccionar el preu a esborrar.2. Eliminar el preu de la llista de preus.
Postcondició:	El preu ha estat eliminat correctament del sistema.

5.4.5.3 *Modificació Preu*

	Cas d'ús: Modificació Preu
Funcionalitat:	Modificar el preu al sistema.
Descripció:	Modificar els atributs del preu al sistema.
Actors:	Usuari Gestor.
Casos d'ús relacionats:	
Precondició:	L'usuari gestor s'ha identificat al sistema. La data d'inici ha d'ésser més petita que la data de fi.
Flux:	<ol style="list-style-type: none">1. Seleccionar el preu a modificar.2. Editar les dades.<ol style="list-style-type: none">2.1. Si no existeix la tipologia.<ol style="list-style-type: none">2.1.1 Afegir la tipologia.2.2 Altrament.3. Guardar el preu a la llista de preus.
Postcondició:	El preu ha estat modificat correctament al sistema.

5.4.6 Interfícies d'usuari

5.4.6.1 Alta Preu

Un cop s'ha accedit a l'aplicació "com-property", cal entrar a la pestanya "Preus". Després, cal clicar sobre "Nou".

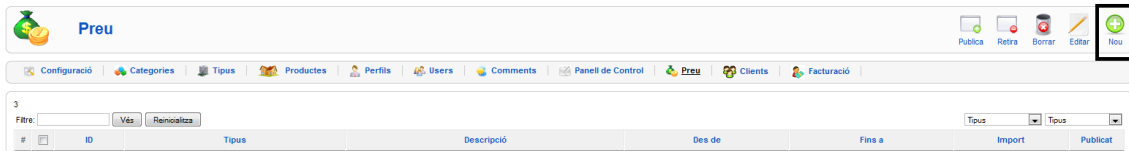


Figura 5.4.6.1.1 Imatge de l'aplicatiu. Nou preu.

Es selecciona un tipus d'habitatge i s'entra l'import i les setmanes.

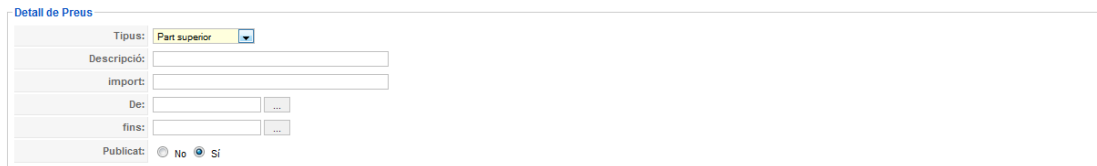
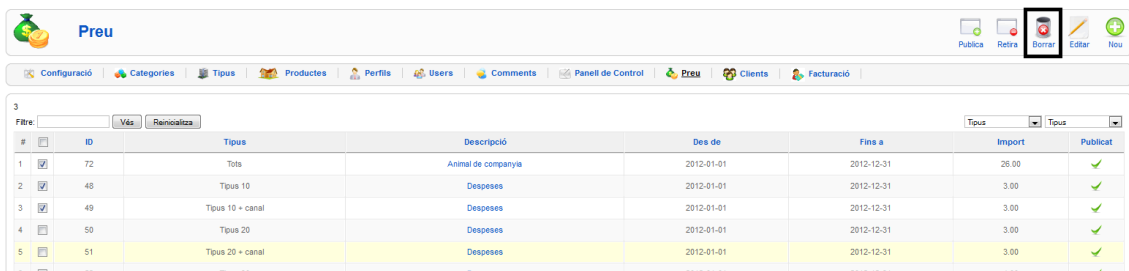
The screenshot shows the 'Detall de Preus' form. It contains several input fields: 'Tipus' (dropdown menu), 'Descripció' (text input), 'import' (text input), 'De' (date input), 'fins' (date input), and 'Publicat' (radio buttons for 'No' and 'Sí').

Figura 5.4.6.1.2 Imatge de l'aplicatiu. Pantalla entrar nou preu.

5.4.6.2 Baixa Preu

Una vegada s'ha accedit a l'aplicació "com-property", cal entrar a la pestanya "Preus". A continuació, s'ha de seleccionar els diferents preus a eliminar i clicar sobre "Borrar".

The screenshot shows the application interface with a table of preus. The 'Borrar' button in the top navigation bar is highlighted with a red box. The table below has the following data:

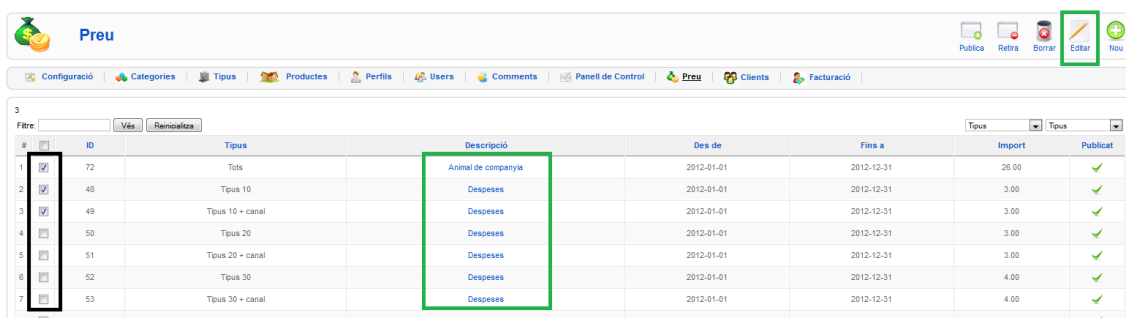
#	ID	Tipus	Descripció	Des de	Fins a	Import	Publicat
1	72	Tots	Animal de companyia	2012-01-01	2012-12-31	26.00	✓
2	48	Tipus 10	Despeses	2012-01-01	2012-12-31	3.00	✓
3	49	Tipus 10 + canal	Despeses	2012-01-01	2012-12-31	3.00	✓
4	50	Tipus 20	Despeses	2012-01-01	2012-12-31	3.00	✓
5	51	Tipus 20 + canal	Despeses	2012-01-01	2012-12-31	3.00	✓

Figura 5.4.6.2.1 Imatge de l'aplicatiu. Pantalla eliminar preu.

5.4.6.3 Modificar Preu

Una vegada s'ha accedit a l'aplicació "com-property", cal entrar a la pestanya "Preus". Després, s'ha de seleccionar un preu de la llista de preus. La modificació es pot realitzar de dues formes diferents:

- Clicant directament sobre la descripció del preu per accedir al seu detall
- Marcant el registre amb el "check", i clicar sobre "Editar".



#	ID	Tipus	Descripció	Des de	Fins a	Import	Publicat
1	72	Tots	Animal de companyia	2012-01-01	2012-12-31	26.00	✓
2	48	Tipus 10	Despeses	2012-01-01	2012-12-31	3.00	✓
3	49	Tipus 10 + canal	Despeses	2012-01-01	2012-12-31	3.00	✓
4	50	Tipus 20	Despeses	2012-01-01	2012-12-31	3.00	✓
5	51	Tipus 20 + canal	Despeses	2012-01-01	2012-12-31	3.00	✓
6	52	Tipus 30	Despeses	2012-01-01	2012-12-31	4.00	✓
7	53	Tipus 30 + canal	Despeses	2012-01-01	2012-12-31	4.00	✓

Figura 5.4.6.3.1 Imatge de l'aplicatiu. Pantalla modificar preu.

5.5 Mòdul de Facturació

La part més important del projecte fa referència al mòdul de facturació. Cal recordar que el component "com-property" no disposa d'un mòdul de facturació.

5.5.1 Objectius

L'objectiu és crear un mòdul de facturació, integrat amb el component "com-property" per tal de generar factures en línia i que el client final pugui, des del frontend de l'aplicació consultar l'estat i el detall d'aquestes.

5.5.2 Model d'Entitat-Relació

Per tal de realitzar el mòdul de facturació s'ha interactuat amb les entitats del component "com-property" i amb les noves entitats de "preus" i de "clients" creades i explicades anteriorment.

- **Preus:** Taula jos_properties_price. És la taula on hi ha enregistrat els imports de cada propietat per tipus.
- **Productes:** Taula jos_properties_products. És la taula on s’hi emmagatzemen tots els productes (immobles de l’aplicació). Pertany al mòdul “com-property”.
- **Factura:** Taula jos_properties_bill. És la taula on es guarda la capçalera de la factura. Hi ha la informació bàsica per generar una factura.
- **Detall d’una factura:** Taula jos_properties_bill_detail. És la taula que emmagatzema els atributs que conformen el detall de cada factura. Una factura pot tenir més d’un detall. Per exemple, un mateix client pot fer una reserva de dos habitatges en períodes diferents.

5.5.2.1 Diagrama Entitat-Relació

El Diagrama Entitat-Relació entre les entitats “FACTURA”, “DETALL FACTURA”, “CLIENTS”, “PREUS” i “ESTAT FACTURA” és el següent:

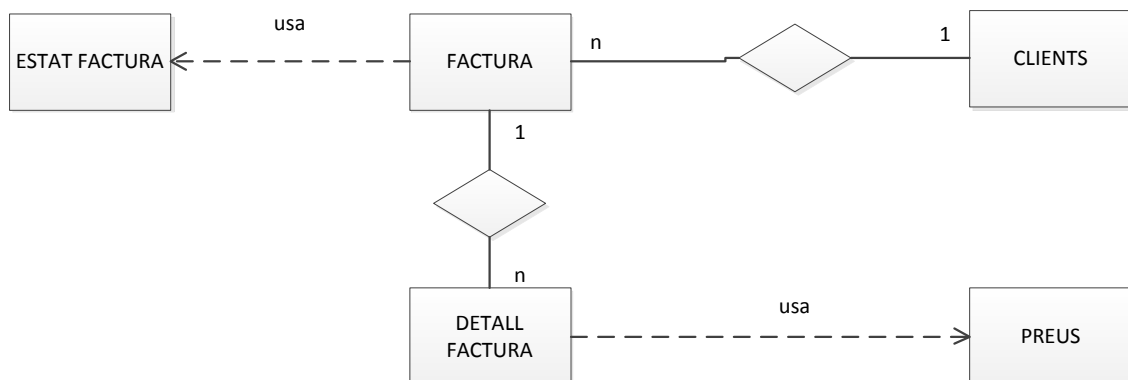


Figura 5.5.2.1.1 Diagrama Entitat – Relació del mòdul de facturació.

5.5.2.2 Model Relacional

La definició de les taules és la següent, tenint en compte que els valors remarcats amb negreta són **claus primàries**, els valors amb cursiva són *claus foranes* i els valors que estan entre parèntesi és la taula origen dels valors:

La taula *JOS_PROPERTIES_BILL* guarda la informació sobre les factures.

joseproperties_bill		
Nom	Descripció	Tipus
id	Identificació únic de factura	int(11)
date	Data de creació de la factura	date
<i>id_client</i>	Identificació de client. (JOS_PROPERTIES_CLIENTS.ID)	int(11)
<i>stat</i>	Estat de la factura. (JOS_PROPERTIES_BILL_STAT.ID)	tinyint(7)

La taula *JOS_PROPERTIES_BILL_DETAIL* guarda la informació sobre el detall d'una factura. Un detall de factura només pot tenir assignada una factura.

joseproperties_bill_detail		
Nom	Descripció	Tipus
id	Identificador únic de detall de factura	int(11)
<i>id_bill</i>	Identificador de factura. (JOS_PROPERTIES_BILL.ID)	int(11)
<i>id_product</i>	Identificador de product. (JOS_PROPERTIES_BILL.ID)	int(11)
import	Import	decimal(15,3)
from	Data inici	date
to	Data final	date
aux	Auxiliar	varchar(12)
<i>id_type</i>	Tipus d'habitatge. (JOS_PROPERTIES_TYPE.ID)	int(2)
ind_canal	Indicador de canal	int(1)
ind_sup_bed	Indicador de llit supletori	int(1)
ind_linen_bed	Indicador de lloguer de llençols	int(1)
ind_clean	Indicador de neteja final	int(1)
ind_animal	Indicador de animal	int(11)

La taula *JOS_PROPERTIES_BILL_STAT* guarda la informació sobre l'estat de les factures.

jos_properties_bill_stat		
Nom	Descripció	Tipus
id	Identificador de Estat de la factura	tinyint(7)
name	Descripció	varchar(20)

La taula *JOS_PROPERTIES_BILL_STAT* guarda la informació dels clients del sistema.

jos_properties_clients		
Nom	Descripció	Tipus
id	Codi d'identificació del client	int(11)
nif	Codi Identificatiu del client	varchar(50)
name	Nom del client	varchar(255)
alias	Sobrenom del client	varchar(50)
email	Correu electrònic del client	varchar(100)
surname_1	Primer Cognom client	varchar(255)
surname_2	Segon Cognom client	varchar(255)
address	Adreça postal client	varchar(255)
pcode	Codi postal de la població	varchar(10)
locality	Localitat o Població	varchar(255)
state	Estat	varchar(255)
country	País	varchar(255)
phone	Telèfon	varchar(30)
mobile	Telèfon Mòbil	varchar(255)
fax	Fax	varchar(30)
ccc	Nº de compte bancari	varchar(20)
bank	Nom del banc	varchar(255)
published	Indicador de publicat o no publicat	tinyint(1)
ordering	Ordre	int(3)
image	Nom de la imatge relacionada	varchar(70)
mid	Codi d'identificació de l'usuari	int(11)
<i>gid</i>	Indicador de grup. (JOS_GROUP.ID)	tinyint(3)

La taula JOS_PROPERTIES_PRICE guarda la informació dels preus d'un tipus entre dues dates.

jos_properties_price		
Nom	Descripció	Tipus
id	Identificador únic de preu	int(11)
description	Descripció del preu	text
import	Import	decimal(15,2)
validfrom	Data inici	date
validto	Data fi	date
rateperday	Preu per dia	decimal(15,3)
rateperweek	Preu per setmana	decimal(15,3)
published	Indicador de publicació	tinyint(1)
ordering	Indicador d'ordre	int(6)
cid	Tipus d'habitatge. (JOS_PROPERTIES_TYPE.ID)	int(2)

Representació gràfica de les taules, els seus camps i les seves relacions.

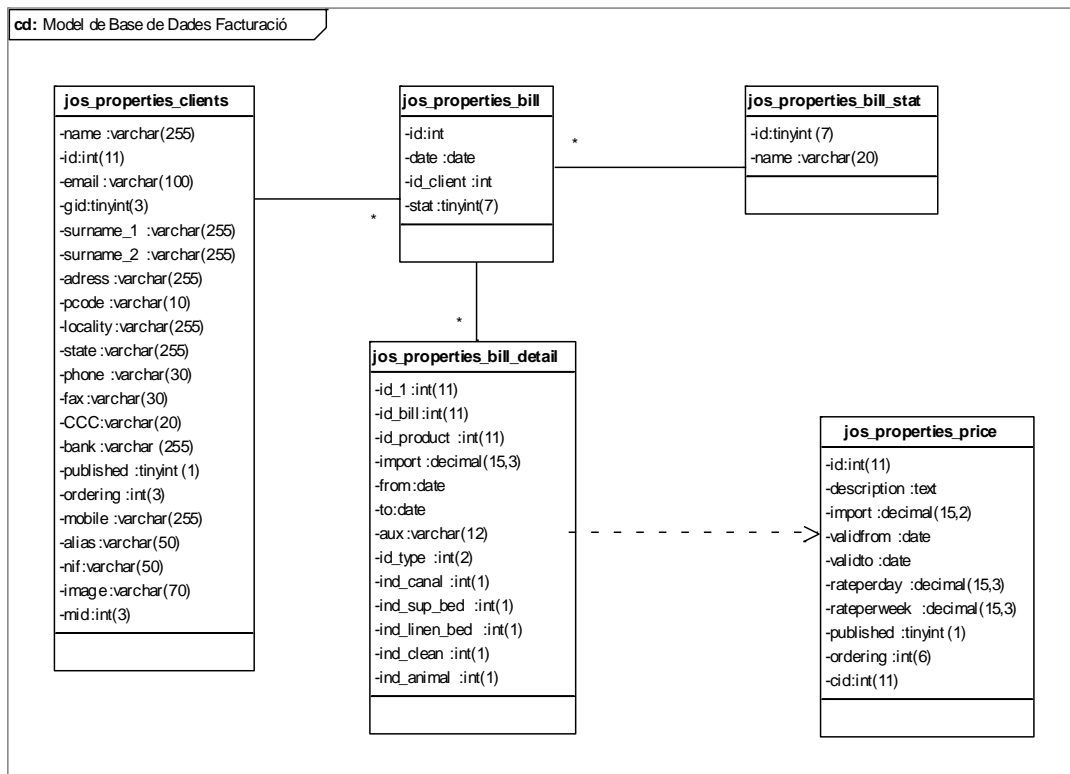


Figura 5.5.2.2.1 Model Relacional de base de dades – Relació del mòdul de facturació.

5.5.3 Descripció general

Aquest sistema permetrà:

- Afegir, modificar i eliminar factures.
- Afegir, modificar i eliminar els detalls de les factures.
- Calcular l'import total del detall d'una factura.
- Calcular l'import total de la factura.

L'aplicatiu té dos actors diferents. **L'actor gestor** (que s'encarrega del manteniment dels clients, de les factures i dels preus) i que accedeix a l'aplicació a través del backend, i **l'actor Usuari** (un client) que accedeix a través del frontend.

5.5.4 Diagrama de casos d'ús usuari Gestor

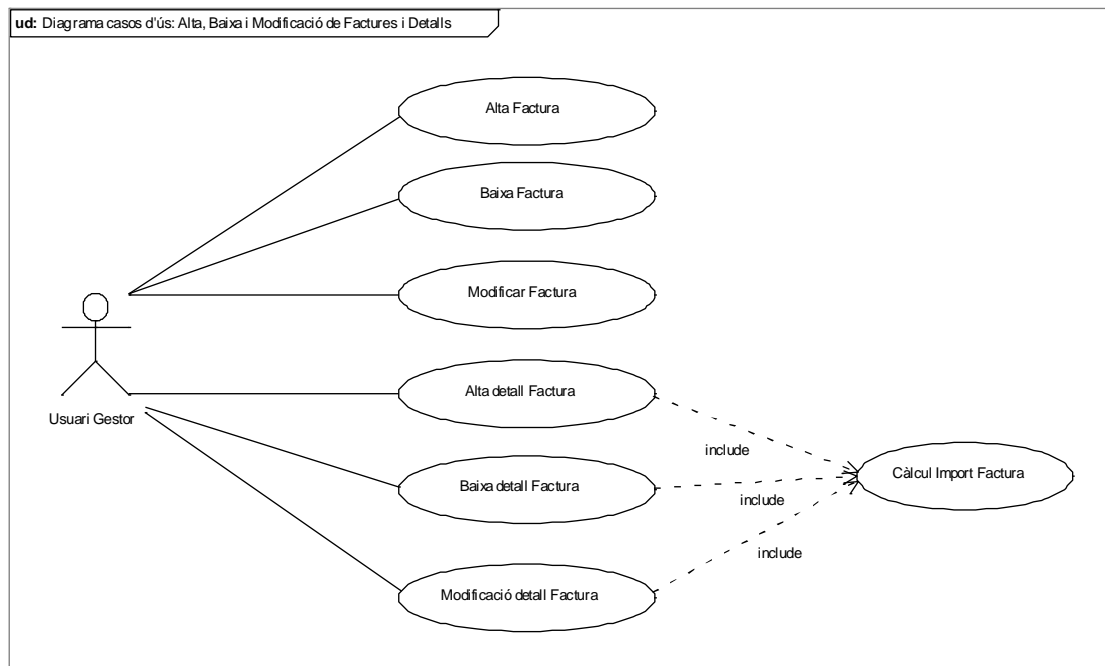


Figura 5.3.4.1 Diagrama casos d'ús – Alta, baixa i modificació de factures i detalls.

5.5.5 Casos d'ús

5.5.5.1 Alta Factura

	Cas d'ús: Alta Factura
Funcionalitat:	Crear una factura.
Descripció:	Modificació dels atributs de la factura.
Actors:	Usuari Gestor.
Casos d'ús relacionats:	Afegir un detall de la factura, modificar un detall de la factura i eliminar un detall de la factura.
Precondició:	L'usuari gestor s'ha identificat al sistema.
Flux:	<ol style="list-style-type: none">1. Entrar dades.2. Crear una factura.3. Assignar una factura a un client.4. Afegir una factura a la llista de factures.
Postcondició	Factura és enregistrada correctament al sistema. Si el camp data és nul, l'aplicatiu guarda per defecte la data actual.

5.5.5.2 Modificar Factura

	Cas d'ús: Modificar Factura
Funcionalitat:	Crear una factura.
Descripció:	S'introdueixen les dades (client, data facturació i estat de la factura).
Actors:	Usuari Gestor.
Casos d'ús relacionats:	
Precondició:	L'usuari gestor s'ha identificat al sistema.
Flux:	<ol style="list-style-type: none">1. Seleccionar la factura.2. Modificar les dades.3. Actualitzar la factura a la llista de factures.
Postcondició	La factura s'actualiza correctament al sistema. Si el camp data és nul, l'aplicatiu guarda per defecte la data actual.

5.5.5.3 *Baixa Factura*

	Cas d'ús: Baixa Factura
Funcionalitat:	Eliminar una factura.
Descripció:	Seleccionar la factura, i eliminar les factures i els detalls corresponents.
Actors:	Usuari Gestor.
Casos d'ús relacionats:	Eliminar el detall de la factura.
Precondició:	L'usuari gestor s'ha identificat al sistema.
Flux:	<ol style="list-style-type: none"> 1. Seleccionar la factura. 2. Si la factura té detall. <ol style="list-style-type: none"> 2.1 Eliminar el detall de la factura de la llista de detalls de factures. 3. Eliminar la factura de la llista de factures.
Postcondició	La factura i el seu detall han estat eliminats de la llista de factures i de la llista de detalls de factures.

5.5.5.4 *Alta Detall Factura*

	Cas d'ús: Alta detall Factura
Funcionalitat:	Crear un detall d'una factura.
Descripció:	Alta d'un detall d'una factura.
Actors:	Usuari Gestor.
Casos d'ús relacionats:	Afegir una factura, modificar una factura i eliminar una factura.
Precondició:	L'usuari gestor s'ha identificat al sistema. La factura existeix al sistema. La data inici i la data fi són vàlides.
Flux:	<ol style="list-style-type: none"> 1. Obtenir l'identificador de la factura. 2. Introduir les dades de la factura. 3. Calcular l'import. 4. Afegir un detall a la factura de la llista de detalls de factures.
Postcondició	El detall de la factura és enregistrat correctament al sistema.

5.5.5.5 *Modificar Detall Factura*

	Cas d'ús: Modificar Detall d'una Factura
Funcionalitat:	Modificació d'un detall d'una factura.
Descripció:	Modificació d'un detall d'una factura.
Actors:	Usuari Gestor.
Casos d'ús relacionats:	Afegir una factura, modificar una factura i eliminar una factura.
Precondició:	L'usuari gestor s'ha identificat al sistema. La factura existeix al sistema. La data inici i la data fi són vàlides.
Flux:	<ol style="list-style-type: none">1. Introduir noves dades al detall de la factura.2. Calcular l'import.3. Actualitzar el detall de la factura a la llista de detalls de factures.
Postcondició	El detall de la factura és enregistrat correctament al sistema.

5.5.5.6 *Baixa Detall Factura*

	Cas d'ús: Baixa detall d'una Factura
Funcionalitat:	Eliminar un detall d'una factura.
Descripció:	Seleccionar el detall d'una factura, i eliminar-lo.
Actors:	Usuari Gestor.
Casos d'ús relacionats:	Eliminar la factura.
Precondició:	L'usuari gestor s'ha identificat al sistema.
Flux:	<ol style="list-style-type: none">1. Seleccionar el detall d'una factura.2. Eliminar el detall d'una factura de la llista de detalls de factures.
Postcondició	El detall és eliminat de la llista de detalls de factures.

5.5.6 Diagrama de casos d'ús usuari Client

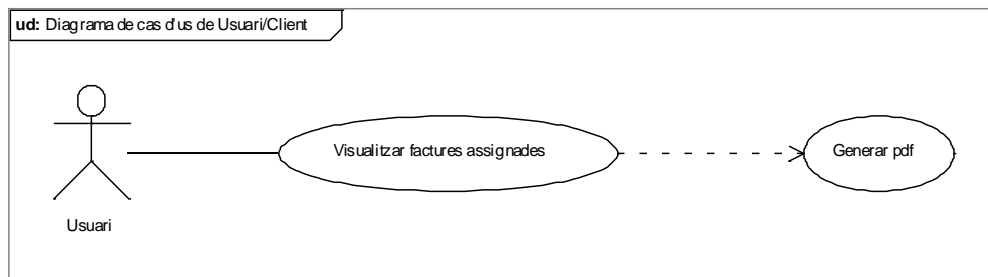


Figura 5.5.6.1 Diagrama casos d'ús – Visualitzar factures assignades.

5.5.6.1 Visualitzar Factures

	Cas d'ús: Visualitzar Factures
Funcionalitat:	Mostrar per pantalla les factures assignades a un determinat usuari.
Descripció:	L'usuari client, una vegada ha accedit a l'aplicatiu, disposa i pot descarregar-se les factures assignades a ell.
Actors:	Usuari client.
Casos d'ús relacionats:	
Precondició:	L'usuari client s'ha identificat al sistema.
Flux:	<ol style="list-style-type: none"> 1. Obtenir l'identificador del usuari. 2. Mostrar les factures assignades al usuari. 3. Si l'usuari client vol descarregar-se la factura. <ol style="list-style-type: none"> 3.1 Generar PDF.
Postcondició	Factura mostrada.

5.5.7 Interfícies d'usuari

5.5.7.1 Alta Factura

Una vegada s'ha accedit a l'aplicació "com-property", cal entrar a la pestanya "Facturació". A continuació, cal clicar sobre "Nou".

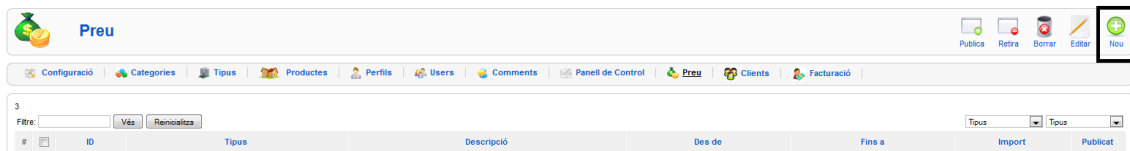


Figura 5.5.7.1.2 Imatge de l'aplicatiu. Nova factura.

Seleccionen un client, entrem la data i la desem.

Facturació

ID: 0

Data: []

Client: David

Estat: Pendent

Import: 0

Afegir / Modificar / Eliminar Detall Factura

Figura 5.5.7.1.2 Imatge de l'aplicatiu. Pantalla entrar nova factura.

5.5.7.2 Baixa Factura

Una vegada s'ha accedit "com-property", cal entrar a la pestanya "Facturació". Seleccionem la factura a eliminar, i cliquem sobre "Borrar".

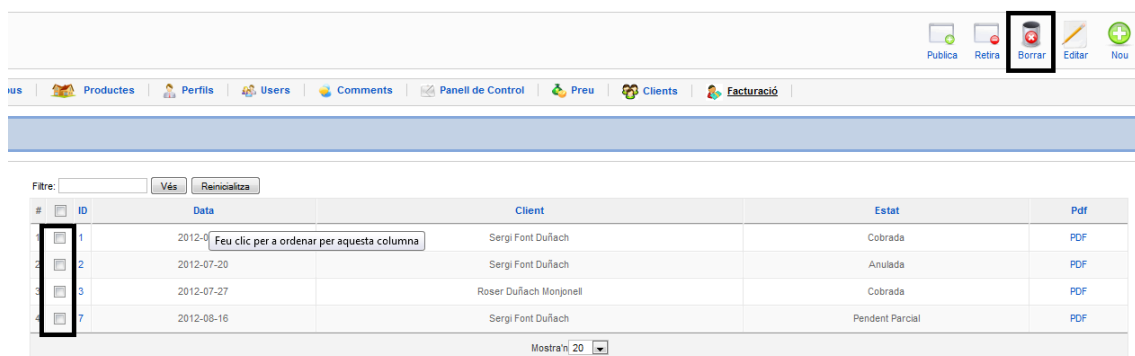


Figura 5.5.7.2.1 Imatge de l'aplicatiu. Pantalla esborrar factura.

5.5.7.3 Modificació Factura

Una vegada s'ha accedit a "com-property", cal entrar a la pestanya "Facturació". Després, s'ha de seleccionar una factura de la llista de factures. La modificació es pot realitzar de dues formes diferents:

- Clicant directament sobre la descripció de la factura per accedir al seu detall i fer la modificació corresponent.
- Marcant el registre amb el "check" i clicar sobre "Editar".

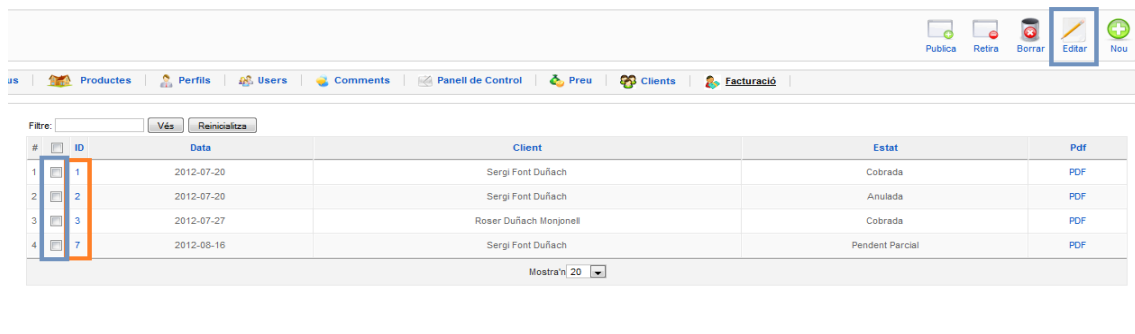


Figura 5.5.7.3.2 Imatge de l'aplicatiu. Pantalla modificació factura. Llistat.

Facturació

ID:	1
Data:	2012-07-20
Client:	Sergi Font Duñach
Estat:	Cobrada
Import:	14152.000

Figura 5.5.7.3.2 Imatge de l'aplicatiu. Pantalla modificació factura. Llistat

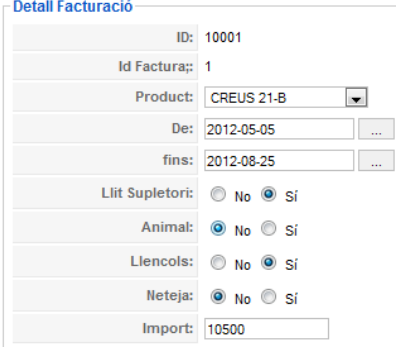
5.5.7.4 Alta, Baixa i Modificació Detall Factura

Un cop s'ha accedit a l'aplicació "com-property", cal entrar a la pestanya "Facturació". A continuació, s'ha de seleccionar una factura i entrar el detall corresponent.



The screenshot shows a form titled "Facturació" with the following fields: ID: 0, Data: (empty), Client: David (dropdown), Estat: Pendent (dropdown), and Import: 0. To the right of the form, the text "Afegir / Modificar / Eliminar Detall Factura" is displayed in blue.

Figura 5.5.7.4.1 Imatge de l'aplicatiu. Alta, baixa i modificació detall factura.

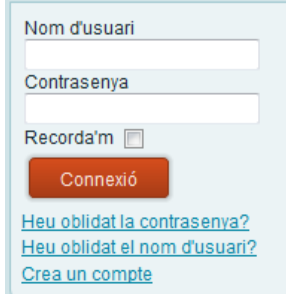


The screenshot shows a form titled "Detall Facturació" with the following fields: ID: 10001, Id Factura: 1, Product: CREUS 21-B (dropdown), De: 2012-05-05 (calendar icon), fins: 2012-08-25 (calendar icon), Llit Supleteri: No (radio), Sí (radio), Animal: No (radio), Sí (radio), Llencols: No (radio), Sí (radio), Neteja: No (radio), Sí (radio), and Import: 10500.

Figura 5.5.7.4.2. Imatge de l'aplicatiu. Modificació detall factura.

5.5.7.5 Visualitzar Factures

Per tal de visualitzar les factures, des d'un usuari client, s'ha d'entrar a la web www.portlasardana.com i registrar-se com a usuari, el qual necessita una contrasenya.



The screenshot shows a login form with the following elements: "Nom d'usuari" (text input), "Contrasenya" (password input), "Recordar'm" (checkbox), a "Connexió" button, and three links: "Heu oblidat la contrasenya?", "Heu oblidat el nom d'usuari?", and "Crea un compte".

Figura 5.5.7.5.1 Entrada a l'aplicació des del frontend.

Menú principal

- Inici
- Contacte

Menú de usuario

- Tu Perfil
- Enviar article
- Enviar enllaç
- Tancar Sessió

Factura

- Facturació

🇪🇸 🇫🇷 🇩🇪 🇮🇹

Llista de Factures





#		Id	Date	Estat	Import	Pdf
1		1	2012-07-20	Cobrada	14632.000	PDF
2		2	2012-07-20	Anulada	400.000	PDF
1		7	2012-08-16	Pendent Parcial	0	PDF

Figura 5.5.7.5.2 Visualització de factures des del frontend.

6 IMPLEMENTACIÓ I PROVES

6.1 Implementació. Càlcul de l'Import

Per tal de realitzar el càlcul de l'import de la factura cal tenir en compte diferents variables que s'expliquen a continuació.

- **Import del lloguer:** Fa referència a l'import pel lloguer d'un tipus d'habitatge durant una setmana.
- **Import de les despeses:** Fa referència a les despeses de gas i electricitat de l'habitatge. Segons el tipus d'habitatge s'aplica un import o un altre. El càlcul és diari.
- **Import neteja:** Fa referència a l'import de la neteja final de l'habitatge. És obligatori i només s'aplica una sola vegada per estada. L'import de neteja varia en funció del tipus de l'habitatge.
- **Import animal de companyia:** Fa referència a l'import per l'estada d'un animal de companyia en un habitatge. S'aplica una sola vegada per estada.
- **Import lloguer llençols:** Fa referència al lloguer dels llençols. S'aplica una sola vegada per estada.

A part d'aquestes variables, cal tenir present que l'estada pot afectar a dos períodes amb diferents preus setmanals.

En el següent quadre, proporcionat pel client, observar les diferents variables descrites anteriorment.

Llista de Preus:

- **TIPUS 10:** Estudi amb menjador-estar i sofà llit, cuina i bany o dutxa (2 persones)
- **TIPUS 20:** Apartament amb un dormitori de dos llits, menjador-estar, sofà-llit, cuina i bany o dutxa (3 persones).
- **TIPUS 30:** Apartament amb dos dormitoris, menjador-estar, sofà-llit, cuina i bany amb dutxa o banyera (4-5 persones).
- **TIPUS 40:** Xalet amb dos dormitoris, menjador-estar, sofà-llit, cuina i bany amb dutxa o banyera (4-5 persones).
- **TIPUS 45:** Casa adossada amb tres dormitoris, menjador-estar, sofà-llit, cuina, bany (6-7 persones).
- **TIPUS 50:** Xalet amb tres dormitoris, menjador-estar, sofà-llit, cuina, bany (6-7 persones).
- **TIPUS 60:** Xalets i apartaments amb piscina, consultar preus.

PREUS PER SETMANA			
TEMPORADA	30/07 – 25/08	31/03 – 30/07 i 25/08 – 29/09	29/09 – 31/03
TIPUS 10	225 €	165 €	115 €
TIPUS 20	335 €	245 €	165 €
TIPUS 30	405 €	295 €	230 €
TIPUS 40	525 €	395 €	290 €
TIPUS 45	560 €	460 €	320 €
TIPUS 50	665 €	495 €	355 €

Els xalets i apartaments situats dins d'una parcel·la amb canal tindran un augment del 10%.

DESPESES i GAS		NETEJA FINAL	
TIPUS 10 i 20	3 € / dia	TIPUS 10 i 20	40 €
TIPUS 30 i 40	4 € / dia	TIPUS 30 i 40	50 €
TIPUS 45 i 50	5 € / dia	TIPUS 45 i 50	55 €
		GOS	26 €

Figura 6.1.1.1 Imatge de preus del client.

6.2 Càlcul

Per realitzar correctament el càlcul de l'import de l'estada, cal dividir el càlcul en setmanes que hi ha entre la data d'entrada i la data de sortida de l'estada i fer el càlcul de totes les variables setmanalment.

L'algorisme amb pseudocodi queda de la següent forma:

```
Calcular n° setmanes entre data entrada i data sortida
```

```
Mentre quedin setmanes per calcular fer
```

```
    Calcular import setmana segons tipus
```

```
    Sumar import al total
```

```
    Següent setmana
```

```
Fi mentre
```

6.2.1 Exemple de Càlcul

Per entendre millor el funcionament de la funció que calcula l'import total d'una factura uses presenta un exemple.

Suposem que ens trobem en aquest cas:

Detall Facturació	
ID:	1
Id Factura:	1
Product:	Alberes 61 - A
De:	2012-07-21
fins:	2012-09-01
Llit Supletori:	<input type="radio"/> No <input checked="" type="radio"/> Sí
Animal:	<input type="radio"/> No <input checked="" type="radio"/> Sí
Llencols:	<input type="radio"/> No <input checked="" type="radio"/> Sí
Neteja:	<input type="radio"/> No <input checked="" type="radio"/> Sí
Import:	4132

Figura 6.1.3.1 Imatge de l'aplicatiu. Detall d'una factura.

La factura nº1 (id_bill) té un detall (ID) corresponent al Producte “Alberes 61 A” entre les dates 21-07-2012 i 01-09-2012 i requereix un llit supletori i llençols. També, té un animal de companyia i se l’hi haurà de cobrar la neteja final.

Amb totes aquestes dades, podem saber que el Producte “Alberes 61 A”, pertany al tipus “Tipus 50”. Per tant, per calcular l’import total, només s’ha de tenir en compte el “Tipus 50” de la taula de preus.

Un cop s’ha trobat el seu tipus, cal recordar que cada producte té un tipus assignat, el que es fa dividir en setmanes les dades entrades pel gestor. S’avalua i es calcula setmana per setmana.

Com s’ha detallat anteriorment, hi ha conceptes que s’apliquen setmanalment i d’altres diàriament.

El quadre següent és la traça del que la funció de calcular l’import de l’estada:

Data inici	Data Fi	Import Lloguer	Import Despeses	Import Llit Supletori	Animal	Llençols	Neteja
21-07-2012	28-07-2012	495€	35€	30€	26€	11€	55€
28-07-2012	04-08-2012	665€	35€	30€			
04-08-2012	11-08-2012	665€	35€	30€			
11-08-2012	18-08-2012	665€	35€	30€			
18-08-2012	25-08-2012	665€	35€	30€			
25-08-2012	01-09-2012	495€	35€	30€			
Total		3.650 €	210 €	180 €	26 €	11 €	55 €

TOTAL 4.132€

6.3 Generació de Factures amb PDF

Una vegada s’ha calculat l’import de la factura, cal generar-la per a poder ser impresa i visualitzada pel client. Per poder generar la factura amb PDF vaig buscar una llibreria en PHP que em pogués permetre passar de HTML a PDF. La llibreria en qüestió és DOMPDF, la qual per poder utilitzar-la, cal descarregar-se les llibreries i instal·lar-les al servidor.

El seu funcionament és molt fàcil i entenedor. Cal inserir i concatenar tot el codi HTML a una variable, i utilitzar el DOMPDF perquè transformi el codi HTML que hi ha inserit a la variable en PDF.

Però, per problemes d'integració no s'ha pogut inserir el fitxer PHP a dins el Joomla!. Després de varies proves, quan s'encapsula dintre el Joomla! no es genera correctament el PDF. La solució que es va adoptar és no implementar el fitxer PHP dins del Joomla! i generar la factura en PDF de forma externa al Joomla!.



Empuriarent
info@empuriarent.com

Factura # 2
Creatada el 2012-07-20

Detalls del Client
Nif: 404444912
Nom: Sergi Font Duñach
Adreça: del Pont 44
Email: u1055829@correu.udg.edu

Estat Factura: Anulada

Descripció Producte	Setmana Inici	Setmana Fi	Import
Port La Sardana 12	2012-08-09	2012-08-24	400.000€
		Total	400€

Aquesta comunicació és per a l'ús exclusiu del destinatari i poden contenir informació privilegiada, confidencial o privilegiada. Si no és el destinatari indicat, qualsevol ús, còpia, divulgació, difusió o distribució està estrictament prohibida.

© Empuriarent All Rights Reserved

Figura 6.1.4.1 Model de factura generada amb PDF.

6.4 Implementació. Directoris

Un altre punt a destacar és tenir correctament referenciat els directoris que han sofert la inserció o la modificació d'alguns dels seus arxius per tal de facilitar la tasca a la gent que continuï desenvolupant el component "com-property".

Cal diferenciar els directoris i fitxers creats al backend (panell d'administració) o el frontend (part visible per l'usuari registrat).

Backend:

- administrator/components/com_properties/
 - controllers/
 - bill.php
 - bill_detail.php
 - clients.php
 - price.php
 - elements/
 - helpers/
 - select.php
 - includes
 - img/
 - bill.png
 - clients.png
 - price.png
 - t_bill.png
 - t_clients.png
 - t_price.png
 - models/
 - bill.php
 - bill_detail.php
 - clients.php
 - price.php
 - tables/
 - bill.php
 - bill_detail.php
 - clients.php
 - price.php

- updates
- views/
 - bill/
 - tpl/
 - default.php
 - form.php
 - formpdf.php
 - index.html
 - index.html
 - view.html.php
 - billdetail/
 - tpl/
 - default.php
 - form.php
 - index.html
 - index.html
 - view.html.php
 -
 - clients/
 - tpl/
 - default.php
 - form.php
 - index.html
 - index.html
 - view.html.php
 -
 - price/
 - tpl/
 - default.php
 - form.php
 - index.html
 - index.html
 - view.html.php

Frontend:

- /components/com_properties/
 - controllers/
 - bill.php
 - elements/
 - helpers/
 - select.php
 - includes
 - models/
 - bill.php
 - tables/
 - updates
 - views/
 - bill/
 - tmp/
 - default.php
 - form.php
 - formpdf.php
 - index.html
 - index.html
 - view.html.php

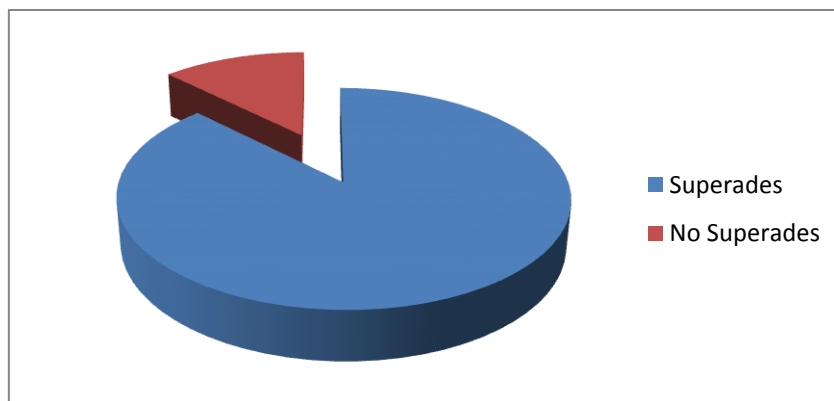
6.5 Proves

A continuació es pot veure el resultat de les proves unitàries realitzades sobre l'aplicació:

Codi	Verificació	Resultat Esperat	Test
1	Afegir, modificar i eliminar clients	L'aplicació ha de afegir, modificar i eliminar clients correctament. Tots els atributs es guarden correctament.	OK
2	Afegir, modificar i eliminar preus	L'aplicació ha de afegir, modificar i eliminar preus correctament. Tots els atributs es guarden correctament.	OK
3	Afegir, modificar i eliminar factures	L'aplicació ha de afegir, modificar i eliminar factures correctament. Tots els atributs es guarden correctament.	OK
4	Afegir, modificar i eliminar detalls de factura	L'aplicació ha de afegir, modificar i eliminar detalls de factura correctament. Tots els atributs es guarden correctament.	OK
5	Mostrar les factures assignades a un usuari al frontend de l'aplicació.	L'aplicació ha de mostrar per pantalla les factures assignades a un usuari.	OK
6	Generar factures amb PDF "online".	L'aplicació ha de generar factures, i el seu detall amb format PDF.	KO

7	Eliminar una factura i el seu detall	Un cop eliminat una factura, cal eliminar el detall de la factura corresponent.	OK
8	Càlcul del detall d'una factura	L'aplicació ha de calcular internament l'import de la factura.	OK

Proves Superades	Proves NO Superades	Percentatge
8	1	87,50%



6.5.1.1 Conclusió de les proves realitzades

Les proves d'integritat documentades seguint l'estàndard IEE 829-1983 ha donat un resultat positiu. L'aplicació és 100% funcional, tot i que en una de les proves s'ha detectat un problema.

El problema és degut a problemes d'integració entre Joomla! i DOMPDPF. DOMPDPF és una llibreria de PHP que s'utilitza per generar PDF's. Es varen realitzar proves i l'aplicació no funcionava quan estava integrat a Joomla!, però funcionava correctament quan estava separada i es feia independent del Joomla!.

6.6 Accés a l'aplicació

Per accedir a l'aplicació cal entrar al següent enllaç:

www.portalasardana.com/administrator/



Figura 6.1.7.1 Imatge de l'entrada a l'administració a Joomla!.

S'ha creat un usuari de proves per poder visualitzar l'aplicatiu.

- **Nom d'usuari:** joan
- **Contrasenya:** joan01 (amb la interfície configurada en català)

A continuació, cal prémer "Connexió".

S'accedeix a la pantalla principal de Joomla!. Una vegada s'hi ha accedit, s'ha d'anar al menú horitzontal "Components", i clicar sobre el menú "Properties".



Figura 6.1.7.2 Imatge dels Menús interiors del Joomla!.

En aquest moment s'entra al component "com-property".

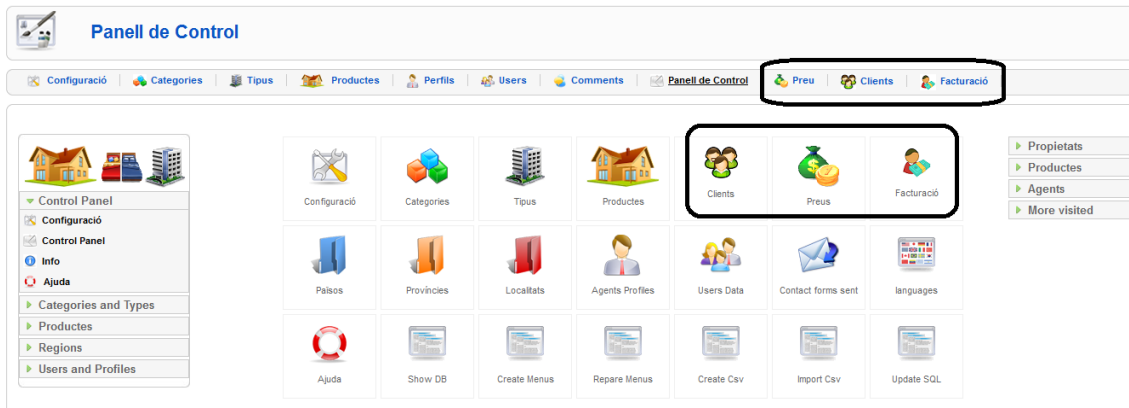


Figura 6.1.7.3 Imatge dels menús interiors del Joomla!. Preus, Clients i Facturació.

I a partir d'aquest moment es pot començar a navegar pels mòduls creats que incorpora l'aplicació.

7 CONCLUSIONS

Personalment valoro molt positivament l'experiència i considero que he complert amb els objectius proposats a l'inici del Projecte Final de Carrera.

Del treball realitzat puc extreure els següents aprenentatges:

- La formalització dels requeriments a partir de l'anàlisi i una bona definició de l'abast del projecte són aspectes vitals per a poder planificar correctament els projectes.
- El disseny és essencial per estructurar el projecte, concebre els subsistemes i donar garanties al procés de desenvolupament.

Pel que fa a la implementació:

- Treballar en projectes de programari lliure permet veure exemples fets per altres programadors, la qual cosa permet comparar i se'ns dubte això, ajuda a millorar la qualitat del codi.
- Fer servir un framework de desenvolupament estalvia considerablement línies de codi, la qual cosa implica alhora menys errors.
- El patró de programació Model Vista Controlador millora la llegibilitat del codi enormement i fa que altres programadors puguin entendre el programa ràpidament.

També, m'agradaria comentar que durant el procés de definició dels objectius i dels requeriments del sistema te n'adones que amb una possible millora a l'aplicació, pots aconseguir una millora substancialment notable en el seu treball diari i un increment en la seva productivitat. Parlant amb el client, vaig fer la proposta de que una vegada la factura estava generada, que l'aplicació mateixa generés un document (en format PDF). En el meu cas, havia estat fent molta feina, però el client no veia els resultats, per tant la creació de la factura en PDF ha estat de gran utilitat per visualitzar la feina feta.

En aquest projecte s'ha posat en pràctica els coneixements adquirits en les assignatures com ara: Enginyeria del Software, Metodologia i Tecnologia de la Programació, Base de Dades I, Base de Dades II, Sistemes Operatius i Gestió de Sistemes de la Informació.

8 TREBALL FUTUR

Arribat a aquest punt s'ha de tenir en compte què és el que es pot modificar de la implementació proposada.

Propostes de millores:

- En el sistema actual, una vegada l'usuari té assignat un client, i aquest disposa de factures les pots visualitzar a través del frontend de l'aplicatiu. Una possible millora seria que quan canviés d'estat la factura (cobrada, pendent,...) s'enviés un e-mail a l'usuari per tal de informar-lo que l'estat de la factura s'ha modificat.
- Com s'ha explicat anteriorment el component "com-property" feia el càlcul dels preus en funció del producte en qüestió i no del tipus de producte. Amb la nova ampliació del component, els càlculs ja es fan a partir del tipus de producte. Però, hi ha un apartat de "reserves" a la web que encara fa el càlcul segons el producte i no per tipus. Aquest punt no es va poder implementar, perquè en el moment de implementar-lo coincidia amb plena temporada turística i es deixa per més endavant.

9 INSTAL·LACIÓ

Per instal·lar tot el component, cal anar al menú “Extensions” del Joomla! i entrar a l’opció “Instal·la/Desinstal·la”.



Figura 9.1 Imatge dels Menús interiors del Joomla!. Instal·lació.

Una vegada s’ha accedit, cal clicar sobre “examinar”... i seleccionar l’arxiu “com_property.zip”. A continuació, s’ha de clicar “penjar el fitxer & Instal·la”.

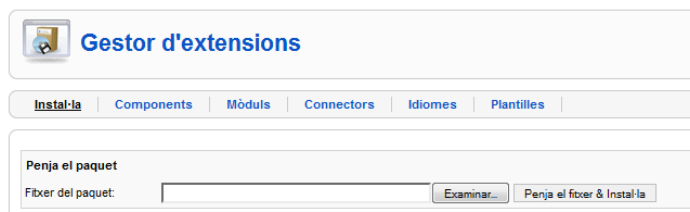


Figura 9.2 Imatge dels Menús interiors del Joomla!. Penjar el fitxer.

10 BIBLIOGRAFIA / WEBGRAFIA

- **Serrats, Carles** (2008). Programació de components MVC per Joomla! 1.5.x [document en línia]. [Data de consulta: 19 de Maig de 2012].
<http://downloads.joomlancode.org/frsrelease/3/7/6/37642/JoomlaSpanish.pdf>
- **Trejejo Alonso, Juan Antonio** (2007). Joomla! Para Principiantes (258 pàgines).Sense editorial. ISBN 84-611-3754-X.
- **PHP Hipertext Preprocessor, manual:**
www.php.net
- **Dompdf, manual i llibreries:**
<http://code.google.com/p/dompdf/>
- **Enciclopèdia lliure:**
<http://www.wikipedia.org>
- **Joomla!**
<http://www.joomlaspanish.org/>
<http://www.joomla.org/>
- **Component Com-property**
<http://www.com-property.com/>

11 ÍNDEX DE FIGURES

Figura 2.3.1.1. Icona de PHP.	10
Figura 2.3.1.2. Icona de MySQL.....	10
Figura 2.3.1.3. Icona de HTML.....	11
Figura 2.3.2.1. Icona de Notepad.	12
Figura 2.3.3.1. Icona de Poseidon for Uml 8.0.	12
Figura 2.3.3.2. Icona de Microsoft Word.	12
Figura 2.3.6.1 Model Vista Controlador de Joomla!.	16
Figura 5.1.2.1 Diagrama seqüència Model Vista Controlador de Joomla!.....	22
Figura 5.2.2.1 Model Relacional de base de dades del component “com-property”.....	23
Figura 5.2.3.1 Estructura de Directoris i Fitxers frontend.	24
Figura 5.2.3.2 Estructura de Directoris i Fitxers backend.	24
Figura 5.3.1.1 Pantalla d’administració de clients en format Microsoft Access.	26
Figura 5.3.2.1 Diagrama Entitat – Relació del mòdul de clients.	27
Figura 5.3.2.2 Model Relacional de base de dades – Relació del mòdul de clients.....	30
Figura 5.3.4.1 Diagrama casos d’ús – Alta, baixa i modificació de clients i assignar usuari a client.....	31
Figura 5.3.6.1.1 Imatge de l’aplicatiu. Nou client.	34
Figura 5.3.6.2.2 Imatge de l’aplicatiu. Entrar dades nou client.	34
Figura 5.3.6.2.1 Imatge de l’aplicatiu. Esborrar client.	35
Figura 5.3.6.3.1 Imatge de l’aplicatiu. Seleccionar client a editar.	35
Figura 5.3.6.3.2 Imatge de l’aplicatiu. Client a editar.	36
Figura 5.4.2.1.1 Diagrama Entitat-Relació del Mòdul de Clients.	37
Figura 5.3.2.2 Model Relacional de base de dades – Relació del mòdul de preus.	40

Figura 5.3.4.1 Diagrama de casos d'ús – Alta, baixa i modificació de preus.....	41
Figura 5.4.6.1.2 Imatge de l'aplicatiu. Pantalla entrar nou preu.	43
Figura 5.5.2.1.1 Diagrama Entitat – Relació del mòdul de facturació.....	45
Figura 5.3.4.1 Diagrama casos d'ús – Alta, baixa i modificació de factures i detalls.	49
Figura 5.5.6.1 Diagrama casos d'ús – Visualitzar factures assignades.....	53
Figura 5.5.7.1.2 Imatge de l'aplicatiu. Pantalla entrar nova factura.....	54
Figura 5.5.7.3.2 Imatge de l'aplicatiu. Pantalla modificació factura. Llistat.	55
Figura 6.1.1.1 Imatge de preus del client.....	59
Figura 6.1.3.1 Imatge de l'aplicatiu. Detall d'una factura.	60
Figura 6.1.4.1 Model de factura generada amb PDF.	62
Figura 6.1.7.1 Imatge de l'entrada a l'administració a Joomla!.....	68
Figura 6.1.7.2 Imatge dels Menús interiors del Joomla!.....	68
Figura 6.1.7.3 Imatge dels menús interiors del Joomla!. Preus, Clients i Facturació.....	69
Figura 9.1 Imatge dels Menús interiors del Joomla!. Instal·lació.....	72
Figura 9.2 Imatge dels Menús interiors del Joomla!. Penjar el fitxer.	72