



UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA



UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA

CAMPUS D'ALCOI

UNIVERSITAT POLITÈCNICA DE VALÈNCIA

Escola Politècnica Superior d'Alcoi

Anàlisi i desenvolupament d'un mòdul de gestió de
biblioteques per a l'ERP Odoo

Treball Fi de Grau

Grau en Enginyeria Informàtica

AUTOR/A: Toledo Coloma, Jordi

Tutor/a: Tomas Miquel, Jose Vicente

CURS ACADÈMIC: 2021/2022

Resum:

Els bibliotecaris ja no utilitzen el sistema tradicional de paper i bolígraf, sinó un software a partir del qual fan totes les gestions. Aquests programaris tenen uns desavantatges: segueixen una lògica complexa d'entendre per als novells, poc òptims, i que tenen dos grans problemàtiques:

- Actuen de forma aïllada amb els altres processos de la biblioteca. Açò potser una problemàtica de cara a resoldre incidències, ja que no hi ha administrador del sistema.
- No són personalitzables.

El canvi en el model que segueixen les biblioteques, per tant, no és una mala idea. El problema principal és que la implementació d'un ERP a les biblioteques no és factible per el tipus de negoci, ja que no produeixen ni facturen. No obstant, sí que seria factible amb la personalització d'un ERP.

Existeix un ERP, anomenat Odoo, que permet exactament això, i així amb la creació d'un mòdul que implemente la funcionalitat de la gestió bibliotecària a l'ERP, aquest podria ser utilitzat per les biblioteques com a alternativa al software privat.

Aquest treball de fi de grau consisteix en desenvolupar software que permeti dur a terme el procés de gestió bibliotecària dins d'un ERP, i que així pugui ser integrat en una biblioteca.

A més, el desenvolupament d'aquest programari permet oferir la major automatització possible del treball de gestió bibliotecària, i permet oferir un servei de manteniment per tal de millorar constantment el producte.

Paraules clau: Biblioteca, Gestió de biblioteques, ERP, Odoo.

Summary:

Librarians no longer use the traditional system of paper and pen, but a software from which they do all the work. These softwares have some disadvantages: they follow a logic that is difficult to understand for novices, not very optimal, and they have two big problems:

- They act in an isolated way with the other processes of the library. This may be a problem in terms of resolving incidents, since there is no system administrator.
- They are not customizable.

The change in the model that libraries follow, therefore, is not a bad idea. The main problem is that the implementation of an ERP in libraries is not feasible due to the type of business, since they do not produce or invoice. However, it could be more feasible with the customization of an ERP.

There is an ERP, called Odoo, that allows exactly this, and thus with the creation of a module that implements the library management functionality into the ERP, it could be used by libraries as an alternative to private software.

This final thesis consists of developing software that allows to carry out the process of library management within an ERP, so that it can be integrated into a library, but does not cover the implementation of the ERP as such in the organization.

Moreover, the development of this program allows to offer the greatest possible automation of the library management work and allows to offer a maintenance service to constantly improve the product.

Key words: Library, Library management, ERP, Odoo.

Contingut

1. Introducció al treball.	5
1.1 Definició d'un ERP	5
1.2 Definició d'Odoo	6
1.3 Conceptes d'Odoo	7
1.4 Motivació	10
1.5 Objectius	11
1.6 Estructura de la memòria	11
1.7 Relació amb assignatures	12
2. Introducció a la problemàtica	13
2.1 Model de negoci d'una biblioteca	13
2.2 Desenvolupament del mòdul en Odoo	17
2.3 Procés de gestió bibliotecària	19
3. Anàlisi de requeriments	22
3.1 Especificació de requeriments	22
3.2 Especificació de casos d'ús i permisos	24
3.3 Anàlisi de sol·lucions	28
3.4 Elecció d'una sol·lució	32
4. Disseny de la sol·lució	33
4.1 Capa de presentació	33
4.2 Capa de negoci	39
4.3 Capa de dades	41
5. Desenvolupament de la sol·lució	43
5.1 Ferramentes utilitzades	43
5.1.1 Python, PostgreSQL, Docker y Odoo11	43
5.1.2 VSCode	45
5.1.3 Github	45
5.2 Estructura dels fitxers programari	46
5.2.1 Mòdul de gestió bàsica	47
5.2.2 Mòdul per a préstecs	51
5.3 Contingut i funcionalitat	54
5.3.1 Llibres	56
5.3.2 Autors	61
5.3.3 Editorials	64
5.3.4 Categories	67
5.3.5 Ordinadors	69

5.3.6	Sucursals	74
5.3.7	Membres	77
5.3.8	Préstecs	83
5.3.9	Reserves	89
6.	Implementació de la sol·lució	94
6.1	Implantació de l'ERP en la biblioteca.....	94
6.2	Manteniment.....	94
6.3	Temps de vida del producte.....	97
7.	Conclusions i línies futures.....	99
8.	Bibliografia.....	100

1. Introducció al treball.

Aquest TFG es tracta d'un projecte de desenvolupament software, que busca solventar una problemàtica real com la integració de la gestió bibliotecària en un ERP, concretament en Odoo, per tal que aquest pugui ser utilitzat a una biblioteca municipal.

Abans de parlar del projecte, és necessari comprendre uns conceptes bàsics.

1.1 Definició d'un ERP

Els ERPs (Enterprise Resource Planning) són programaris que permeten dur a terme una gestió global de la empresa, fent-se càrrec de moltes operacions i activitats internes. Integra tots els processos de la mateixa en un únic programari i una única base de dades. Es pot dir que és un programari que coordina tots els sistemes d'una empresa.

Una definició més formal, segons Viquipèdia (2022), seria:

*“Un sistema de **planificació de recursos empresarials** o **ERP** (de l'anglès Enterprise Resource Planning) integra (o pretén integrar) totes les dades i els processos d'una organització en un sistema unificat. Un sistema ERP típic utilitzarà múltiples components de programari i maquinari per aconseguir la integració. Un component clau de la major part d'ERP és l'ús d'una base de dades única per emmagatzemar la informació dels diferents mòduls del sistema.”*

Aquest concepte ha començat a posar-se en pràctica a partir de la dècada del 2000, i a empreses grans solventa una immensa quantitat de problemes de gestió i optimització. Entre alguns dels avantatges que ofereixen els ERPs, segons Tomás-Miquel et al. (2008), trobem:

- Millora en l'optimització de processos, ja que tots els usuaris del sistema comparteixen la mateixa base de dades.
- Visió global de la empresa per part dels treballadors. Sense l'ERP, els departaments es troben separats i no estan comunicats. A més, les sucursals d'una empresa funcionen com a una empresa global, en lloc de funcionar com si es tractara d'empreses diferents.
- Reducció de costos e increment de productivitat.
- Aporta avantatges competitiu.

A la empresa, cada treballador té el seu usuari en l'ERP amb els seus respectius permisos, i treballa actualitzant la base de dades de forma concurrent, a la vegada que els altres treballadors de cada respectiu departament.

1.2 Definició d'Odoo

Odoo, anteriorment conegut com a OpenERP, és un ERP de codi obert programat amb Python (encara que una part de la web utilitza Javascript). Es tracta d'un ERP modular, on cadascun dels mòduls s'encarrega de dur a terme una funció empresarial. Per exemple:

- Mòdul de vendes: Abarca la funcionalitat de les vendes de productes, usada pels encarregats de la tenda o web.
- Mòdul de comptabilitat: Utilitzat pels comptables, i conté tota la informació necessària per a fer anàlisis i informes.
- Mòdul de compra: Permet dur a terme les compres als proveïdors.
- Mòdul d'inventari: Queden registrats els productes i les quantitats de cadascun d'ells, usat per els encarregats de magatzem.
- Etc.

Cada mòdul deu ser instal·lat a l'ERP de forma individual, de manera que es poden utilitzar únicament els mòduls necessaris per a cada negoci, optimitzant el sistema al instal·lar únicament funcionalitats que s'utilitzaran.

Segons Viquipèdia (2022), Odoo té molta història:

“Odoo (conegut anteriorment com a OpenERP i inicialment com a TinyERP) és un sistema d'ERP integrat de codi obert actualment produït per l'empresa belga Odoo S.A.”

Una definició més formal, segons Tomás-Miquel et al. (2008):

En el mercat existeixen una gran quantitat d'ERPs modulars, però el gran avantatge que ofereix Odoo és que és una tecnologia de codi obert. Això implica que el codi del programari pot ser modificat per altres programadors, i no és necessari disposar únicament de la versió base, com indiquen Tomás-Miquel et al. (2008). Així per exemple, en el cas d'aquest TFG, es poden modificar els mòduls base o crear-ne de nous.

Això ha obert la porta a la creació de moltes empreses, que ofereixen un servei a altres empreses personalitzant-los l'ERP al seu gust. Aquest treball té fins a un nom, “Odoo developing”.

1.3 Conceptes d'Odoo

Per a entendre diverses de les solucions que es proposen, així com la manera en la que es desenvolupa el projecte, és necessari entendre alguns conceptes de programació d'Odoo, així com la lògica interna que segueix.

Els conceptes bàsics que es detallen:

- Mòdul: Conjunt de fitxers que implementen una funcionalitat empresarial a l'ERP.
- Model: Model de dades on s'introdueixen tots els camps i les funcions necessàries d'un objecte. En ell es creen les classes de programació.
- Dependència: Existeixen mòduls que depenen dels altres.
- Herència: Heretar un model de dades, i així obtindrà tots els seus camps i funcions més els nous que es vulguen ficar. Utilitzat per a especialitzar (exemple: Treballador → Metge, Infermer) i per a introduir funcionalitats en un model original d'un altre mòdul.
- Camp: Valor que s'introdueix al model de dades. Similar a variable.
- Estat: Camp selector, que pot tenir uns únics valors fixes. Utilitzat per a indicar l'estat en el que es trobe un determinat objecte de la base de dades (exemple: Validat, Entregat, Pagat, Finalitzat).
- Menú: El mòdul disposa de diversos models de dades, que tots junts formen el programari. Per a accedir a les vistes de cada objecte i crear-ne o editar-ne, es crea un menú.
- Vista form: És una vista de l'objecte, en forma de formulari. En ella s'introdueixen les dades necessàries de cada objecte nou que es crea a la base de dades.
- Vista tree: És una vista que ens mostra tots els objectes que hi ha a la base de dades d'un model de dades. Els mostra en forma de llista.
- Vista kanban: És una altra vista que ens mostra els objectes d'un model de dades, però en forma de targeta.
- Traçabilitat: La traçabilitat de missatges consisteix a, en la vista de formulari, incloure informació de les actualitzacions dels valors dels camps. S'introdueixen en aquesta vista, missatges informatius que indiquen l'usuari que realitza l'actualització de valors del camp, el vell i el nou valor del camp.
- Fitxer `__init__.py`: És un fitxer que està present en tots els mòduls. En ell s'importen tots els fitxers `.py` per a carregar la lògica del programari.
- Fitxer `manifest.py`: Es tracta d'un fitxer on s'inclou tota la informació a carregar en temps de compilació. S'inclou el nom del mòdul, l'autor, els fitxers `.xml` a carregar, dades de demostració, etc...

La lògica interna que segueix Odoo es basa en:

- **Productes:** És el model de dades en el que es basa qualsevol bé que proporciona la empresa. Aquest model es troba en un mòdul, anomenat “product”, i és utilitzat per la gran majoria de mòduls. Per exemple, per a fer factures, el mòdul de comptabilitat es relaciona amb el model de dades de producte. És importantíssim per tant, incloure aquest mòdul com a dependència del mòdul que es pretén crear, i heretar d’ell per a especialitzar productes i que així altres mòduls tinguin accés als nous tipus de productes. Exemple: Un llibre serà un tipus de producte.

Trobem dos models:

- **Product.template:** La plantilla del producte.
- **Product.product:** El producte complet, que hereta de la plantilla i se li afegeixen propietats úniques per a diferenciar-lo d’altres productes de la mateixa plantilla. És del product.product d’on s’ha d’heretar per a crear productes especialitzats.

The screenshot shows the Odoo Product Template configuration interface. At the top, there is a search bar for the product name, which is currently empty. To the right of the search bar, there are three summary boxes: 'Movimientos de producto' (0), 'Ventas' (0), and 'Compras' (0). Below the search bar, there are two checked checkboxes: 'Puede ser vendido' and 'Puede ser comprado'. On the right side, there is a status indicator 'Activo'. The main form area is divided into tabs: 'Información General', 'Ventas', 'Compra', 'Inventario', 'Contabilidad', and 'Notas'. The 'Información General' tab is active. It contains several fields: 'Tipo de producto' (Consumible), 'Precio de venta' (1,00), 'Categoría' (All), 'Precio de coste' (0,00), 'Referencia interna', and 'Código de barras'.

Figura 1: Captura de pantalla del product.template d’Odoo, en la versió 11

- Ordres: Les ordres són els models de dades relacionats amb vendes o compres. Cada venda o compra queda registrada en el sistema, per a poder traure la comptabilitat. És per això que si es vol calcular la comptabilitat, traure balanços i en general fer operacions sobre allò que ha venut la empresa, qualsevol operació que es pague o es facture deu ser inclosa com a model de dades, i heretar de l'ordre.

Nuevo

Cliente

Fecha de caducidad

Plazos de pago

Líneas del pedido
Otra información

Producto	Descripción	Ctdad pedida	Precio de Unidad	Impuestos	Subtotal
Añadir un elemento					

Configure los términos y condiciones predeterminados en su configuración de ventas...

Base imponible: 0,00

Impuestos: 0,00

Total: **0,00**

Figura 2: Captura de pantalla del sale.order d'Odoo, en la versió 11.

- **Partners:** Els partners fan referència a les entitats. És un partner qualsevol entitat d'una empresa, un usuari, una companyia, un client, un proveïdor, un empleat...,etc. Tots aquests models de dades hereten del partner. És la forma més senzilla de la que disposa Odoo per a facilitar el desenvolupament software i la lògica interna, ja que així totes les entitats estan relacionades, cosa la qual deu ser permesa (per exemple, una companyia pot ser un client, o un empleat pot ser un usuari...). Totes les entitats que es creen deuen heretar del partner.

Figura 3: Captura de pantalla del res.partner d'Odoo, en la versió 11.

1.4 Motivació

Els ERPs són programaris que ajuden en gran mesura a moltes empreses de producció, però desgraciadament existeixen negocis per als quals no estan pensats. No obstant això, gràcies a l'aparició de l'anomenat "Odoo developing", l'ERP Odoo pot ser modificat d'una gran quantitat de formes. El cel és el límit.

Gràcies a Odoo, negocis que no estaven pensats per a ser implementats en un ERP, ara poden ser-ho.

Ja hem comentat els avantatges que pot tenir per al negoci una correcta implementació d'un ERP, així que ajudar a una empresa de serveis a que pugui implementar un ERP és una gran oportunitat.

1.5 Objectius

Els principals objectius que abasta el desenvolupament d'aquest programari són els següents:

- Permetre la integració d'un ERP en una biblioteca, eliminant els softwares d'ofimàtica en aquestes.
- Elaboració d'un programari que permeti dur a terme la gestió bibliotecària.
- Que aquest programari, siga susceptible d'unir-se amb altres fitxers per a formar un ERP complet.
- Facilitar el treball als bibliotecaris, tant als més novells com als més experts, amb una interfície clara i senzilla.
- Obrir una porta que permeti la millora del negoci de les biblioteques, posant aquesta alternativa a disposició d'aquestes.

1.6 Estructura de la memòria

Centrant-se ara exclusivament en el contingut d'aquesta memòria, inclourà els següents punts:

- Introducció a la problemàtica, on parlarem del model de negoci d'una biblioteca i tractarem de demostrar el per què és susceptible de ser integrat en un ERP, i concretament en Odoo. També parlarem del procés de gestió bibliotecària, ja que és el procés que tractarem d'adaptar a software durant el desenvolupament d'aquest treball.
- Anàlisi de requeriments, on tractarem primer requeriments necessaris que ha d'abastar el programari realitzat. Es parlarà de casos d'ús on es mostraran casos d'interacció entre usuari i sistema, indicant requeriments que té que complir el sistema. Després es plantejaran possibles solucions al problema, analitzant cadascuna d'elles, i finalment es seleccionarà una.
- Disseny de la solució. En aquest punt, tractarem de dissenyar sobre el paper la solució elegida, abans de desenvolupar-la. Es parlarà de les tres capes del model de desenvolupament software de capes:
 - Presentació (interfície).
 - Negoci (lògica).
 - Dades.
- Desenvolupament de la solució, on es parlarà de les ferramentes utilitzades per a fer el treball, l'estructura dels fitxers, i finalment es parlarà de com s'ha desenvolupat la solució. Aquesta part serà la més extensa de la memòria, ja que es parlarà de tots els fitxers creats, els camps i les funcions de cada model de dades.
- Implementació de la solució. En aquest apartat, es plantejarà la implementació i l'ús que pot donar un bibliotecari del mòdul elaborat. També es parlarà del manteniment que se li pot donar al projecte, i el temps de vida que pot tindre, per a analitzar el producte a llarg termini.

- Conclusions.
- Bibliografia.
- Annexos, on s'adjunten una gran part dels codis realitzats.

1.7 Relació amb assignatures

Finalment, remarcar que analitzant profundament l'índex, s'han extret conclusions respecte a la relació d'aquest TFG amb les assignatures vistes en la carrera. En aquest TFG, s'inclou contingut de les següents assignatures d'enginyeria informàtica:

- Fonaments d'organització d'empresa.
- Llenguatges, tecnologies i paradigmes de la programació.
- Interfícies persona computador.
- Enginyeria del software.
- Bases de dades i sistemes de la informació.
- Tecnologia de sistemes de la informació en xarxa.
- Anàlisi de requisits de negoci.
- Gestió de tecnologies de la informació.
- Sistemes integrats d'informació a les organitzacions.
- Sistemes d'informació estratègics.
- Disseny i gestió de base de dades.
- Models de negoci i àrees funcionals de l'organització.

2. Introducció a la problemàtica

Per a entendre millor la problemàtica que es planteja resoldre, hem d'entendre primer d'on sorgeix la problemàtica. Aquesta apareix a partir del model de negoci de les biblioteques municipals.

2.1 Model de negoci d'una biblioteca

És imprescindible comprendre com és el funcionament intern d'una biblioteca per tal de saber si es pot implementar un ERP en aquesta. Degut a aquesta necessitat, analitzarem els nou elements clau del model de negoci d'una biblioteca, i comprovarem si es poden integrar els processos en un ERP utilitzant la tecnologia d'Odoo.

Els 9 elements clau del model de negoci d'una biblioteca municipal són:

- **Socis clau:** Es tracta dels socis i ajuda externa a la institució que rep l'organització o negoci. En el cas de les biblioteques municipals són:
 - Ajuntament: Encarregat d'aportar ajudes econòmiques si són necessàries.
 - Estat: També subvenciona les instal·lacions i material, ajudant en menor mesura.
 - Ciutadans: Alguns d'ells formen associacions de veïns, encarregats de donar productes a la biblioteca.

- **Activitats clau:** Són les activitats més importants que desenvolupa una biblioteca municipal. En les quals podem trobar:
 - Donar d'alta un llibre: Implica donar d'alta un llibre en la base de dades de la biblioteca. L'exemplar pot haver sigut obtingut de moltes maneres.
 - Donar d'alta un membre: Implica donar d'alta un membre en la base de dades de la biblioteca.
 - Donar d'alta un ordinador: Implica donar d'alta un ordinador en la base de dades de la biblioteca. Recordem que les biblioteques també lloguen ordinadors portàtils.
 - Prestar un llibre: Aquesta activitat consta en llogar un llibre a un membre de la biblioteca. El llibre es lloga amb una data de retorn, que dependrà del temps d'alquilar que tinga cada membre.
 - Prestar un ordinador: El mateix que el procés anterior, però amb ordinadors.
 - Reservar un llibre: Esta activitat busca reservar un exemplar a un membre durant unes determinades dades. No és el mateix que llogar, ja que el llibre segueix a la biblioteca. També es pot reservar un llibre ja llogat per a unes dates futures. La reserva té una duració màxima de 15 dies.

- Reservar un ordinador: El mateix que el procés anterior, però amb ordinadors.

Totes aquestes activitats són desenvolupades per els bibliotecaris, creant el procés de gestió bibliotecària. No obstant, existeixen més processos, que inclouen activitats com:

- Anàlisi de préstecs: Analitzar la quantitat de préstecs, les edats del membres, i traure estadístiques favorables per a estudis. Aquesta activitat està realitzada per l'Ajuntament una vegada la biblioteca li transmet les dades.
 - Compra de llibres: Adquisició d'exemplars, duta a terme per l'administrador.
 - Gestió d'inventari: Dur a terme un anàlisi profund de l'inventari. Així podem saber per exemple, si un llibre que el sistema posa que tenim, es troba perdut. Una vegada siga finalitzat l'anàlisi, fer els canvis necessaris a la base de dades (aquesta part la realitza el bibliotecari). Aquesta activitat és periòdica, es realitza cada mes.
- **Recursos clau**: Són els recursos dels que disposa una biblioteca municipal, i que són necessaris per a dur a terme les activitats clau. Són principalment:
- Recursos humans: Personal, en la seua majoria bibliotecaris, encarregats de dur a termes els processos de negoci.
 - Recursos materials: Entre els quals es troben els ordinadors dels que fan ús els empleats, així com els béns els quals són llogats (llibres i ordinadors).
 - Instal·lacions: La biblioteca municipal pot tindre diverses sucursals, per exemple en el cas d'Alcoi, la de la Zona Nord, Santa Rosa i el Centre. Cadascuna disposa de unes instal·lacions.
 - Capital: Es tracta d'una organització subvencionada per l'Ajuntament, ja que ofereix un servei públic, el préstec de llibres i en alguns casos de computadores.
- **Proposta de valor**: La proposta de valor que ofereix una biblioteca pública consisteix en millorar indirectament la qualitat de vida dels ciutadans. El que busca, principalment, és oferir al ciutadà un préstec d'un llibre de manera completament gratuïta, la qual cosa permet que qualsevol persona, independentment de les seues condicions econòmiques, pugua passar una bona estona amb un llibre.

Amb el pas dels anys, noves propostes de valor han sorgit per al negoci, com el préstec d'ordinadors per a aquelles persones que no pugen tindre un i el requereixen, o l'oferta d'un espai on poder gaudir d'una lectura o una sessió d'estudi.

- **Relacions amb el client:** Les biblioteques es relacionen constantment amb els seus clients, oferint-los un servei post-registre mitjançant el qual els clients poden llogar llibres, ordinadors, i fer ús de les instal·lacions. Els clients seran atesos tantes vegades com vulguen fer ús del servei públic, donant lloc així a algunes de les activitats clau anteriorment mencionades.
- **Canals:** Les biblioteques municipals arriben al client a través de publicitat, tant a les xarxes socials com en cartells, repartits per tota la ciutat.
- **Segments de client:** Els clients als que va dirigit aquest model de negoci, són tot tipus de persones. A una biblioteca pot acudir qualsevol persona amb diferents necessitats, com la recerca d'un llibre per a llegir, un ordinador per a treballar, o un espai de treball. A més, es tracta d'un model de negoci en el que el client final no paga per el servei, ja que és un servei públic ofert per l'Ajuntament. Per la qual cosa, la biblioteca va dirigida a un mercat de masses, on els clients són tots els ciutadans que resideixen a la ciutat.
- **Estructura de costos:** Els costos de mantindre aquest model de negoci es divideixen en dos:
 - Pagar als bibliotecaris, encarregats de dur a terme activitats claus per al negoci. Cost fixe.
 - Pagar els diferents establiments, on es poden establir diferents sucursals de la biblioteca. En el cas d'Alcoi, una per barri. Cost fixe.
 - Pagar part del material (llibres i ordinadors) que entra en la biblioteca per a ser llogat. Cost variable.

- **Fonts d'ingressos:** Les biblioteques tenen un model de negoci mitjançant el qual no obtenen ingressos per part dels clients de forma directa, ja que és un servei gratuït. No obstant, per tal de finançar el projecte, l'Ajuntament finança les biblioteques municipals amb part dels diners cobrats d'impostos, destinats a serveis públics.

Socis clau <ul style="list-style-type: none"> ● Ajuntament (realment els propietaris del negoci). ● Estat. ● Ciutadans. 	Activitats clau <ul style="list-style-type: none"> ● Donar d'alta un llibre. ● Donar d'alta un ordinador. ● Donar d'alta un membre. ● Reservar un llibre. ● Reservar un ordinador. ● Prestar un llibre ● Prestar un ordinador. ● Anàlisi de préstecs. ● Compra de llibres. ● Gestió d'inventari. 	Propostes de valor <ul style="list-style-type: none"> ● Oferir un servei gratuït al ciutadà de préstec de llibres. ● Oferir al ciutadà un espai on poder fer activitats en silenci. ● Oferir un servei al ciutadà de préstec d'ordinadors gratuït. 	Relacions amb els clients <ul style="list-style-type: none"> ● Servei post-inscripció. 	Segments de client <ul style="list-style-type: none"> ● Ciutadans d'una ciutat. ● Persones amb poques prestacions econòmiques.
	Recursos clau <ul style="list-style-type: none"> ● Recursos humans. ● Recursos materials. ● Instal·lacions. ● Capital. 		Canals <ul style="list-style-type: none"> ● Xarxes socials. ● Propaganda. 	
Estructura de costos Fixes: <ul style="list-style-type: none"> ● Pagar les nòmines dels bibliotecaris i empleats. ● Pagar instal·lacions de les sucursals. Variables: <ul style="list-style-type: none"> ● Pagar llibres i ordinadors. 		Fonts d'ingressos <ul style="list-style-type: none"> ● Impostos públics. 		

Figura 4: Model Canvas d'una biblioteca. Font: elaboració pròpia

A més, les biblioteques municipals disposen de la següent estructura jeràrquica:

- Ajuntament: Controla el negoci subvencionant-lo en general.
 - Sucursals: Establiment on es realitza el negoci. Hi ha diverses, cadascuna d'elles actuant de manera independent, però totes compartint el mateix propietari i fent les mateixes activitats. Dintre de les sucursals hi ha productes (llibres i ordinadors) que es presten a membres de la biblioteca durant un període de temps.

Després de vore els elements claus d'aquest model de negoci, i entendre que és el que fa una biblioteca i com ho fa, vegem que la integració d'un ERP a aquest negoci és més que possible. La integració dels processos en aquest cas, permetria:

- Unió de les diferents sucursals, fent que tinguen una visió global de la biblioteca.
- Facilitar la comunicació amb els socis del negoci.
- Tindre la informació més organitzada per a accedir a ella, i facilitar el desenvolupament de les activitats claus.

Aleshores, la idea mereix l'atenció, ja que la implantació d'un ERP a una biblioteca té altes probabilitats de millorar el negoci.

2.2 Desenvolupament del mòdul en Odoo

Pel que fa a les activitats clau d'una biblioteca, Odoo disposa de mòduls bàsics que poden ser utilitzats:

- El mòdul base de compra, permet gestionar la compra de llibres (productes) a editorials (proveïdors).



Figura 5: Captura de pantalla del mòdul de compra d'Odoo, en la versió 11. Font: elaboració pròpia

- El mòdul base de comptabilitat podrà dur a terme la gestió econòmica, que en cas d'una biblioteca, almenys deurà registrar els gastos fets amb els materials.

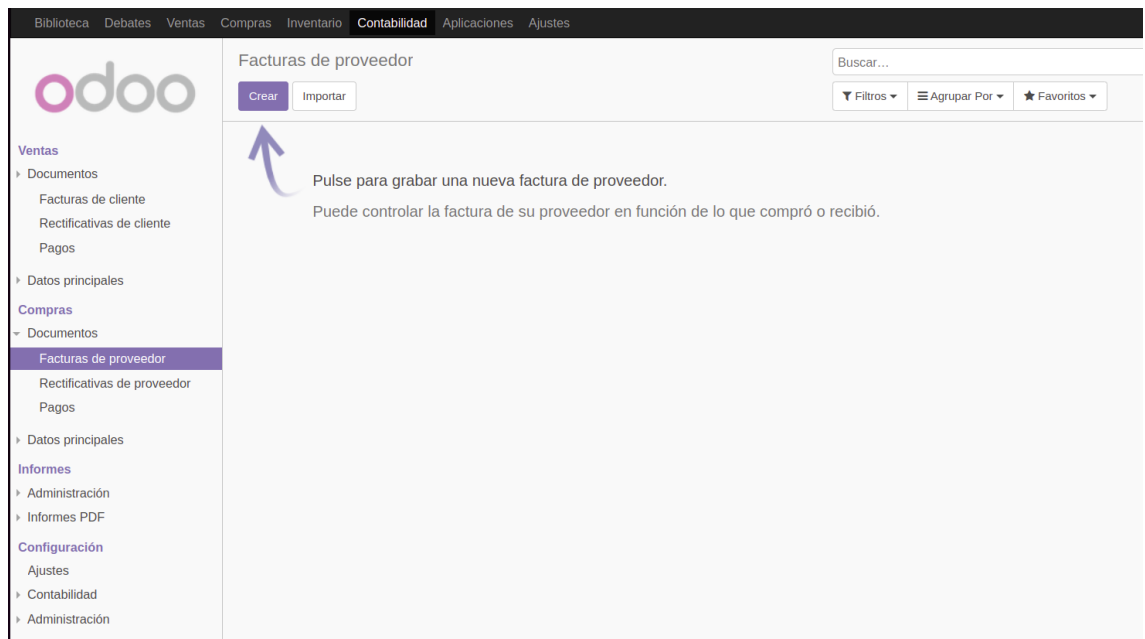


Figura 6: Captura de pantalla del mòdul de compra d'Odoo, en la versió 11. Font: elaboració pròpia

Però per a dur la gestió bibliotecària, no hi cap mòdul que pugui fer aquesta funcionalitat, i això és el que impedeix a les biblioteques implementar un ERP amb Odoo.

En aquest punt és on arribem a la problemàtica a resoldre. Per tal de que les biblioteques puguin implementar Odoo, necessiten un mòdul encarregat de dur a terme la gestió bibliotecària del qual el codi font d'Odoo no disposa, i això és el que es desenvoluparà en aquest projecte software. Es crearà un mòdul d'Odoo que permetrà dur a terme la funcionalitat dels bibliotecaris, i així es permetrà a les biblioteques usar aquesta tecnologia.

2.3 Procés de gestió bibliotecària

Abans de dur a terme l'anàlisi de requeriments per tal de saber com desenvolupar el programari, seria convenient conèixer com és el procés de la gestió bibliotecària, que és el que es vol implementar en software, i a partir d'ahí especificar els requeriments.

Aquest procés de negoci és realment un superprocés, format per altres tres processos els quals es repeteixen constantment.

Aquests processos son:

- Llogar un llibre o ordinador.
- Reservar un llibre o ordinador.
- Donar d'alta un llibre, ordinador o membre.

A continuació, s'inclou un diagrama BPMN de cadascun d'ells:

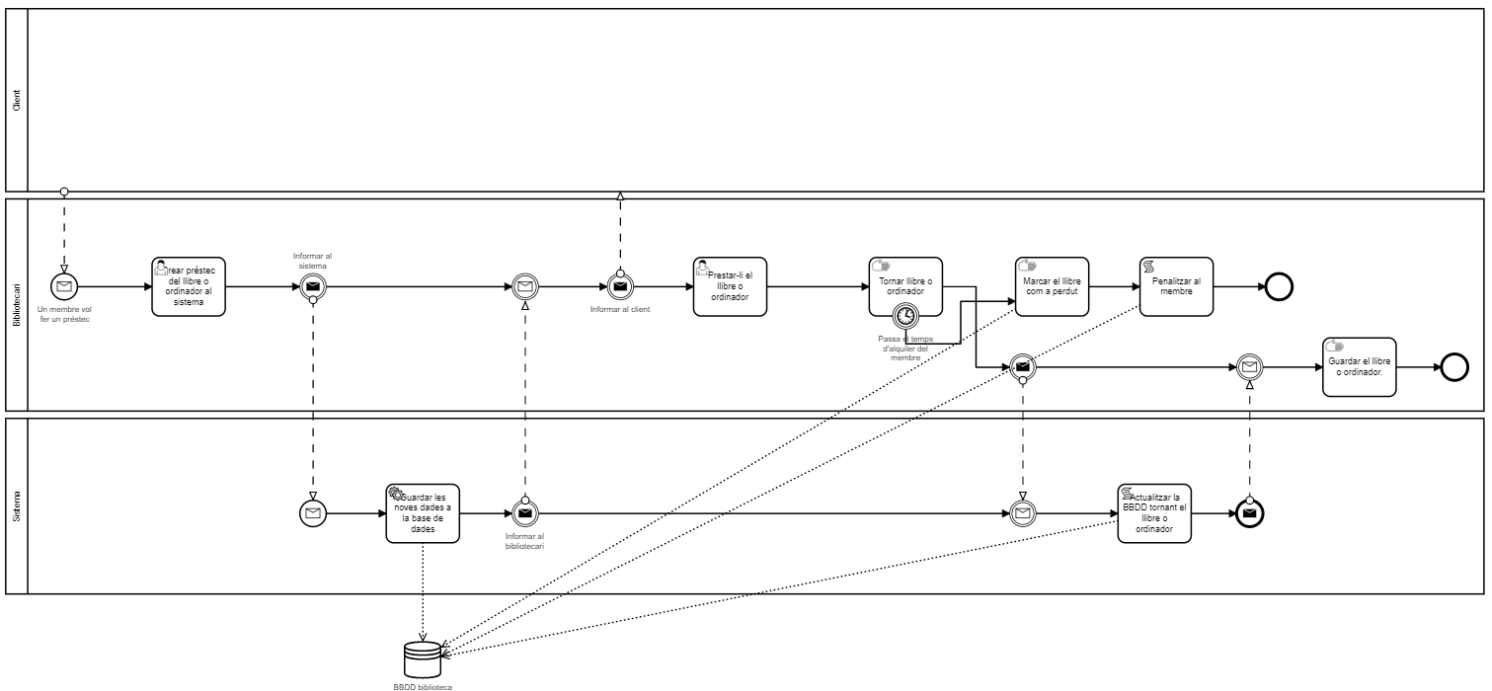


Figura 7: Diagrama BPMN. Llogar un llibre o ordinador. Font: elaboració pròpia a partir de Camunda Modeler.

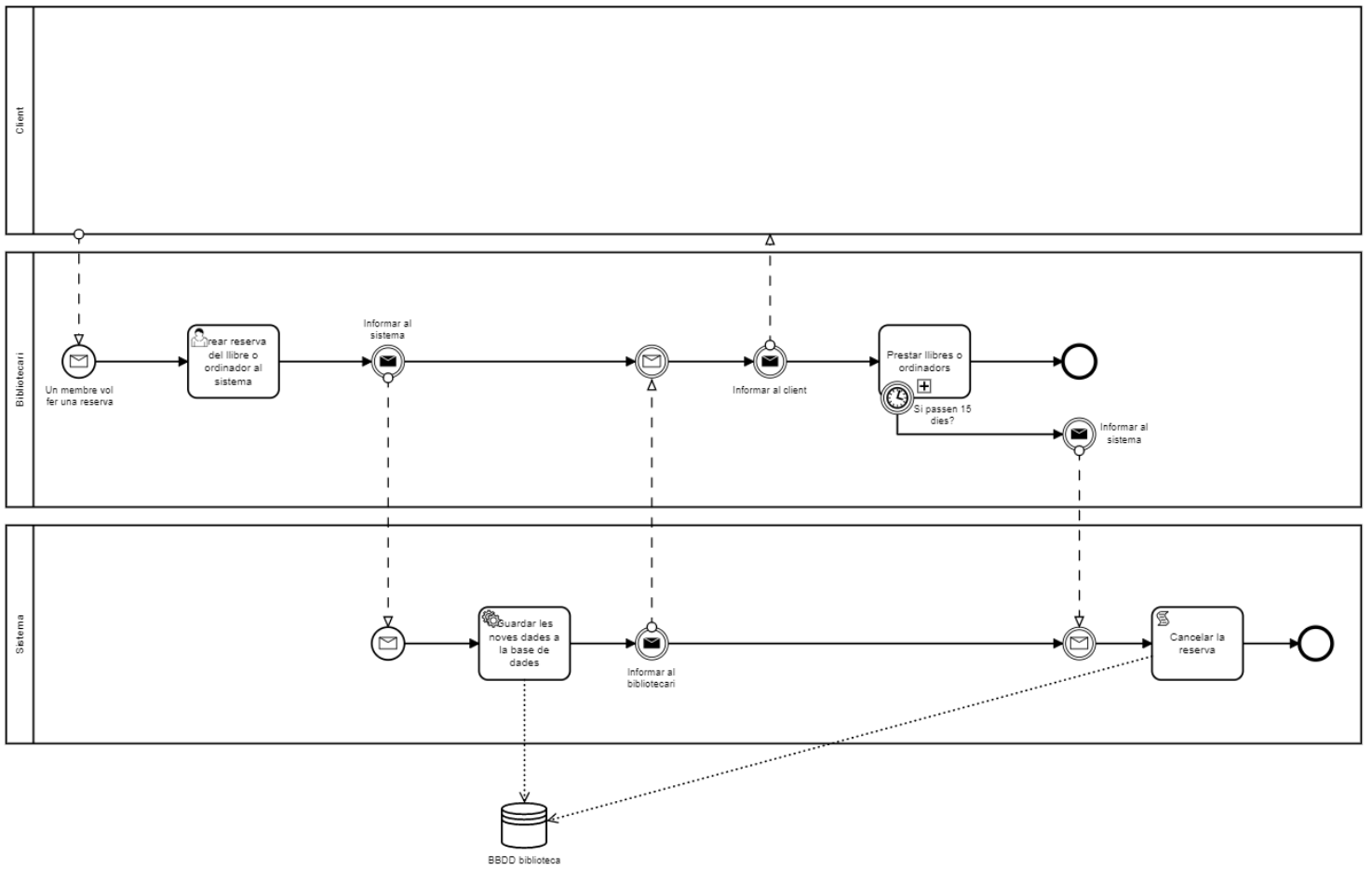


Figura 8: Diagrama BPMN. Reservar un llibre o ordinador. Font: elaboració pròpia a partir de Camunda Modeler.

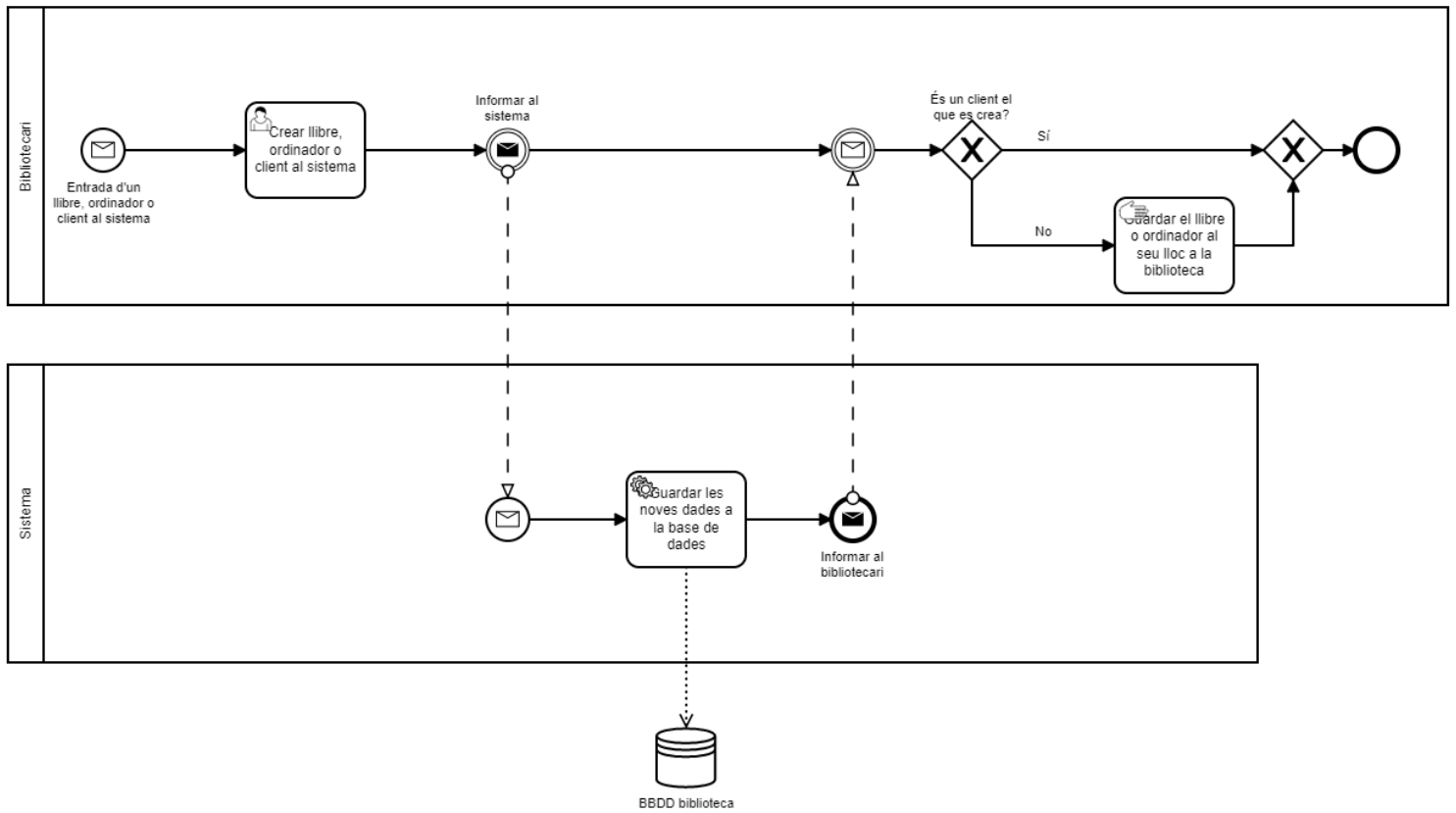


Figura 9: Diagrama BPMN. Donar d'alta un llibre, ordinador o membre. Font: elaboració pròpia a partir de Camunda Modeler.

3. Anàlisi de requeriments

Per a poder assegurar l'èxit en el desenvolupament software d'aquest projecte resulta necessari identificar primer els requeriments. Dit d'altra forma, hem de saber que deu fer l'aplicació i com la utilitzarà l'usuari final. Després hem de plantejar possibles formes de resoldre la problemàtica, i triar-ne una.

3.1 Especificació de requeriments

Els principals requeriments que deu solucionar el mòdul d'Odoo són els següents:

- Requeriments funcionals:

El mòdul deu tindre les següents seccions:

- Secció de llibres:
 - Els usuaris podran crear, duplicar i editar un llibre.
 - Els usuaris podran relacionar els llibres amb els autors que requereixen.
 - Els usuaris podran relacionar els llibres amb les editorials que requereixen.
 - Els usuaris podran relacionar els llibres amb les sucursals que requereixen.
 - Els usuaris podran relacionar els llibres amb les categories que requereixen.
 - Del llibre es deu conèixer: Nom, codi, autor (opcional), editorial (opcional), sucursal, nom de cerca i estat del llibre.
- Secció d'autors:
 - Els usuaris podran crear, duplicar i editar un autor.
 - Els usuaris podran observar els llibres que ha escrit un autor.
 - De l'autor s'ha de conèixer: Nom, DNI, llibres publicats, email i país (opcional).
- Secció d'editorials:
 - Els usuaris podran crear, duplicar i editar un llibre.
 - Els usuaris podran observar els llibres que ha publicat una editorial.
 - De la editorial s'ha de conèixer: Nom, llibres publicats, email i país (opcional).
- Secció de sucursals:
 - Els usuaris podran crear, duplicar i editar una sucursal.
 - Els usuaris podran observar els llibres que hi ha en cada sucursal.
 - De la sucursal s'ha de conèixer: Nom, codi, llibres publicats, email i país (opcional).
- Secció de categories:
 - Els usuaris podran crear, duplicar i editar una categoria.
 - Els usuaris podran crear subcategories.

- Secció d'ordinadors:
 - Els usuaris podran crear, duplicar i editar un ordinador.
 - Els usuaris podran relacionar els ordinadors amb les sucursals que requereixen.
 - De l'ordinador es deu conèixer: Nom, codi, sucursal, i estat de l'ordinador.
- Secció de membres:
 - Els usuaris podran crear, duplicar i editar un membre.
 - Els usuaris podran observar els llibres que hi ha en cada sucursal.
 - Del membre s'ha de conèixer: Nom, DNI, nombre de llibres llogats, email, temps de préstec i país (opcional).
- Secció de préstecs:
 - Els usuaris podran crear, duplicar i editar un préstec.
 - Els usuaris podran relacionar els préstecs amb els llibres que requereixen.
 - Els usuaris podran relacionar els préstecs amb els membres que requereixen.
 - Del préstec s'ha de conèixer: Num, llibre, membre, temps de préstec i estat.
- Secció de reserves:
 - Els usuaris podran crear, duplicar i editar una reserva.
 - Els usuaris podran relacionar les reserves amb els llibres que requereixen.
 - Els usuaris podran relacionar les reserves amb els membres que requereixen.
 - De la reserva s'ha de conèixer: Num, llibre, membre, data d'inici de reserva, data de final de reserva i estat.
- Requeriments no funcionals:
 - Accés:
 - L'usuari i l'administrador podran accedir al sistema des de qualsevol dispositiu connectat a la xarxa de la biblioteca.
 - Usabilitat i rendiment:
 - El sistema ha d'usar una interfície senzilla.
 - El programari ha d'estar optimitzat, i no persistir o duplicar dades.
 - El sistema ha d'informar a l'usuari de les coses que pot i les que no pot fer.
 - El menú dels models de dades ha d'estar estructurat.

3.2 Especificació de casos d'ús i permisos

Per tal d'entendre millor els usos que podrien fer els usuaris sobre el sistema, s'introdueix un diagrama de casos d'ús. Interfereixen tres tipus d'actors diferents: membre, bibliotecari i administrador:

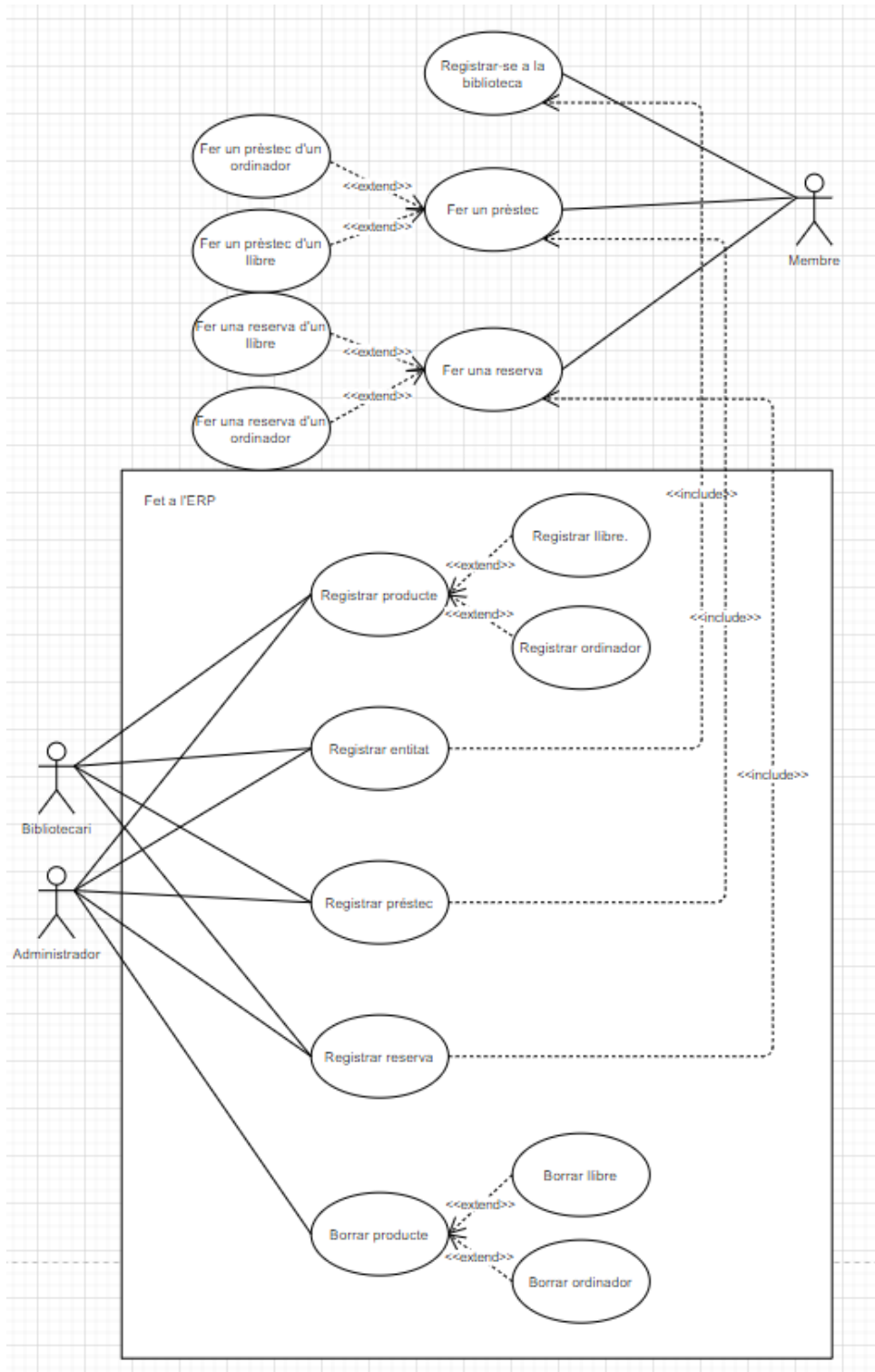


Figura 10: Diagrama de casos d'ús. Font: elaboració pròpia a partir de draw.io

No obstant això, els actors que ens interessen són els que interactuen amb el sistema. És per això que, per a analitzar individualment cada cas d'ús, no tractarem els del membre perquè aquest no interactua amb el sistema software, però és important saber el paper d'aquest ja que molts casos d'ús depenen de la seua interacció amb el bibliotecari/administrador.

○ **Casos d'ús del bibliotecari:**

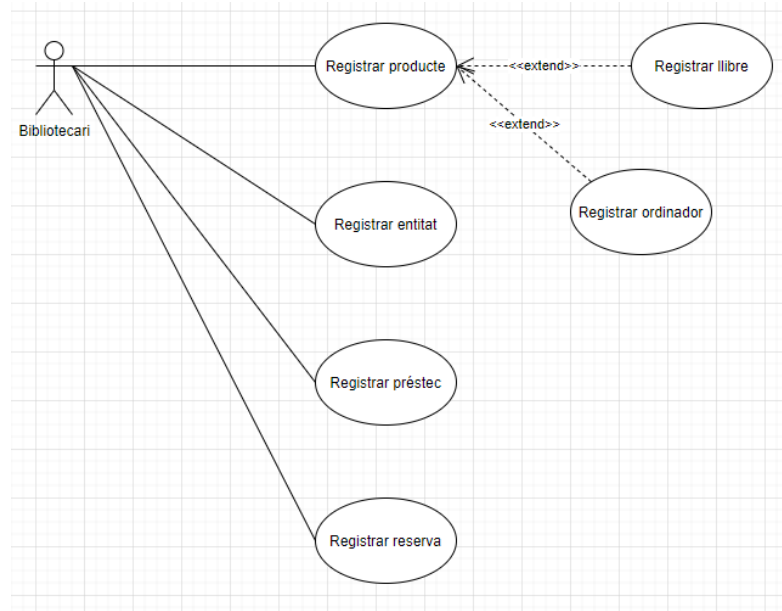


Figura 11: Diagrama de casos d'ús. Actor: bibliotecari. Font: elaboració pròpia a partir de draw.io

Cas d'ús: Registrar producte	
Precondició: Producte no registrat a la biblioteca.	
Postcondició: Producte registrat a la biblioteca.	
Usuari:	Sistema:
Amb l'arribada d'un producte (llibre o ordinador) a la biblioteca, l'usuari introdueix les dades de l'objecte i el crea.	
	Emmagatzema al sistema el producte i li genera un identificador.

Taula 1: Cas d'ús: Registrar producte. Font: elaboració pròpia

Cas d'ús: Registrar entitat	
Precondició: Membre no registrat a la biblioteca.	
Postcondició: Membre registrat a la biblioteca.	
Usuari:	Usuari:
Introdueix les dades del nou membre al formulari i el crea.	
	Emmagatzema al sistema el membre i li genera un identificador.

Taula 2: Cas d'ús: Registrar entitat. Font: elaboració pròpia

Cas d'ús: Registrar préstec	
Precondició: Un membre sol·licita un préstec d'un producte.	
Postcondició: El membre disposa del producte per un temps.	
Usuari:	Sistema:
Introdueix les dades del préstec al formulari , emparellant el producte amb el membre.	
	Emmagatzema al sistema el préstec i no permet que el producte es torne a llogar fins que torne a estar disponible.

Taula 3: Cas d'ús: Registrar préstec. Font: elaboració pròpia

Cas d'ús: Registrar reserva	
Precondició: Un membre sol·licita una reserva d'un producte.	
Postcondició: El membre disposa de la reserva per un temps.	
Usuari:	Sistema:
Introdueix les dades de la reserva al formulari , emparellant el producte amb el membre.	
	Emmagatzema al sistema la reserva i no permet que el producte es torne a reservar o llogar fins que torne a estar disponible.

Taula 4: Cas d'ús: Registrar reserva. Font: elaboració pròpia

- **Casos d'ús de l'administrador:** Són els mateixos que els de l'usuari comú, amb la diferència de què l'administrador pot eliminar productes. És per això que analitzarem únicament aquest cas ara.

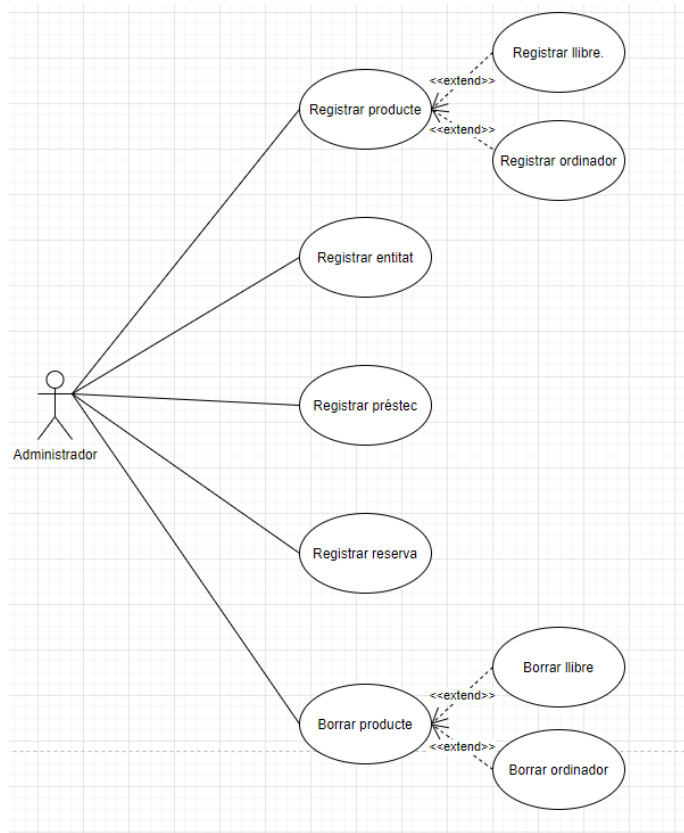


Figura 12: Diagrama de casos d'ús. Actor: administrador. Font: elaboració pròpia a partir de draw.io

Cas d'ús: Eliminar producte	
Precondició: Un producte es deixa de posseir.	
Postcondició: No queden registres del producte a la base de dades.	
Usuari:	Sistema:
Accedeix al formulari del producte i l'elimina.	
	Elimina el producte de la base de dades.

Taula 5: Cas d'ús: Eliminar producte. Font: elaboració pròpia

3.3 Anàlisi de solucions

Es tracta d'un projecte software, i com a tal, pot tenir diverses solucions possibles. Existeixen diverses formes de programar el mòdul de gestió bibliotecària en Odoo. A continuació mostrarem 6 possibles formes de plantejar el problema, argumentant el pros i els contres.

Cal dir que totes les solucions comparteixen avantatges, com desenvolupar-se a través d'una metodologia àgil o tindre una interfície senzilla i clara. Eixos avantatges no s'argumenten en les taules següents, ja que es busca diferenciar les solucions.

També cal comprendre que per tal de crear un mòdul que siga compatible amb els altres, és necessari incloure dependències. Per exemple, per tal que el mòdul de gestió bibliotecària fora compatible amb el de vendes, els dos deurien de considerar els llibres i els ordinadors com a la mateixa classe, producte. És per això que en Odoo l'herència de classes és molt important. Les classes genèriques que més s'utilitzen són el producte, el partner (entitat) i l'ordre (de compra o venda).

- **Crear un únic mòdul independent.** Aquesta solució consta en crear un únic mòdul per a resoldre la problemàtica, amb models de dades completament independents del nucli d'Odoo.

Avantatges	Inconvenients
No cal conèixer el funcionament intern d'Odoo ni quina és la lògica que segueix.	Falta de compatibilitat amb els altres mòduls, per no heretar del producte, ni partner ni de l'ordre de venda.
	S'han d'instal·lar totes les funcionalitats, encara que no es vulguen.
	S'haurien de modificar els mòduls de compra i comptabilitat per tal de posar-ho en pràctica.
	Es tornen a fer lògiques ja implementades a Odoo.

Taula 6: Comparativa opció 1. Font: elaboració pròpia

Degut a tots aquests inconvenients, i sobretot per tindre que modificar els mòduls bàsics, aquesta opció és inviable i es descarta.

- **Crear un únic mòdul dependent del producte.** Esta alternativa passa per buscar un solució també integrant tot el procés de gestió bibliotecària en un únic mòdul, però fent que els llibres i els ordinadors depenguin del model de dades del producte.

Avantatges	Inconvenients
És compatible amb algunes funcionalitats dels mòduls per heretar del producte.	Falta de compatibilitat amb els altres mòduls, per no heretar del partner i de l'ordre de venda.
Estalviem reutilitzar codi d'Odoo pel que fa als llibres i ordinadors.	S'han d'instal·lar totes les funcionalitats, encara que no es vulguen.
	Es tornen a fer lògiques ja implementades a Odoo.
	Cal conèixer el funcionament intern d'Odoo i quina és la lògica que segueix.
	S'haurien de modificar els mòduls de compra i comptabilitat per tal de posar-ho en pràctica.

Taula 7: Comparativa opció 2. Font: elaboració pròpia

Vegem que aquesta solució és una mica millor, però seguim havent de modificar els mòduls base per tal que funcione el mòdul en l'ERP, pel que també queda descartada.

- **Crear un únic mòdul dependent del producte i del partner.** En esta opció es planteja usar un únic mòdul en el que estiga integrada tota la funcionalitat, però que a més d'heretar del producte els llibres i els ordinadors, també s'herete del partner les entitats (membres, autors, editorials i sucursals).

Avantatges	Inconvenients
És compatible amb moltes funcionalitats dels mòduls per heretar del producte i partner.	Falta de compatibilitat amb els altres mòduls, per no heretar de l'ordre de venda.
Estalviem reutilitzar codi d'Odoo pel que fa als llibres i ordinadors i entitats.	S'han d'instal·lar totes les funcionalitats, encara que no es vulguen.
	Es tornen a fer lògiques ja implementades a Odoo.

Taula 8: Comparativa opció 3. Font: elaboració pròpia

Aquesta opció ja és acceptable, degut a que ja no devem de modificar les mòduls base.

- **Crear dos mòduls dependents del producte i del partner.** Esta alternativa és igual que la anterior, però aquesta vegada dividint la funcionalitat del procés en dos mòduls, un dedicat a gestionar l'inventari de llibres, ordinadors i membres, i l'altre dedicat als préstecs i les reserves.

Avantatges	Inconvenients
És compatible amb moltes funcionalitats dels mòduls per heretar del producte i partner.	Falta de compatibilitat amb els altres mòduls, per no heretar de l'ordre de venda.
Estalviem reutilitzar codi d'Odoo pel que fa als llibres i ordinadors i entitats.	Es tornen a fer lògiques ja implementades a Odoo.
No és necessari instal·lar totes les funcionalitats.	

Taula 9: Comparativa opció 4. Font: elaboració pròpia

Vegem que en aquesta opció tenim un avantatge més que en l'anterior, ja que ara podem no activar totes les funcionalitats si no volem.

- **Crear un únic mòdul dependent del producte, partner i l'ordre:** En aquesta solució es planteja utilitzar un únic mòdul on estiga tota la funcionalitat integrada, i per a dur-la a terme herete del producte, el partner i l'ordre.

Avantatges	Inconvenients
És compatible amb moltes funcionalitats dels mòduls per heretar del producte, partner i ordre.	S'han d'instal·lar totes les funcionalitats, encara que no es vulguen.
Estalviem reutilitzar codi d'Odoo pel que fa als llibres i ordinadors, entitats i ordres (reserves i préstecs).	Els préstecs i reserves no són exactament ordres de venda, i es consideren com a tal.

Taula 10: Comparativa opció 5. Font: elaboració pròpia

Aquesta opció té l'avantatge de poder ser compatible amb més mòduls bàsics (encara que no els necessitem), però el concepte de préstec i reserva es confon amb ordre de venda.

- **Crear dos mòduls dependents del producte, del partner i de l'ordre.**
Aquesta opció és igual que la anterior, però repartint la funcionalitat del procés en dos mòduls, un dedicat a gestionar l'inventari de llibres, ordinadors i membres, i l'altre dedicat als préstecs i les reserves.

Avantatges	Inconvenients
És compatible amb moltes funcionalitats dels mòduls per heretar del producte, partner i l'ordre.	Els préstecs i reserves no són exactament ordres de venda, i es consideren com a tal.
Estalviem reutilitzar codi d'Odoo pel que fa als llibres i ordinadors, entitats i ordres (reserves i préstecs).	
No és necessari d'instal·lar totes les funcionalitats.	

Taula 11: Comparativa opció 6. Font: elaboració pròpia

Vegem que en aquesta opció tenim un avantatge més que en l'anterior, ja que ara podem no activar totes les funcionalitats si no volem.

3.4 Elecció d'una solució

Dintre de les quatre últimes opcions (ja que les dos primeres ja queden descartades), s'ha considerat que la opció de fer dos mòduls és millor que fer un únic mòdul. Això és degut a que dona més llibertat a l'usuari final, ja que si aquest vol, pot instal·lar únicament una part de la funcionalitat.

L'estructura consistiria en dos mòduls:

- Un mòdul on estiga implementada la funcionalitat de crear llibres, ordinadors, membres, autors, categories, editorials i sucursals.
- Un segon mòdul, dependent del primer (necessari tindre instal·lat el primer per a instal·lar-lo), on es duga a terme els préstecs i les reserves, a més d'algunes modificacions a les funcionalitats anteriors.

Per fer una completa gestió bibliotecària serà necessari tindre els dos mòduls instal·lats, però mai se sap quines poden ser les necessitats d'alguns clients, i si no necessiten gestionar préstecs o reserves els resultaria més òptim.

Dit la qual cosa, ens queden dues opcions. Dos mòduls que hereten del producte i partner, o dos mòduls que hereten del producte, partner i l'ordre.

Si bé és cert que heretant l'ordre i utilitzant-la com a préstec i reserva estalviem refer alguna lògica, estaríem implementant una lògica equivocada. És a dir, un llibre sí que pot ser considerat un producte, i un membre també deu ser considerat un partner, però un préstec no pot ser considerat una entitat, ja que la lògica varia massa.

Per la qual cosa, s'ha decidit que la manera en la que es desenvoluparà el mòdul serà seguint la proposta 4, crear dos mòduls dependents del producte i del partner. Es treballarà amb la versió 11 d'Odoo, degut a que és la versió més utilitzada a les empreses.

4. Disseny de la solució

Abans de passar al desenvolupament del projecte, una vegada especificats els requeriments, és necessari dissenyar la solució.

Per a dissenyar el projecte software, es seguirà el model de capes.

4.1 Capa de presentació

En aquesta capa parlem de la part que veu l'usuari de l'aplicació, és a dir, la interfície.

Les vistes finals seran fulles de formularis on es trobarà tota la informació disponible de l'objecte per a editar-la, com qualsevol vista d'Odoo.

L'usuari final no sabrà com està implementada la lògica, ell soles veurà la interfície, així que deu ser atractiva i senzilla d'entendre.

Per tal de comprendre com seran les vistes finals, estaran basades en el següent (amb els dos mòduls instal·lats, gestió bibliotecària i préstecs):

- Vista de llibres:

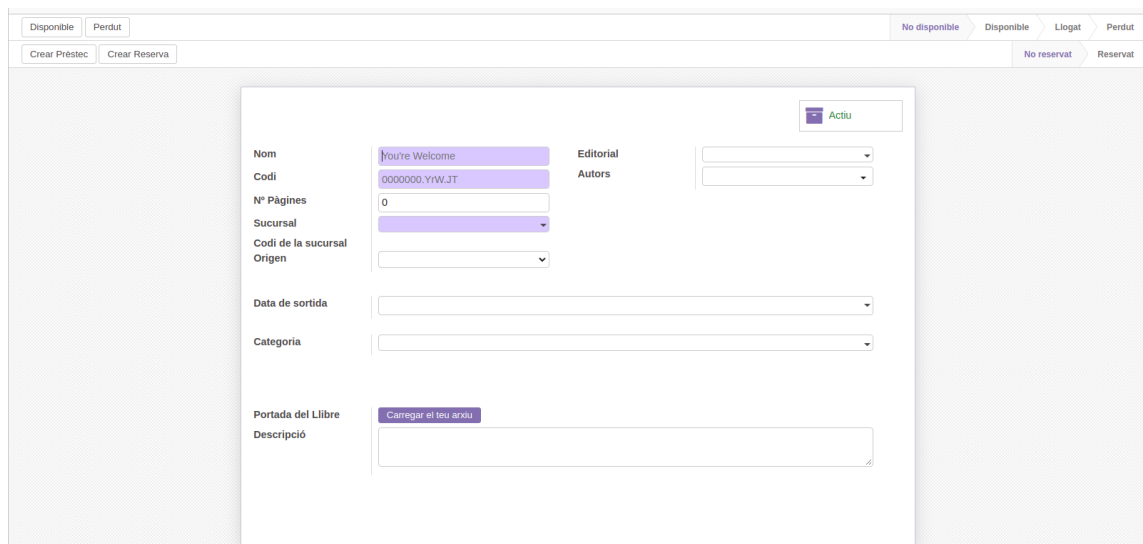
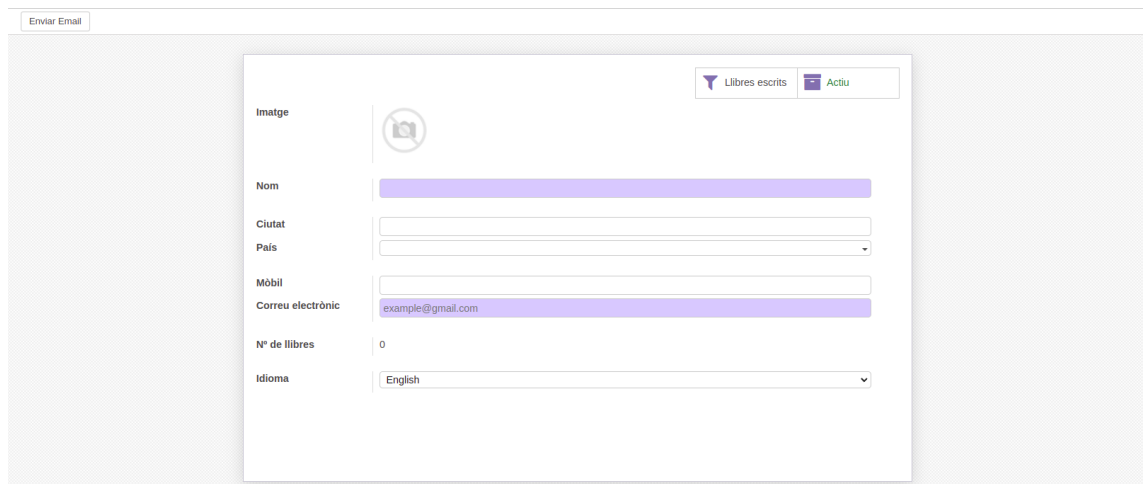
Aquesta imatge mostra una captura de pantalla d'un formulari de creació de llibres a Odoo 11. El formulari està dividit en seccions amb camps d'entrada i botons d'acció. A la part superior hi ha botons per a l'estat del llibre (Disponible, Perdut) i per a la gestió de préstecs (Crear Préstec, Crear Reserva). El formulari principal inclou camps per a: Nom (amb el valor 'You're Welcome'), Codi (amb el valor '000000.YrW.JT'), N° Pàgines (amb el valor '0'), Sucursal (amb una llista desplegable), Codi de la sucursal i Origen (amb una llista desplegable), Data de sortida (amb una llista desplegable), Categoria (amb una llista desplegable) i Portada del Llibre (amb un botó 'Carregar el teu arxiu'). A la dreta del formulari hi ha camps per a Editorial i Autors (amb llistes desplegables) i un botó 'Actiu' a la part superior dreta.


Figura 13: Captura de pantalla del prototip d'interfície del llibre a Odoo 11. Font: elaboració pròpia

○ Vista d'autors:



Enviar Email

Libres escrits Actiu

Imatge 

Nom

Ciutat

País

Mòbil

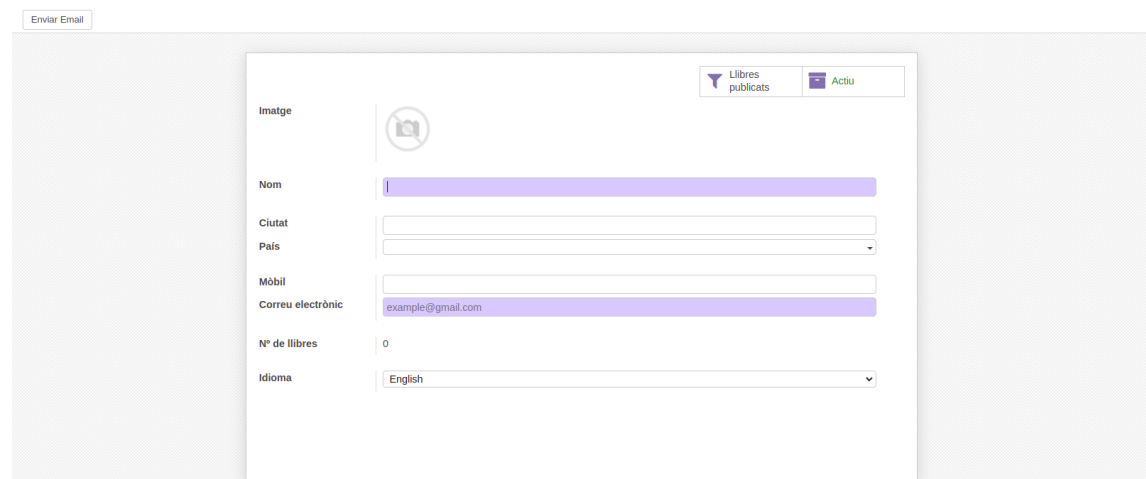
Correu electrònic

Nº de llibres 0

Idioma


Figura 14: Captura de pantalla del prototip d'interfície del autor a Odoo 11. Font: elaboració pròpia

○ Vista d'editorials:



Enviar Email

Libres publicats Actiu

Imatge 

Nom

Ciutat

País

Mòbil

Correu electrònic

Nº de llibres 0

Idioma

Figura 15: Captura de pantalla del prototip d'interfície de la editorial a Odoo 11. Font: elaboració pròpia

○ Vista d'ordinadors:

The screenshot shows a web interface for managing computers. At the top, there are tabs for 'Disponible' and 'Perdut'. Below that, there are buttons for 'No disponible', 'Disponible', 'Logat', and 'Perdut'. On the left, there are buttons for 'Crear Prèstec' and 'Crear Reserva'. On the right, there are buttons for 'No reservat' and 'Reservat'. The main form has a purple 'Actiu' button in the top right corner. The form fields are: 'Nom' (Model 2134), 'Codi' (0000.MO), 'Sucursal' (dropdown), 'Codi de la sucursal Origen' (dropdown), 'Polzades' (20"), 'RAM' (8GB), 'HDD' (512GB), 'SO' (dropdown), 'Data de sortida' (dropdown), 'Imatge de l'ordinador' (Carregar el teu arxiu), and 'Descripció' (text area).

Figura 16: Captura de pantalla del prototip d'interfície de l'ordinador a Odoo 11. Font: elaboració pròpia

○ Vista de categories:

The screenshot shows a web interface for managing categories. At the top right, there is a purple 'Actiu' button. The form has three fields: 'Categoria' (text input), 'Categoria principal' (dropdown), and 'Descripció' (text area).

Figura 17: Captura de pantalla del prototip d'interfície de les categories a Odoo 11. Font: elaboració pròpia

○ Vista de sucursals:

The screenshot shows a form for managing branches. At the top right, there are two buttons: 'Llibres' (with a purple icon) and 'Actiu' (with a purple icon). The form fields are as follows:

Nom	<input type="text"/>
Codi	<input type="text"/>
Carrer	<input type="text"/>
Mòbil	<input type="text"/>
Correu electrònic	<input type="text"/>
N° de llibres	0
Idioma	English

Figura 18: Captura de pantalla del prototip d'interfície de les sucursals a Odoo 11. Font: elaboració pròpia

○ Vista de membres:

The screenshot shows a form for managing members. At the top, there are several buttons: 'Perdonar càstig', 'Crear Prèstec', 'Enviar Email', 'Sense llibre prestat', 'Amb llibre prestat', 'Castigat', 'Perdonar càstig', 'Crear Reserva', 'Sense ordinador prestat', 'Amb ordinador prestat', and 'Castigat'. The form fields are as follows:

Imatge	<input type="text"/>
Nom	<input type="text"/>
TIN	56894387Q
Ciutat	<input type="text"/>
País	<input type="text"/>
Data de naixement	<input type="text"/>
Mòbil	<input type="text"/>
Correu electrònic	example@gmail.com
N° Prèstecs	0
Temps de préstec	30
N° de reserves	0
N° del membre	NS
Data de registre	15/06/2022
Idioma	English

Figura 19: Captura de pantalla del prototip d'interfície dels membres a Odoo 11. Font: elaboració pròpia

○ Vista de préstec:

Validar

Finalitzar Enviar per Email

Creat Validat Finalitzat

Actiu

Préstec OS

Producte

Membre

Nom del membre

Data de préstec 02/06/2022

Dies 0

Figura 20: Captura de pantalla del prototip d'interfície del préstec a Odoo 11. Font: elaboració pròpia

○ Vista de reserva:

Validar

Finalitzar Enviar per Email

Creat Validat Finalitzat Cancel·lat

Actiu

Reserva BS

Producte

Membre

Nom del membre

Data de reserva

Figura 21: Captura de pantalla del prototip d'interfície de les reserves a Odoo 11. Font: elaboració pròpia

- Menú principal:

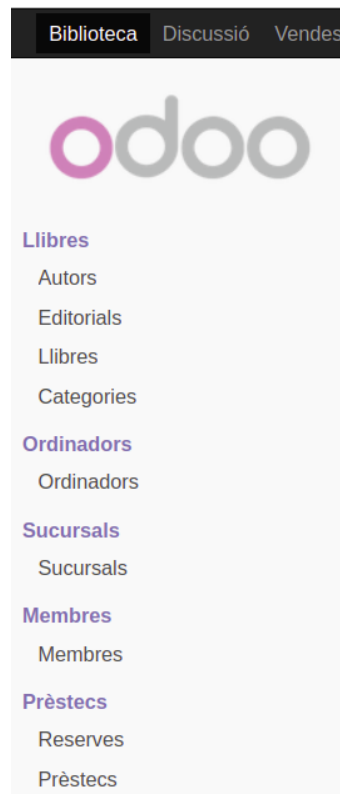


Figura 22: Captura de pantalla del prototip del menú del mòdul a Odoo 11. Font: elaboració pròpia

4.2 Capa de negoci

En aquesta capa parlem de la lògica que deu tindre el programari.

Parlarem per tant, de les funcions que deu tindre el sistema, com són activades per l'usuari i perquè són importants cadascuna d'elles. Dit d'altra forma, en aquesta capa indicarem la lògica que deu seguir el programari i en la que es basem per a crear el model de dades.

Tant els llibres i ordinadors, com els membres, com els préstecs i les reserves disposen d'estats. Els estats ens indicaran:

- Llibres i ordinadors: Disponibilitat per llogar i per a reservar.
- Membres: Si poden o no alquilar llibres i ordinadors (únicament 1 ordinador y un llibre a la vegada per persona, però fins a 3 reserves en total per persona)
- Préstec o reserva: Estat actual de la operació. Si ha estat validada, finalitzada o cancel·lada.

En el nostre cas, cal diferenciar entre dos tipus de funcions:

- **Funcions implícites**: No caldrà programar-les, ja que vendran de sèrie per heretar del mòdul "base":
 - Crear: En el moment l'usuari punxe el botó de "Crear" en un model de dades específic, el sistema ha de ser capaç d'obrir-li una vista formulari d'eixe model de dades (llibre, ordinador, categoria, membre, autor, editorial, sucursal, reserva o préstec) preparada per tal que l'usuari pugui introduir els valors corresponents del nou objecte en els camps. També han de distingir-se a la vista els camps que són necessaris dels que no, per tal que l'usuari sàpiga que deu introduir.
 - Editar: La mateixa lògica que la mencionada anteriorment, però aquesta vegada editant un formulari ja creat.
 - Guardar: En el moment l'usuari punxe el botó de "Guardar" en un formulari d'un model de dades específic (llibre, ordinador, categoria, membre, autor, editorial, sucursal, reserva o préstec) que ha creat, el sistema ha de ser capaç d'emmagatzemar-lo a la base de dades.
 - Relacionar: Hi haurà molts camps de models que seran una relació amb altres models. Per exemple, en el formulari de préstec, un parell de camps han de fer al·lusió als membres i als llibres. En el moment l'usuari vulga donar valor a algun d'aquests camps i punxe en el selector, el sistema ha de ser capaç de mostrar-li tots els valors possibles, que són totes les entrades de la BBDD del model al que fa referència.

- **Funcions explícites:** Aquestes funcions són les que haurem de programar en els fitxers de codi.
 - Informes: Alguns models han de gaudir d'informes en pdf per a imprimir. Alguns d'ells són molt importants, com en el cas del llibre (identificador únic per a pegar-lo a la portada) o en el cas del membre (carnet del membre). En la vista, l'usuari podrà descarregar-los, i quan punxe en el botó d'imprimir, es deuen descarregar aquests arxius pdf.
 - Lectura de dades: L'usuari deu ser capaç d'interactuar amb el sistema mitjançant teclat i lector de codi de barres. El sistema deu respondre igual a la entrada de dades.

També existeixen models amb estats, com s'ha pogut observar a la capa de presentació: Aquests son:

- Llibre:
 - Disponibilitat préstec: No disponible, disponible, llogat, perdut.
 - Disponibilitat reserva: No reservat, reservat.
- Ordinador:
 - Disponibilitat préstec: No disponible, disponible, llogat, perdut.
 - Disponibilitat reserva: No reservat, reservat.
- Membre:
 - Respecte llibres: Amb llibre llogat, sense llibre llogat, penalitzat.
 - Respecte ordinadors: Amb ordinador llogat, sense ordinador llogat, penalitzat.
- Préstec:
 - Creat, validat, finalitzat
- Reserva:
 - Creat, validat, finalitzat, cancel·lat

A continuació, les funcions de transicions d'estats:

- Transició a validat:
 - Préstec: En el moment l'usuari punxe el botó de validar en un préstec creat, el sistema ha de passar el préstec a estat "validat", i a més el llibre o ordinador ha de passar a estat "prestat" i el membre a estat "amb llibre llogat" o "amb ordinador llogat".
 - Reserva: En el moment l'usuari punxe el botó de validar en una reserva creada, el sistema ha de passar la reserva a estat "validat", i a més el llibre o ordinador ha de passar a estat "reservat" i al membre afegir-li una reserva (màxim 3 en total)".

- Transició a finalitzat:
 - Préstec: En el moment l'usuari punxe el botó de finalitzar en un préstec creat, el sistema ha de passar el préstec a estat "finalitzat", i a més el llibre o ordinador ha de passar a estat "disponible" i el membre a estat "sense llibre llogat" o "sense ordinador llogat", o "penalitzat" en funció de si ha tornat el producte a temps o no,.
 - Reserva: En el moment l'usuari punxe el botó de validar en una reserva creada, el sistema ha de passar la reserva a estat "finalitzat", i a més el llibre o ordinador ha de passar a estat "no reservat" i al membre llevar-li una reserva (màxim 3 en total)".
- Transició a cancel·lat:
 - Reserva: En el moment l'usuari punxe el botó de cancel·lar en una reserva creada, el sistema ha de passar la reserva a estat "cancel·lat", i a més el llibre o ordinador ha de passar a estat "no reservat" i al membre llevar-li una reserva (màxim 3 en total)".
- Marcar com a disponible: Quan en un producte (llibre o ordinador), l'usuari punxa el botó de disponible, el producte ha de passar a estat "disponible".
- Marcar com a prestat: Quan en un producte (llibre o ordinador), l'usuari punxa el botó de prestat, el producte ha de passar a estat "prestat".
- Marcar com a perdut: Quan en un producte (llibre o ordinador), l'usuari punxa el botó de perdut, el producte ha de passar a estat "perdut".
- Perdonar càstig: Quan l'usuari punxe el botó de perdonar càstig, el sistema ha de perdonar al membre la penalització i passar el membre a estat "sense llibre llogat" o "sense ordinador llogat".

4.3 Capa de dades

En aquesta capa, es parlarà de quina serà les estructura de la BBDD. La biblioteca disposarà d'una única base de dades, que serà compartida per tots els mòduls.

Com hem anomenat abans, crearem dos mòduls. El mòdul dels préstecs serà dependent del de gestió bibliotecària, i heretarà els models de dades per a modificar-los i afegir camps.

Per a entendre millor totes les taules que inclourà en la base de dades els dos mòduls que introduïrem, es mostra el següent diagrama de classes UML:

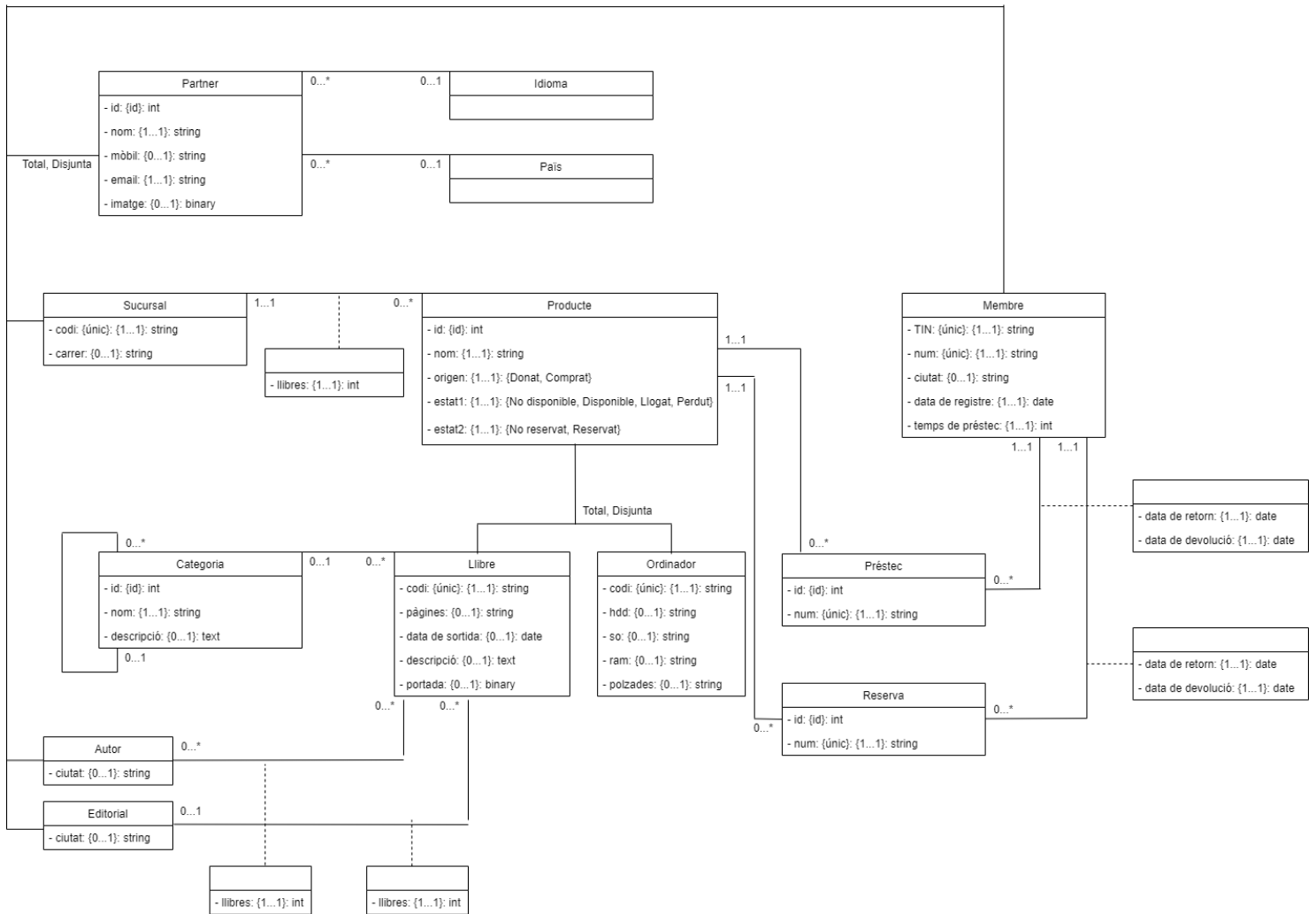


Figura 23: Captura de pantalla del diagrama de classes dels mòduls. Font: elaboració pròpia a partir de draw.io

Com podem observar, hi ha una gran quantitat de taules. Aquestes no seran les úniques taules de l'ERP, ja que estarà compost d'altres mòduls bàsics que tenen les seues taules, entre ells el del “producte” (que inclou les taules del producte), i “base”, que inclou taules bàsiques com: “res.partner”, “res.users” o “res.company”, entre altres.

5. Desenvolupament de la solució

Una volta dissenyada la solució, en aquest capítol es procedirà a desenvolupar-la. Aquesta fase del desenvolupament software és la més llarga. Per tal de comprendre millor com s'ha arribat a realitzar el treball, es parlarà primer de les ferramentes utilitzades.

5.1 Ferramentes utilitzades

Les ferramentes que s'han utilitzat al llarg del projecte es poden classificar en tres grans grups:

5.1.1 Python, PostgreSQL, Docker y Odoo11

Aquestes són les ferramentes lògiques en les que està basat el programari realitzat:

- Python: Python és un llenguatge de programació orientada a objectes. En 2022, és el més utilitzat en les empreses, i en el que està programat Odoo. Segons Delgado (2022), és un llenguatge interpretat, i que busca facilitar la llegibilitat del codi. Té una gran quantitat de llibreries de suport. Per a Odoo 11, utilitzarem la versió 2.7.



Figura 24: Logo de Python.

- PostgreSQL: PostgreSQL és un sistema gestor de bases de dades de codi obert, que suporta l'orientació a objectes. És el servidor que utilitza Odoo degut a que permet una alta concurrència, cosa essencial en un ERP per tal que diversos usuaris actualitzin dades a la vegada.



PostgreSQL

Figura 25: Logo de PostgreSQL.

- Docker: Docker és un sistema operatiu programat amb Go, de codi obert, que és base en el desplegament d'aplicacions a partir de contenidors. Aquests contenidors es poden iniciar i detindre. El que es busca és crear contenidors als quals es vincula una única base de dades amb els fitxers necessaris. Dit d'altra manera, empaquetar una aplicació en un contenidor. Això permet donar portabilitat a l'aplicació, permetent que s'execute en el núvol.



- Odoo 11: Finalment, importarem les llibreries i el codi base d'Odoo, en la seua versió 11. Aquest codi estarà realitzat amb Python, i gestionat en la base de dades amb PostgreSQL. Triem la versió 11 degut a què és una versió estable, correcta i òptima, on s'han automatitzat prou coses respecte a les versions anteriors amb llibreries.



Figura 27: Logo d'Odoo.

5.1.2 VSCode

Per a programar, s'ha utilitzat l'IDE VSCode. VSCode és un IDE que suporta una gran quantitat de llenguatges de programació per a tractar-los. És el que s'ha utilitzat degut a la gran quantitat d'extensions que conté, que permeten programar d'una forma més còmoda, a la vegada que conté suport per a Git.

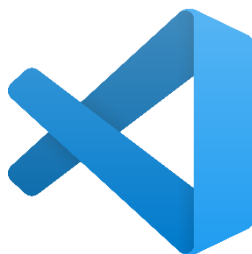


Figura 28: Logo de VSCode.

5.1.3 Github

I finalment, per a guardar els fitxers en un lloc segur, s'ha utilitzat Github. Github és una plataforma que permet la creació de repositoris en el núvol. Mitjançant la ferramenta Git, es poden clonar eixos repositoris en local, treballar amb ells, actualitzar-los, i tornar-los a pujar al núvol.



Figura 29: Logo de Github.

Pel que fa a Git, és una ferramenta software que, mitjançant diverses instruccions, porta un control de les actualitzacions d'un repositori. Permet fer punt de controls (commit) en un moment concret del repositori, i tornar a eixa versió sempre que es vulga. Útil, per exemple, per a recuperar una versió funcional del programari, ja que un canvi pot causar un resultat no desitjat.



Figura 30: Logo de Git.

5.2 Estructura dels fitxers programari

A continuació, una vegada configurades e instal·lades les ferramentes, s'organitzaran els repositoris en els que es trobaran els fitxers del programari.

Per una millor organització, es separaran els fitxers .py (de lògica) i .xml (de vistes) en carpetes diferents. A més, hi ha fitxers que s'encarreguen de crear informes en pdf, o de donar permisos a l'usuari, que també tenen les seues respectives carpetes.

Troblem un total de dos mòduls:

- Mòdul de gestió bàsica (“*library_management_jordi*”):

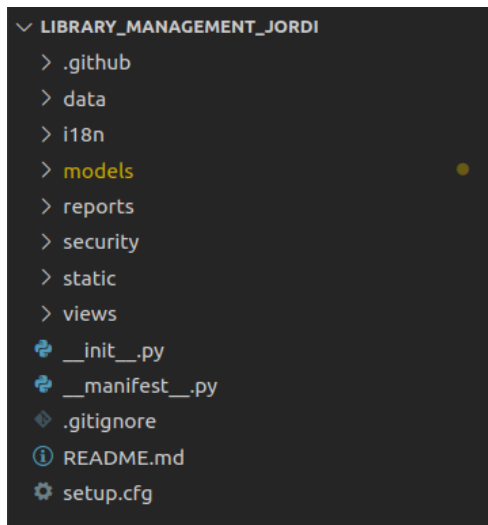


Figura 31: Captura de pantalla del mòdul library_management_jordi des de VSCode. Font: elaboració pròpia.

- Mòdul per a préstecs (“*library_loans_jordi*”):

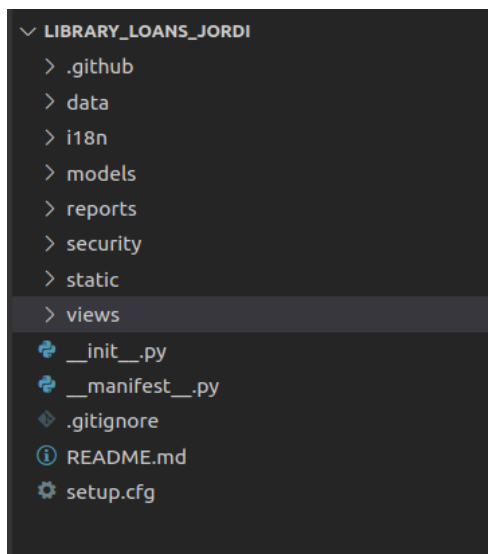


Figura 32: Captura de pantalla del mòdul library_loans_jordi des de VSCode. Font: elaboració pròpia.

5.2.1 Mòdul de gestió bàsica

El mòdul de gestió bàsica de la biblioteca inclou la creació d'entitats i productes. No permet crear préstecs o reserves. L'estructura de les carpetes és la següent:

- Data: Aquesta carpeta conté un fitxer de dades demo, que són objectes que introduïrem a la base de dades quan instal·lem el mòdul. També inclourem un fitxer anomenat “*sequence*”, que li dona automàticament un número de soci al membre quan el crea, i plantilles per als emails.

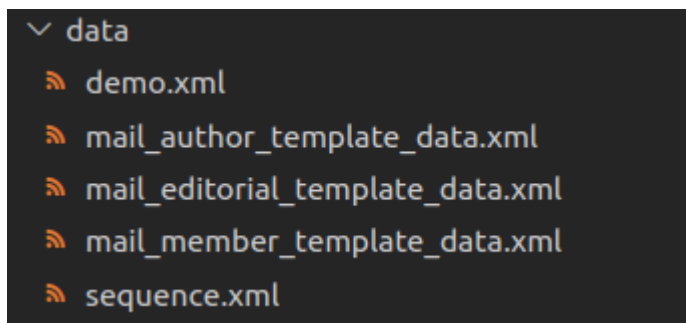


Figura 33: Captura de pantalla de la carpeta data del mòdul library_management_jordi des de VSCode. Font: elaboració pròpia.

- i18n: En aquesta carpeta és on es guarden els fitxers .po, que són els fitxers de traducció. Les traduccions disponibles estan en català i espanyol, tot i que l'idioma en el que està fet el mòdul és en anglés.



Figura 34: Captura de pantalla de la carpeta i18n del mòdul library_management_jordi des de VSCode. Font: elaboració pròpia.

- **Models:** Aquesta carpeta conté tota la lògica de la aplicació. Inclou els fitxers .py. Podem trobar els fitxers de tots els models de dades que volem crear, que corresponen a les diferents taules de la base de dades del mòdul. L'arxiu "base_archive", conté una funcionalitat de bloqueig, que permet bloquejar un objecte per a no utilitzar-lo (per exemple, un membre que es borra de la biblioteca). És per això que eixa funcionalitat fer-la en un fitxer apart i heretar-la en tots els models de dades.

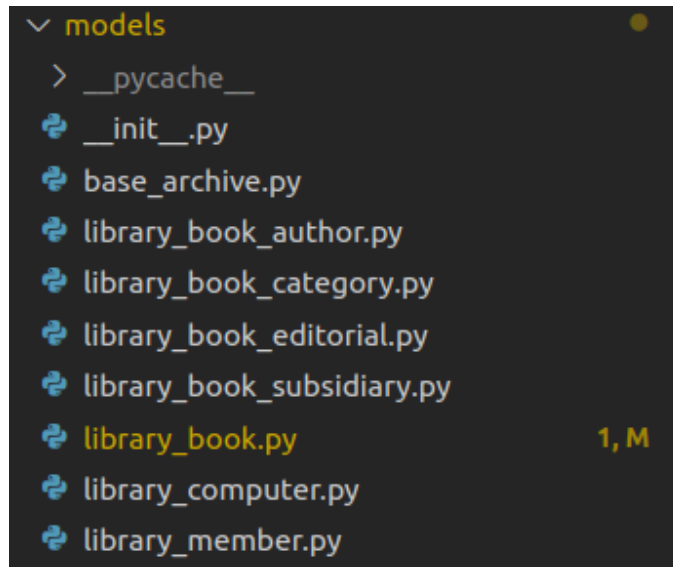


Figura 35: Captura de pantalla de la carpeta models del mòdul library_management_jordi des de VSCode. Font: elaboració pròpia.

- **Reports:** Ací trobem dos tipus de fitxer. Dintre de la carpeta views, trobem totes les plantilles dels informes, les quals seran cridades per a carregar-se en un fitxer anomenat reports.xml. En aquest fitxer, es creen tots els informes del mòdul, i en cada informe es crida a la seua respectiva plantilla.

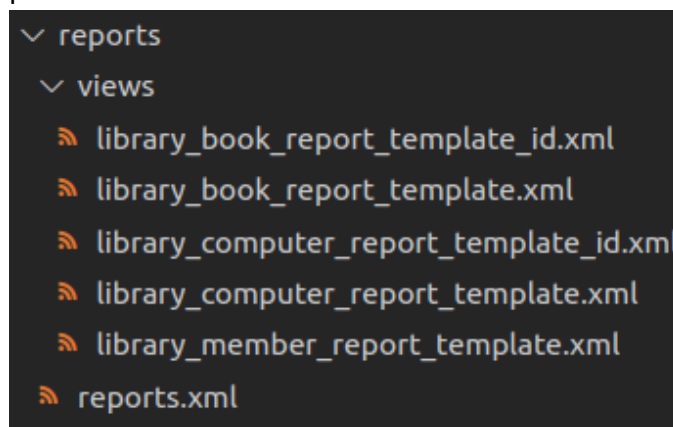


Figura 36: Captura de pantalla de la carpeta reports del mòdul library_management_jordi des de VSCode. Font: elaboració pròpia.

- Security: A la carpeta *security*, trobem dos fitxers. Un que crea els diferents grups d'usuaris del mòdul (usuari i administrador), i l'altre que dona a eixos usuaris els permisos de lectura, escriptura, duplicar i esborrar un objecte.

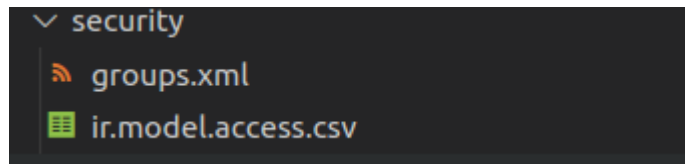


Figura 37: Captura de pantalla de la carpeta security del mòdul library_management_jordi des de VSCode. Font: elaboració pròpia.

- Static i views: I finalment, la carpeta de static conté el css i el js(web), en el cas d'aquest mòdul no n'hem requerit la imatge del mòdul. La carpeta views conté totes les vistes dels diferents models de dades.

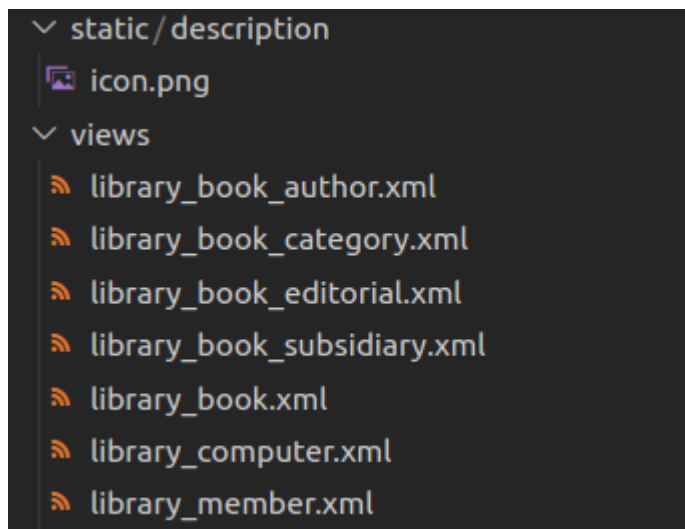


Figura 38: Captura de pantalla de les carpetes views i static del mòdul library_management_jordi des de VSCode. Font: elaboració pròpia.

Finalment, indicar que el fitxer `__init__.py` deu importar tots els fitxers.py. S'ha decidit que el fitxer importe únicament la carpeta de models, i dintre d'aquesta incloure un altre fitxer `__init__.py` que importe tots els models:

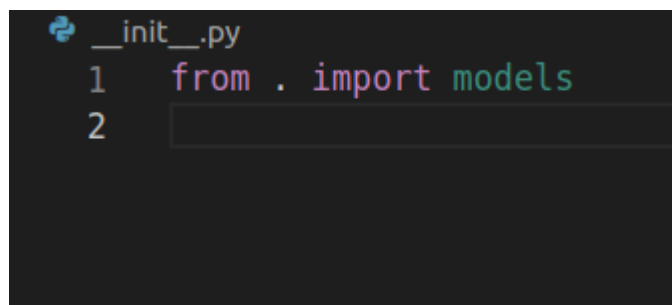


Figura 39: Captura de pantalla del contingut de `__init__.py`. Font: elaboració pròpia.


```
models >  __init__.py
1  from . import base_archive
2  from . import library_book
3  from . import library_book_category
4  from . import library_book_author
5  from . import library_book_editorial
6  from . import library_book_subsidiary
7  from . import library_member
8  from . import library_computer
9
```

Figura 40: Captura de pantalla del contingut de `__init__.py` de la carpeta `models` de `library_management_jordi`. Font: elaboració pròpia.

I el fitxer `__manifest__.py` inclou totes les dependències i les dades que té que carregar el mòdul, així com el nom i l'autor. Es deuen incloure totes les vistes que es desitgen carregar, així com les dades demo.


```
 __manifest__.py
1
2  {
3      "name": "Library",
4      "summary": "library_management_jordi",
5      "version": "11.0.1.0",
6      "category": "Library",
7      "author": "Jordi Toledo Coloma, from Aures TIC Consultors",
8      "installable": True,
9      "depends": [
10         "base",
11         "mail",
12         "product",
13     ],
14     "data": [
15         'data/sequence.xml',
16         'data/mail_author_template_data.xml',
17         'data/mail_editorial_template_data.xml',
18         'data/mail_member_template_data.xml',
19         'security/groups.xml',
20         'security/ir.model.access.csv',
21         'views/library_book.xml',
22         'views/library_computer.xml',
23         'views/library_book_author.xml',
24         'views/library_book_category.xml',
25         'views/library_book_editorial.xml',
26         'views/library_book_subsidiary.xml',
27         'views/library_member.xml',
28         "reports/reports.xml",
29         "reports/views/library_book_report_template.xml",
30         "reports/views/library_book_report_template_id.xml",
31         "reports/views/library_computer_report_template.xml",
32         "reports/views/library_computer_report_template_id.xml",
33         "reports/views/library_member_report_template.xml",
34     ],
35     "demo": [
36         'data/demo.xml',
37     ]
38 }
39
```

Figura 41: Captura de pantalla del contingut de `__manifest__.py`. Font: elaboració pròpia.

5.2.2 Mòdul per a préstecs

El mòdul de gestió de préstecs afegeix la funcionalitat a la aplicació per a poder fer préstecs i reserves. Crea aquests dos nous models de dades i afegeix funcionalitats a alguns vells. És un mòdul que depèn de l'anterior, i per tant hereta tota la seua funcionalitat. L'estructura de les carpetes és la següent:

- Data: Aquesta carpeta conté noves plantilles per a nous emails creats. Té un nou fitxer de seqüència creat, aquesta vegada per a assignar un número als préstecs i les reserves. Observem també un fitxer anomenat *ir_cron*. Un cron és una acció que s'executa automàticament, cada període de temps. En el nostre cas, tenim tres crons, un que recorda als membres de la finalització dels préstecs amb un correu (la plantilla del qual és *loan_reminder_template_data*), un altre que perdona el càstig als membres i un últim que cancel·la reserves quan ha passat el temps per a que la reclame el membre. Per a que funcionen correctament, els crons es vinculen a unes funcions, creades en un fitxer .py (*library_loan*).

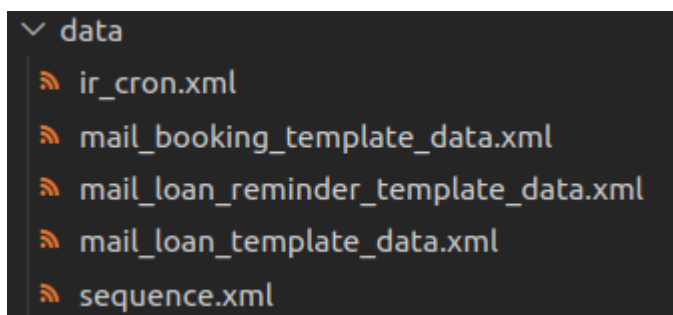


Figura 42: Captura de pantalla de la carpeta data del mòdul library_loans_jordi des de VSCode. Font: elaboració pròpia.

- i18n: En aquesta carpeta guardem els fitxers de les traduccions. S'han creat nous camps, missatges i funcions, i s'han de traduir.



Figura 43: Captura de pantalla de la carpeta i18n del mòdul library_loans_jordi des de VSCode. Font: elaboració pròpia.

- Models: En aquesta carpeta trobem els fitxers .py corresponents a tots els models de dades nous, així com alguns ja existents en `library_management_jordi`, als que se li ha afegit funcionalitats.

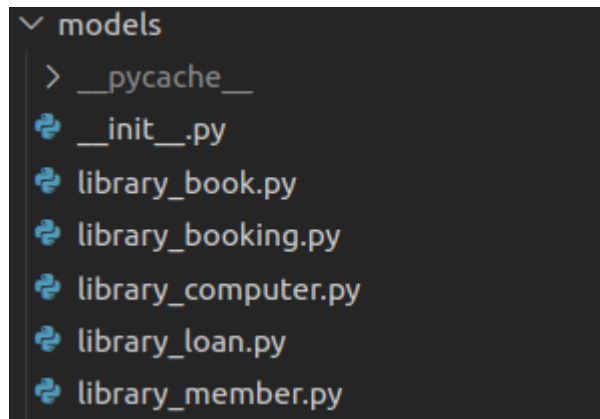


Figura 44: Captura de pantalla de la carpeta models del mòdul `library_loans_jordi` des de VSCode. Font: elaboració pròpia.

- Reports: En aquesta carpeta, seguim el mateix procediment que en l'anterior. Creem un fitxer `reports`, on introduïrem els nous informes, i dintre de `views`, les plantilles dels nous informes.

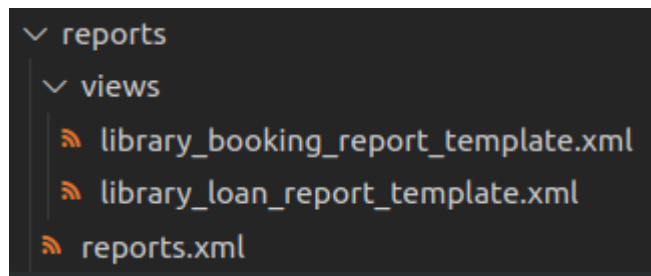


Figura 45: Captura de pantalla de la carpeta reports del mòdul `library_loans_jordi` des de VSCode. Font: elaboració pròpia.

- Security: S'utilitzaran els mateixos grups que els creats a `library_management`, de manera que únicament se li donarà permisos a aquests usuaris en els préstecs i les reserves:

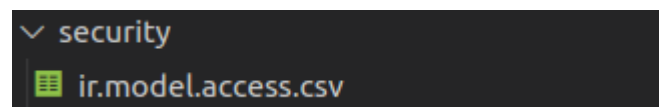


Figura 46: Captura de pantalla de la carpeta security del mòdul `library_loans_jordi` des de VSCode. Font: elaboració pròpia.

- Static i views: Ací, de nou, introduïm en la carpeta de static la imatge del mòdul, i a la carpeta de views les vistes dels models de dades que hem creat o editat.

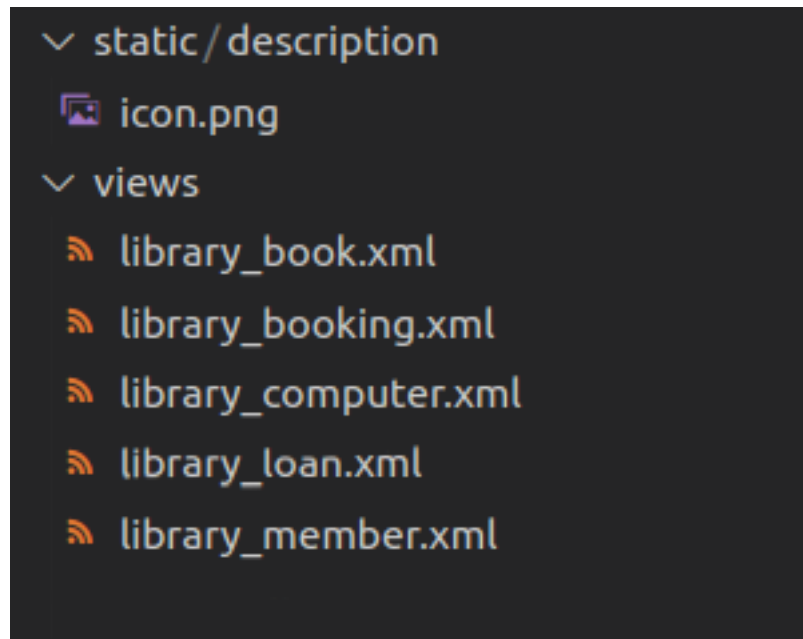


Figura 47: Captura de pantalla de les carpetes views i static del mòdul library_loans_jordi des de VSCode. Font: elaboració pròpia.

I finalment, l'estructura seguida per als fitxers `__init__.py` és la mateixa.

```
models > __init__.py
1  from . import library_loan
2  from . import library_booking
3  from . import library_member
4  from . import library_book
5  from . import library_computer
6
```

Figura 48: Captura de pantalla del contingut de `__init__.py` de la carpeta models de library_loans_jordi. Font: elaboració pròpia.

I el `__manifest__.py` aquesta vegada no conté dades de demo, però sí que introduïm la dependència de `library_management`.

```
__manifest__.py
1  {
2      "name": "Loans",
3      "summary": "library_loans_jordi",
4      "version": "11.0.1.0",
5      "category": "Loans",
6      "author": "Jordi Toledo Coloma, from Aures TIC Consultors",
7      "depends": [
8          "base",
9          "library_management_jordi",
10     ],
11     "data": [
12         "data/sequence.xml",
13         "security/ir.model.access.csv",
14         "reports/views/library_loan_report_template.xml",
15         "reports/views/library_booking_report_template.xml",
16         "reports/reports.xml",
17         "data/mail_loan_template_data.xml",
18         "data/mail_booking_template_data.xml",
19         "data/ir_cron.xml",
20         "data/mail_loan_reminder_template_data.xml",
21         'views/library_book.xml',
22         'views/library_computer.xml',
23         'views/library_member.xml',
24         'views/library_loan.xml',
25         'views/library_booking.xml',
26         'views/library_loan_statistics.xml',
27     ],
28     "demo": [
29     ]
30 }
31
```

Figura 49: Captura de pantalla del contingut de `__manifest__.py`. Font: elaboració pròpia.

5.3 Contingut i funcionalitat

Ara que ja coneixem tant la manera en la que desenvolupar el projecte com les eines que utilitzem i l'estructura que seguiran els fitxers del projecte, anem a desenvolupar la solució.

Hem mostrat els fitxers que componen els mòduls, però no el contingut. Per tal de veure el contingut dels fitxers, amb totes les funcions de lògica i vistes, recercar als annexos adjuntats amb l'entrega.

En aquest punt parlarem únicament dels models de dades i les seues funcionalitats, i no dels fitxers de permisos o dels models genèrics com `base_archive`. Partirem del principi que els dos mòduls estan instal·lats a la base de dades creada en Odoo per tal de mostrar la funcionalitat.

El procediment que es seguirà per a explicar cada model serà:

- Mostrar les vistes del conjunt d'objectes.

- Explicar els camps.
- Explicar els botons i les funcions relacionades a cada botó.
- Altres funcions (si en té).
- Informes (si en té)

També remarcar que tots els models disposen de traçabilitat, tenint el mateix funcionament en totes les vistes formulari. Ho explicarem una única vegada en la vista del llibre, per a simplificar.



Figura 50: Captura de pantalla de la traçabilitat del llibre One Piece 1. Originària d'una base de dades personal. Font: elaboració pròpia.

Com podem veure, ens indica que l'usuari Administrator ha creat el llibre, i li ha donat valor als camps:

Si hi ha alguna modificació a aquests camps, ho indicarà en l'historial de missatges (baix de la vista formulari), i a més es dirà quin usuari ho ha fet. La traçabilitat es pot posar per als camps que desitgen, en el cas del llibre els estats, i en els altres models, altres camps, però amb el mateix funcionament.

Finalment ja explicar que els camps amb fons morat són camps obligatoris, que deuen tindre un valor per a que l'objecte es dese a la base de dades.

5.3.1 Llibres

Hereten del producte.

5.3.1.1 Vistes des del menú

Al model de llibres disposem de dos formes diferents de mostrar els objectes: kanban i llista.

Kanban (per defecte):

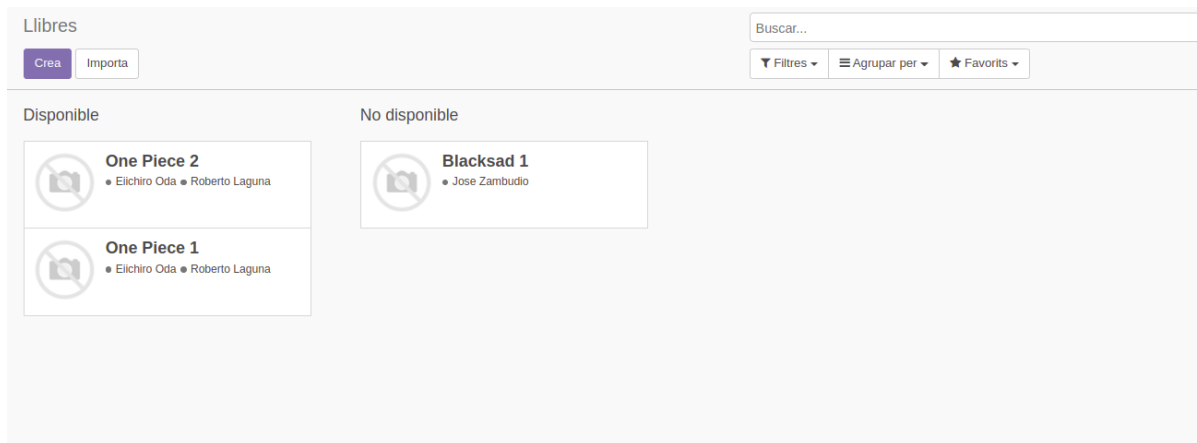


Figura 51: Captura de pantalla de la vista kanban dels llibres. Originària d'una base de dades personal. Font: elaboració pròpia.

Llista:

The screenshot shows a List view for books. The interface includes a table with the following data:

<input type="checkbox"/>	Nom	Data de sortida	N° Pàgines	Editorial	Categoria
<input type="checkbox"/>	One Piece 2	29/07/2004	200	Ivrea	Llengua anglesa / Aventura
<input type="checkbox"/>	One Piece 1	12/05/2004	200	Ivrea	Llengua anglesa / Aventura
<input type="checkbox"/>	Blacksad 1	17/02/2000	50	Ivrea	Llengua anglesa

The top navigation bar includes 'Crea', 'Importa', a search bar, and filters.

Figura 52: Captura de pantalla de la vista llista dels llibres. Originària d'una base de dades personal. Font: elaboració pròpia.

5.3.1.2 Camps

Llibre: One Piece 1

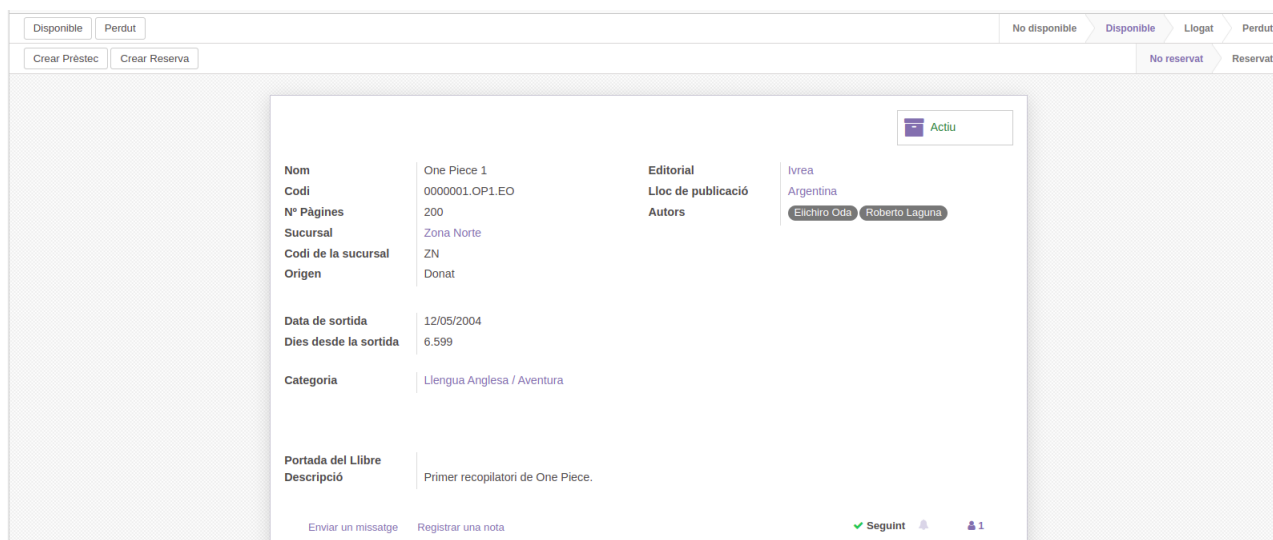


Figura 53: Captura de pantalla d'un llibre del projecte a Odo11. Originària d'una base de dades personal. Font: elaboració pròpia.

Camps:

- **Estat respecte préstec:** Aquest camp és un selector, i ens indica l'estat en què es troba el llibre respecte a disponibilitat en la biblioteca. Per defecte es troba en *No disponible*. Potser: *No disponible, disponible, llogat o perdut*.
- **Estat respecte reserva:** Un altre selector, indica la disponibilitat del llibre per a reservar. Per defecte és *No reservat*. Potser: *No reservat, reservat*.
- **Nom:** Es tracta del nom del llibre.
- **Codi:** És el codi que tindrà el llibre. Aprofita per a imprimir-lo i pegarlo al revers del propi llibre, per saber on ubicar-lo. Cada llibre registrat a la biblioteca gaudeix d'un codi únic. Serà el camp a partir del qual es cree el codi de barres del llibre, per a poder usar un lector de codi de barres i trobar immediatament el llibre a la base de dades.

Exemple de cerca per codi de barres:

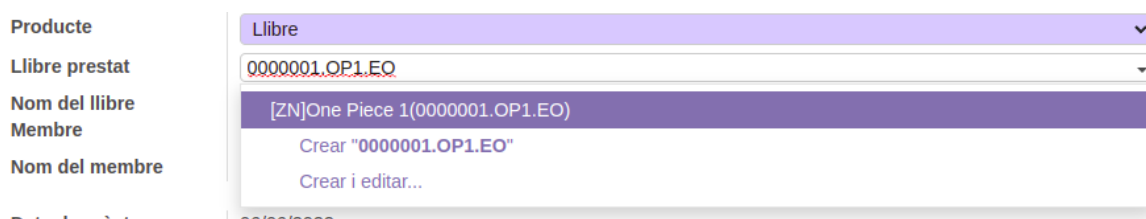


Figura 54: Captura de pantalla d'una cerca del llibre One Piece 1. L'escaneig amb el lector de codi de barres ha reduït la recerca a un objecte, el que buscàvem. Font: elaboració pròpia.

- **Pàgines:** Número de pàgines que té el llibre.
- **Sucursal:** És un camp que relaciona el llibre amb el model de sucursal. Podem seleccionar un objecte sucursal que disposem a la base de dades. Indica la sucursal a la que pertany el llibre, ja que l'ERP integra totes les sucursals de la empresa.
- **Codi de la sucursal:** El codi de la sucursal. S'introdueix automàticament al seleccionar una sucursal.
- **Origen:** L'origen del llibre. Potser donat o comprat.
- **Editorial:** És un camp que relaciona el llibre amb el model d'editorial. Podem seleccionar un objecte editorial que disposem a la base de dades. Indica l'editorial a la que pertany el llibre.
- **Lloc de publicació:** El país en el que es troba ubicada l'editorial. S'introdueix automàticament al seleccionar una editorial.
- **Autors:** És un camp que relaciona el llibre amb el model d'autor. Podem seleccionar diversos objectes autor que disposem a la base de dades. Indica els autors que han escrit el llibre.
- **Data de sortida:** Data en la que surt el llibre.
- **Dies des de la sortida:** Camp que conta els dies des que va sortir el llibre. Es calcula automàticament al posar la data.
- **Categoria:** És un camp que relaciona el llibre amb el model de categoria. Podem seleccionar un objecte categoria que disposem a la base de dades. Indica la categoria a la que pertany el llibre.
- **Portada del llibre:** És la portada del llibre. S'introdueix una imatge.
- **Descripció:** Breu descripció del llibre. És un camp de text.

Camps ocults:

- **Nom complet:** Camp ocult que es calcula automàticament. És una cadena que segueix el format [CodiSucursal]Nom{Codi}. És la cadena de caràcters que identificarà al llibre alhora de buscar-lo. Per exemple, quan en un altre model de dades relacionem un llibre, aquest apareixerà amb aquest nom i no amb el nom simple.
- **Membre:** Camp ocult que es mostra quan el llibre es troba llogat. És el membre amb qui està llogat el llibre (el membre es mostra a través del DNI, no el nom).

- **Data de retorn:** Camp ocult que es mostra quan el llibre es troba llogat. Mostra la data en la que el llibre serà retornat.

Exemple:

The screenshot shows a web interface for a library system. At the top, there are tabs for 'Disponible', 'Perdut', 'No disponible', 'Disponible', and 'Llogat' (the last one is circled in red). Below these are buttons for 'Crear Préstec', 'Crear Reserva', 'No reservat', and 'Reservat'. The main content area displays the details of a book loan. The book is 'One Piece 1' with code '000001.OP1.EO'. The loan is made by member '37513488B' on '06/07/2022'. The return date is '06/07/2022', which is highlighted with a red box and a red arrow. The book is categorized as 'Llengua anglesa / Aventura' and is the first volume of the 'One Piece' series. There are buttons for 'Actiu', 'Seguint', and '1' at the bottom.

Figura 55: Captura de pantalla dels camps al llibre One Piece 1 llogat. Font: elaboració pròpia.

5.3.1.3 Botons i funcions

Botons:

- **Actiu:** Permet arxivar un llibre per tal que no siga utilitzat. Es pot desarxivar després.
- **Disponible:** Permet passar l'estat d'un llibre a *Disponible*.
- **Perdut:** Permet passar l'estat d'un llibre a *Perdut*.
- **Crear préstec:** Ens transporta a la vista formulari de préstecs, amb el llibre ja seleccionat. Indicada a la figura 88.
- **Crear reserva:** Ens transporta a la vista formulari de reserves, amb el llibre ja seleccionat. Indicada a la figura 95.

5.3.1.4 Altres funcionalitats

- **Passar el llibre a llogat:** Quan es valida un préstec relacionat amb el llibre, el llibre automàticament passa a estat *Llogat*, impeding que aquest pugui tornar a ser llogat i apareixent els camps corresponents.
- **Passar el llibre a disponible:** Quan es finalitza el préstec, si el membre ha retornat el llibre, el llibre passa a estat *Disponible* i els camps corresponents són amagats.
- **Passar el llibre a perdut:** Quan es finalitza el préstec, si el membre no ha retornat el llibre, el llibre passa a estat *Perdut* i els camps corresponents són amagats.
- **Passar el llibre a reservat:** Quan es valida una reserva relacionada amb aquest objecte, el llibre passa a estat *Reservat*.

- **Passar el llibre a no reservat:** Quan es finalitza o cancel·la una reserva relacionada amb aquest objecte, el llibre passa a estat *No reservat*.

5.3.1.5 Informes

Els principals informes en pdf que es poden obtindre d'aquest model són:

- **Informe del llibre:**

My Company

One Piece 1:

Autor: Eiichiro Oda, Roberto Laguna	Editorial: Ivrea
Pàgines: 200	Categoria: Llengua anglesa / Aventura
Data de llançament: 2004-05-12	

Correu electrònic: info@yourcompany.com Web: <http://www.yourcompany.com>
Pàgina: 1 / 1

Figura 56: PDF del llibre One Piece 1 inserit al document. Font: elaboració pròpia.

- **Identificador del llibre:** Serà apegat al revers del seu corresponent llibre.

2022-06-06 09:12

My
Company

1 / 1

000001.OP1.EO



Figura 57: PDF de l'identificador del llibre One Piece 1 inserit al document. Font: elaboració pròpia.

5.3.2 Autors

Hereten del partner.

5.3.2.1 Vistes des del menú

Al model d'autors disposem de dos formes diferents de mostrar els objectes: kanban i llista.

Kanban (per defecte):

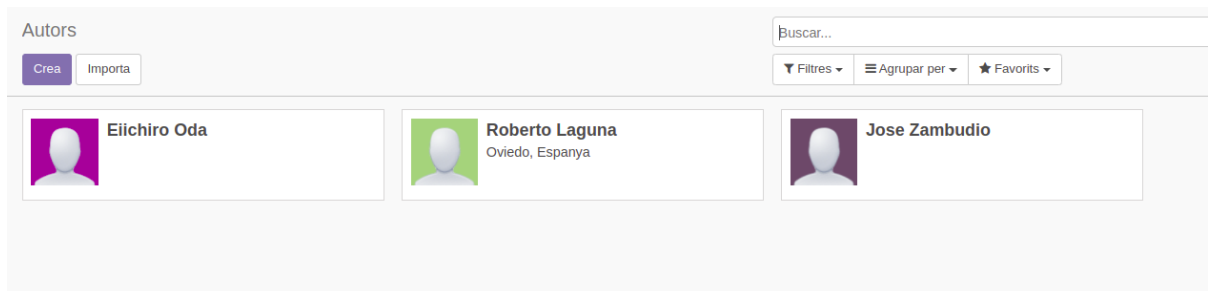


Figura 58: Captura de pantalla de la vista kanban dels autors. Originària d'una base de dades personal. Font: elaboració pròpia.

Llista:

The screenshot shows the 'Autors' interface in List view. It features the same top navigation as the Kanban view. The main content is a table with the following data:

<input type="checkbox"/>	Nom	Ciutat	País	Nº de llibres
<input type="checkbox"/>	Eiichiro Oda			2
<input type="checkbox"/>	Roberto Laguna	Oviedo	Espanya	2
<input type="checkbox"/>	Jose Zambudio			1

Figura 59: Captura de pantalla de la vista llista dels autors. Originària d'una base de dades personal. Font: elaboració pròpia.

5.3.2.2 Camps

Autor: Roberto Laguna

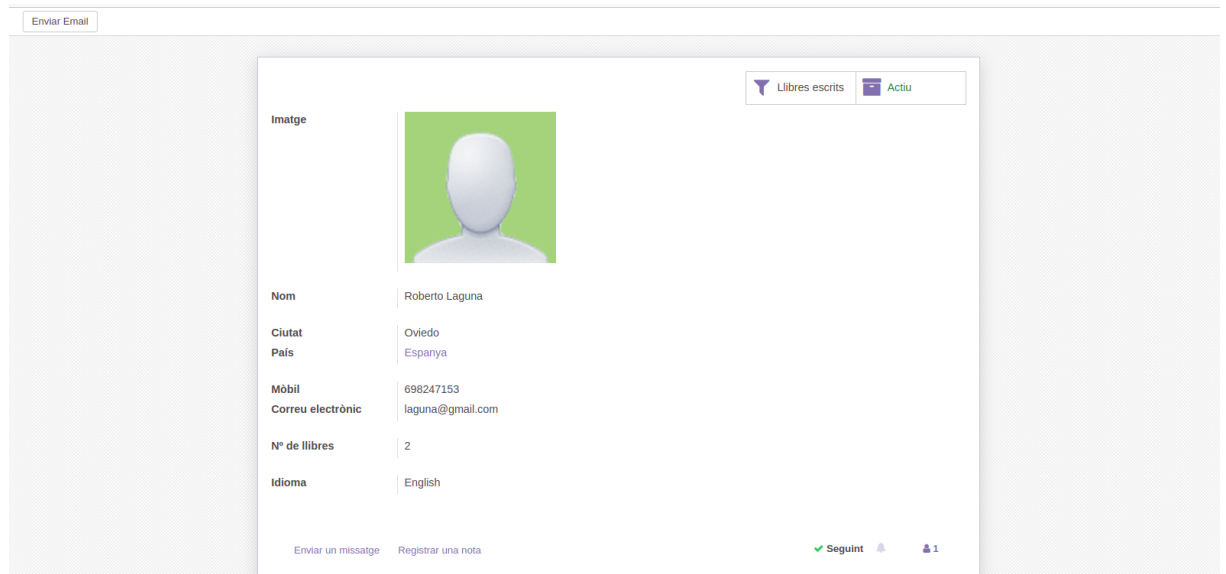


Figura 60: Captura de pantalla d'un autor del projecte Odoo11. Originària d'una base de dades personal.

Camps:

- **Imatge:** Aquest camp és la imatge d'avatar en la que s'identificarà l'autor. Es recomana que no siga de més de 300 x 300 píxels.
- **Nom:** Es tracta del nom de l'autor.
- **Ciutat:** És la ciutat en la que resideix l'autor.
- **País:** És un camp que relaciona l'autor amb el model de res.country. Per defecte disposem de tots els països com a objectes a la base de dades, és la taula res.country, que prové del mòdul *base*.
- **Mòbil:** Introduïm el mòbil de l'escriptor, per a contactar amb ell.
- **Correu electrònic:** Camp on introduïm l'email, per a poder enviar correus a l'autor.
- **Número de llibres:** Aquest camp es calcula automàticament. Mostra el número de llibres que ha escrit l'autor.
- **Idioma:** És un camp que relaciona l'autor amb el model de idioma. Per defecte, a la base de dades, únicament tenim l'anglès instal·lat, però podem instal·lar els idiomes que vulgam, estan els principals idiomes del món. L'idioma indica que tot correu que s'envie a eixe autor, el rebrà en l'idioma que tinga seleccionat (en aquest cas anglès), encara que l'usuari que ho fa ho veja en un idioma diferent (en aquest cas català).

5.3.2.3 Botons i funcions

Botons:

- **Actiu:** Permet arxivar un autor per tal que no siga utilitzat. Es pot desarxivar després.
- **Enviar correu:** Ens permet enviar un correu a l'autor. Existeix una plantilla per on iniciar, amb el correu de l'autor com a destinatari:

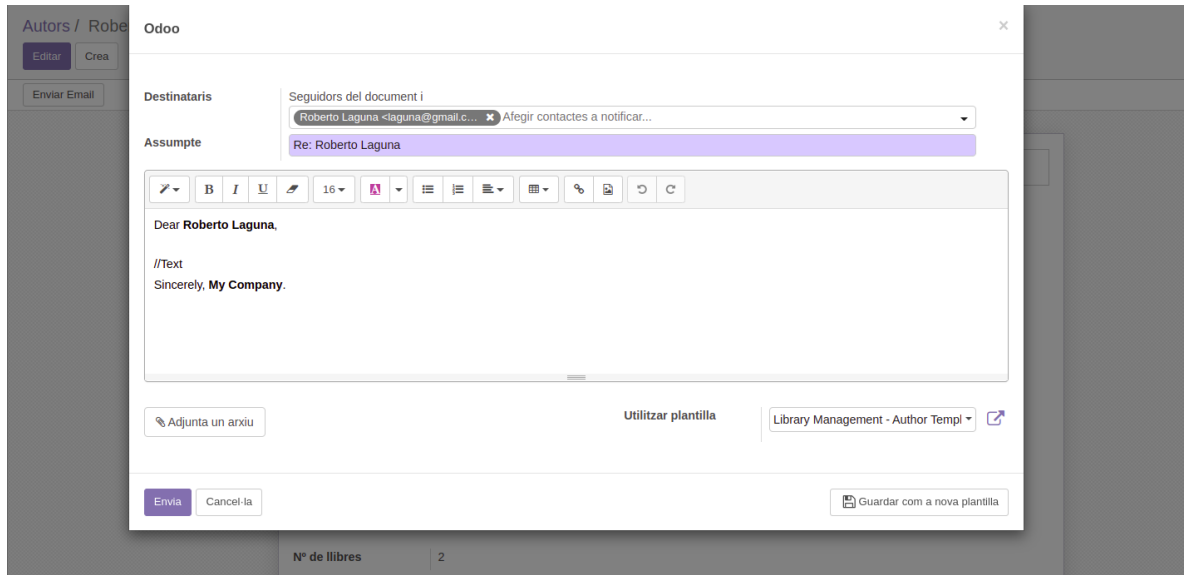


Figura 61: Captura de pantalla d'un correu base per a enviar a Roberto Laguna. Font: elaboració pròpia.

- **Llibres escrits:** Ens transporta a una vista de llista dels llibres, filtrada per els llibres que ha escrit aquest autor, i els llibres agrupats per la seua categoria.

Exemple:

	Nom	Data de sortida	Nº Pàgines	Editorial	Categoria
▼ Llengua anglesa / Aventura (2)			400		
	<input type="checkbox"/> One Piece 2	29/07/2004	200	Ivrea	Llengua anglesa / Aventura
	<input type="checkbox"/> One Piece 1	12/05/2004	200	Ivrea	Llengua anglesa / Aventura

Figura 62: Captura de pantalla dels llibres escrits per Roberto Laguna. Originària d'una base de dades personal. Font: elaboració pròpia.

5.3.2.4 Altres funcionalitats

- **Afegir llibres escrits:** Cada vegada que des del formulari del llibre, es relaciona el llibre amb un autor, aquest llibre es vincularà a l'autor, i per tant, apareixerà quan fem clic en el botó "Llibres escrits". També causarà que augmente la quantitat de llibres (Nº de llibres).

5.3.3 Editorials

Hereten del partner.

5.3.3.1 Vistes des del menú

Al model d'editorials disposem de dos formes diferents de mostrar els objectes: kanban i llista.

Kanban (per defecte):

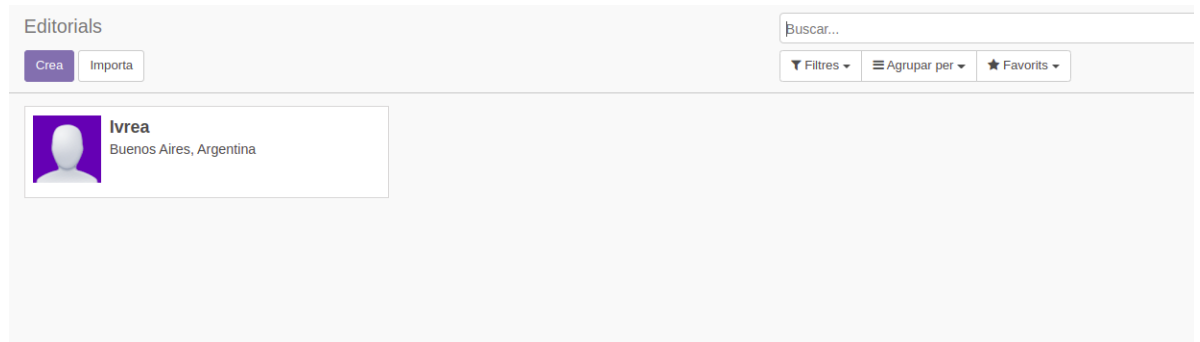


Figura 63: Captura de pantalla de la vista kanban de les editorials. Originària d'una base de dades personal. Font: elaboració pròpia.

Llista:

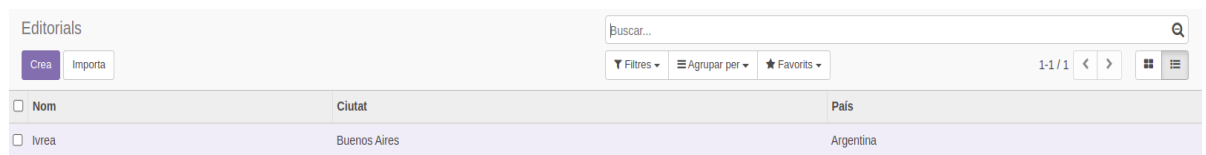


Figura 64: Captura de pantalla de la vista llista de les editorials. Originària d'una base de dades personal. Font: elaboració pròpia.

5.3.3.2 Camps

Editorial: Ivrea

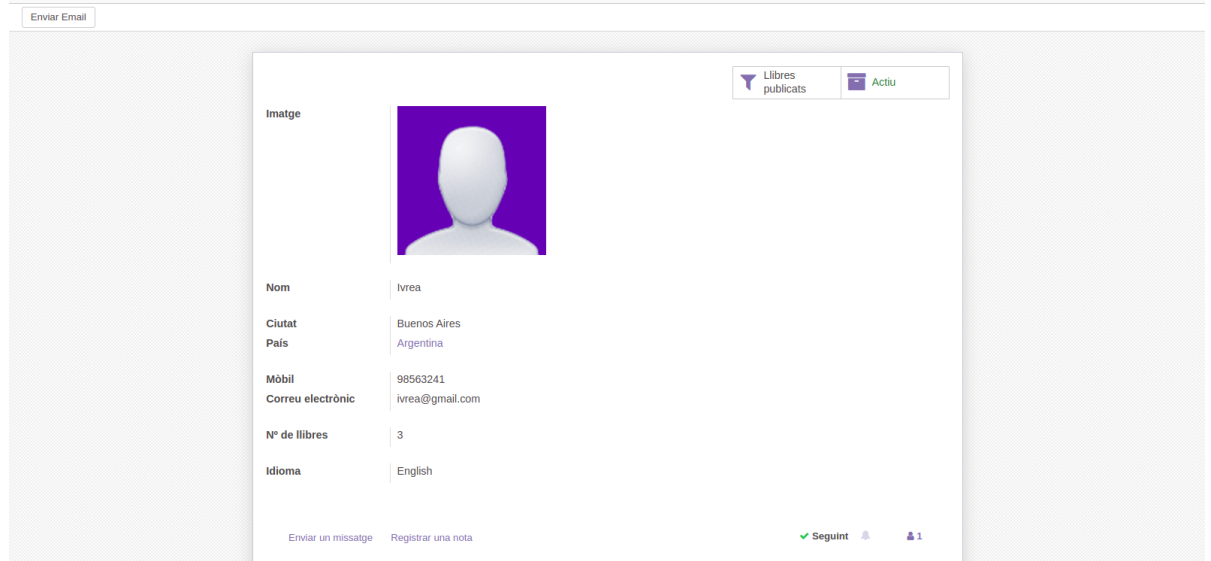


Figura 65: Captura de pantalla d'una editorial del projecte Odoo11. Originària d'una base de dades personal. Font: elaboració pròpia.

Camps:

- **Imatge:** Aquest camp és la imatge d'avatar en la que s'identificarà l'editorial. Es recomana que no siga de més de 300 x 300 píxels.
- **Nom:** Es tracta del nom de l'editorial.
- **Ciutat:** És la ciutat en la es funda l'editorial.
- **País:** És un camp que relaciona l'editorial amb el model de res.country, com hem anomenat anteriorment al model d l'autor.
- **Mòbil:** Introduïm el mòbil de l'editorial, per a contactar amb la companyia.
- **Correu electrònic:** Camp on introduïm l'email, per a poder enviar correus a l'editorial.
- **Número de llibres:** Aquest camp es calcula automàticament. Mostra el número de llibres que ha publicat l'editorial.
- **Idioma:** El camp que relaciona l'editorial amb el model d'idioma. Podrem elegir entre els idiomes instal·lats, i el seleccionat serà en el que ens comunicarem amb l'entitat, independentment de l'idioma que tinga instal·lat l'usuari. Funciona d'igual manera que en el model d'autors.

5.3.3.3 Botons i funcions

Botons:

- **Actiu:** Permet arxivar una editorial per tal que no siga utilitzada. Es pot desarxivar després.
- **Enviar correu:** Ens permet enviar un correu a l'editorial. Existeix una plantilla per on iniciar, amb el correu de l'editorial com a destinatari:

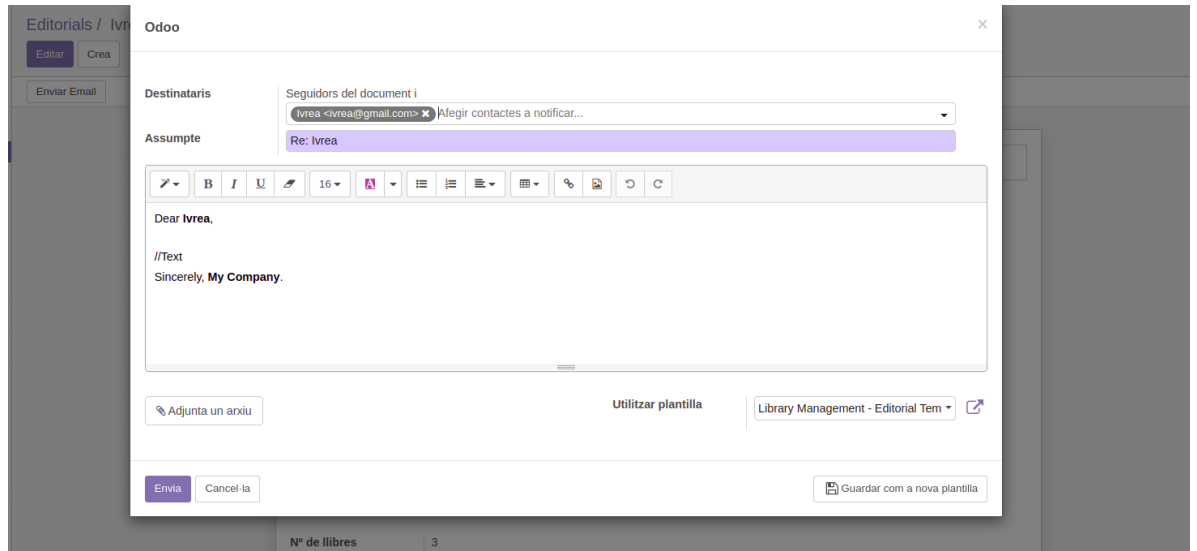


Figura 66: Captura de pantalla d'un correu base per a enviar a lvrea. Font: elaboració pròpia.

- **Llibres publicats:** Ens transporta a una vista de llista dels llibres, filtrada per els llibres que ha publicat aquesta editorial, i els llibres agrupats per la seua categoria.

Exemple:

	<input type="checkbox"/> Nom	Data de sortida	Nº Pàgines	Editorial	Categoria
▼ Llengua Anglesa (1)			50		
	<input type="checkbox"/> Blacksad 1	17/02/2000	50	Ivrea	Llengua Anglesa
▼ Llengua Anglesa / Aventura (2)			400		
	<input type="checkbox"/> One Piece 2	29/07/2004	200	Ivrea	Llengua Anglesa / Aventura
	<input type="checkbox"/> One Piece 1	12/05/2004	200	Ivrea	Llengua Anglesa / Aventura

Figura 67: Captura de pantalla dels llibres publicats per Ivrea. Originària d'una base de dades personal. Font: elaboració pròpia.

5.3.3.4 Altres funcionalitats

- **Afegir llibres publicats:** Cada vegada que des del formulari del llibre, es relaciona el llibre amb una editorial, aquest llibre es vincularà a l'editorial, i per tant, apareixerà quan fem clic en el botó "Llibres publicats". També causarà que augmente la quantitat de llibres (Nº de llibres).

5.3.4 Categories

5.3.4.1 Vistes des del menú

Al model de categories disposem d'una única manera de mostrar els objectes: llista.

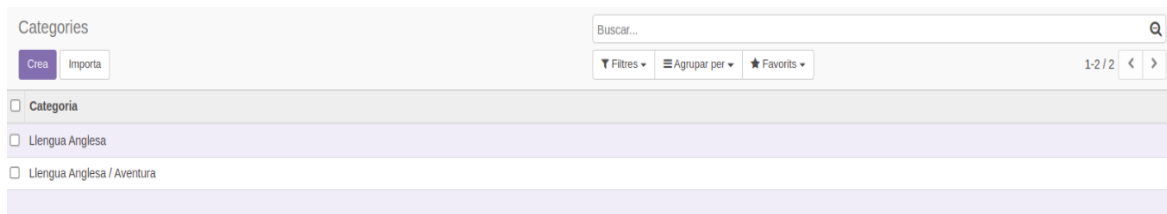


Figura 68: Captura de pantalla de la vista llista de les categories. Originària d'una base de dades personal. Font: elaboració pròpia.

5.3.4.2 Camps

Categoria: Aventura



Figura 69: Captura de pantalla d'una categoria del projecte Odoo11. Originària d'una base de dades personal. Font: elaboració pròpia.

Camps:

- **Categoria:** Es tracta del nom de la categoria.
- **Categoria principal:** Es tracta del pare de la categoria. La lògica de les categories és base en la relació pare fill. Es tracta d'una relació amb el mateix model de categoria, on apunta a un altre objecte categoria. En aquest cas, per exemple, Aventura serà una subcategoria de Llengua anglesa.
- **Descripció:** Una breu descripció de la categoria. És un camp de text.

Camps ocults:

- **Nom complet:** Camp ocult que es calcula automàticament. És una cadena que segueix el format "CategoriaPrincipal / Categoria". És la cadena de caràcters que identificarà la categoria alhora de buscar-la. Per exemple, quan en un altre model de dades relacionem una categoria, aquest apareixerà amb aquest nom i no amb el nom simple (apareixerà com a Llengua anglesa / Aventura, i no com Aventura).

5.3.4.3 Botons i funcions

Botons:

- **Actiu:** Permet arxivar una categoria per tal que no siga utilitzada. Es pot desarxivar després.

5.3.4.4 Altres funcionalitats

- **Relació amb llibre:** Cada vegada que des del formulari del llibre, es relaciona el llibre amb una categoria, aquest llibre es vincularà a la categoria. La categoria vinculada serà, en aquest cas, "Aventura", però al formulari del llibre es mostrarà amb el nom complet, "Llengua Anglesa / Aventura".
- **Relació amb categoria:** Cada vegada que des del formulari de categoria, es relaciona la categoria amb una altra categoria, aquest categoria en la que es trobem es vincularà amb l'altra, sent la seua filla.

5.3.5 Ordinadors

Hereten del producte.

5.3.5.1 Vistes des del menú

Al model d'ordinadors disposem de dos formes diferents de mostrar els objectes: kanban i llista.

Kanban (per defecte):

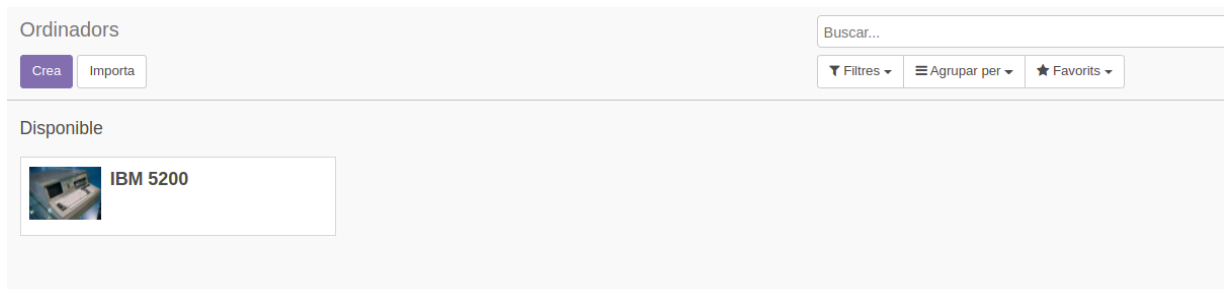
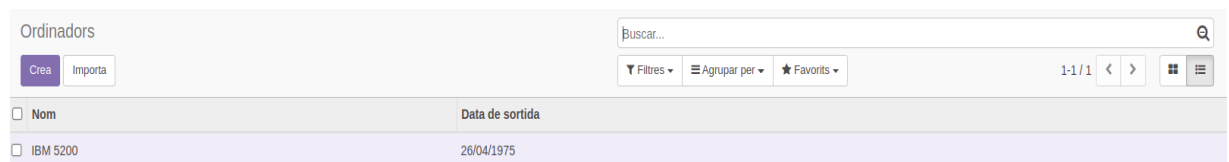


Figura 70: Captura de pantalla de la vista kanban dels ordinadors. Originària d'una base de dades personal. Font: elaboració pròpia.

Llista:



The screenshot shows the 'Ordinadors' (Computers) interface in List view. It features a search bar 'Buscar...' and buttons for 'Crea' and 'Importa'. On the right, there are dropdown menus for 'Filtres', 'Agrupar per', and 'Favorits', along with pagination controls showing '1-1 / 1' and navigation arrows. The table below has two columns: 'Nom' (Name) and 'Data de sortida' (Output date).

<input type="checkbox"/>	Nom	Data de sortida
<input type="checkbox"/>	IBM 5200	26/04/1975

Figura 71: Captura de pantalla de la vista llista dels ordinadors. Originària d'una base de dades personal. Font: elaboració pròpia.

5.3.5.2 Camps

Ordinador: IBM 5200

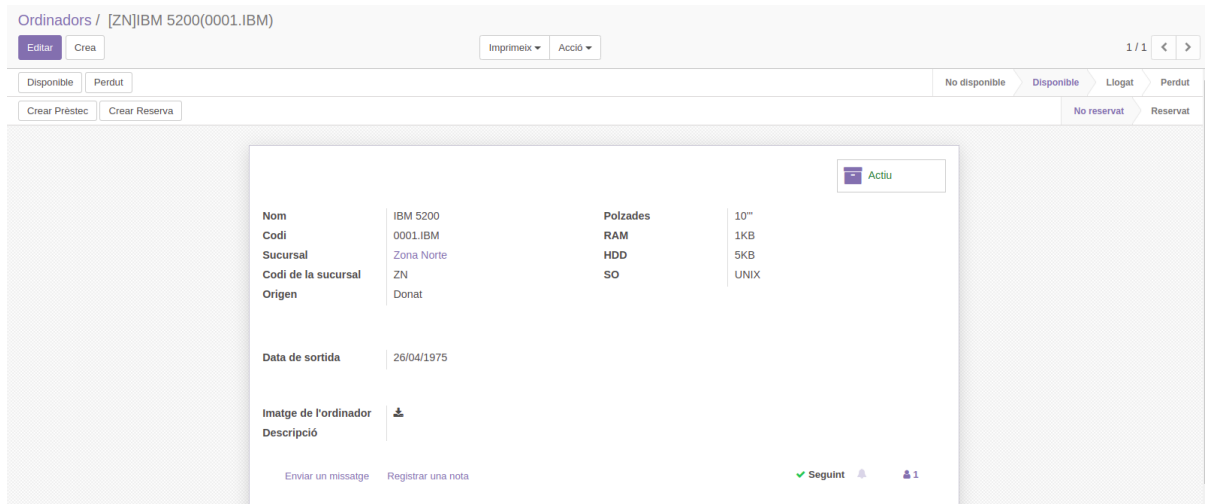


Figura 72: Captura de pantalla d'un ordinador del projecte a Odoo11. Originària d'una base de dades personal. Font: elaboració pròpia.

Camps:

- **Estat respecte préstec:** Aquest camp és un selector, i ens indica l'estat en què es troba l'ordinador respecte a disponibilitat en la biblioteca. Per defecte es troba en No disponible. Potser: No disponible, disponible, llogat o perdut.
- **Estat respecte reserva:** Un altre selector, indica la disponibilitat de l'ordinador per a reservar. Per defecte és No reservat. Potser: No reservat, reservat.
- **Nom:** Es tracta del nom de l'ordinador.
- **Codi:** És el codi que tindrà el l'ordinador. Aprofita per a imprimir-lo i pegar-lo al revers de la computadora, per identificar-la i diferenciar-la. Cada ordinador registrat a la biblioteca gaudeix d'un codi únic. Serà el camp a partir del qual es cree el codi de barres de l'ordinador, amb la mateixa funcionalitat que el codi de barres del llibre (ja que els dos són productes)

Exemple de cerca per codi de barres:

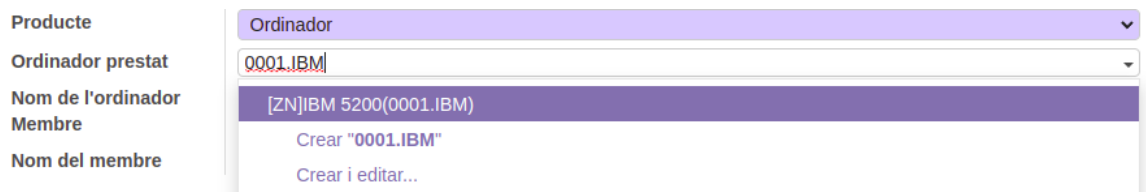


Figura 73: Captura de pantalla d'una cerca de l'ordinador. L'escaneig amb el lector de codi de barres ha reduït la cerca a un objecte, el que buscàvem. Font: elaboració pròpia.

- **Sucursal:** És un camp que relaciona l'ordinador amb el model de sucursal. Podem seleccionar un objecte sucursal que disposem a la base de dades. Indica la sucursal a la que pertany l'ordinador, amb el mateix funcionament que amb el llibre.
- **Codi de la sucursal:** El codi de la sucursal. S'introdueix automàticament al seleccionar una sucursal.
- **Origen:** L'origen de l'ordinador. Potser donat o comprat.
- **Polzades:** És un camp informatiu que indica les polzades que té l'ordinador.
- **RAM:** És un camp que indica la memòria RAM de la que disposa l'ordinador.
- **HDD:** Camp informatiu que indica la quantitat de memòria no volàtil de la que disposa l'ordinador.
- **SO:** Indica el sistema operatiu que està instal·lat a l'ordinador.
- **Data de sortida:** Data en la que surt el l'ordinador.
- **Portada de l'ordinador:** És la portada de l'ordinador. S'introdueix una imatge.
- **Descripció:** Breu descripció de l'ordinador. És un camp de text.

Camps ocults:

- **Nom complet:** Camp ocult que es calcula automàticament. És una cadena que segueix el format [CodiSucursal]Nom{Codi}. És la cadena de caràcters que identificarà a l'ordinador alhora de buscar-lo. Per exemple, quan en un altre model de dades relacionem un ordinador, aquest apareixerà amb aquest nom i no amb el nom simple.
- **Membre:** Camp ocult que es mostra quan l'ordinador es troba llogat. És el membre amb qui està llogat l'ordinador (el membre es mostra a través del DNI, no el nom).

- **Data de retorn:** Camp ocult que es mostra quan l'ordinador es troba llogat. Mostra la data en la que l'ordinador serà retornat.

Exemple:

Ordinadors / [ZN]IBM 5200(0001.IBM)

Editar Crea Imprimeix Acció 1 / 1 < >

Disponibile Perdut No disponible Disponible **Llogat** Perdut

Crear Préstec Crear Reserva No reservat Reservat

Nom	IBM 5200	Polzades	10"
Codi	0001.IBM	RAM	1KB
Sucursal	Zona Norte	HDD	5KB
Codi de la sucursal	ZN	SO	UNIX
Origen	Donat		
Membre	89463750J		
Data de retorn	21/06/2022		
Data de sortida	26/04/1975		
Imatge de l'ordinador			
Descripció			

Actiu

Enviar un missatge Registrar una nota Seguint 1

Figura 74: Captura de pantalla dels camps a l'ordinador IBM 5200 llogat. Font: elaboració pròpia.

5.3.5.3 Botons i funcions

Botons:

- **Actiu:** Permet arxivar un ordinador per tal que no siga utilitzat. Es pot desarxivar després.
- **Disponible:** Permet passar l'estat d'un ordinador a Disponible.
- **Perdut:** Permet passar l'estat d'un ordinador a Perdut.
- **Crear préstec:** Ens transporta a la vista formulari de préstecs, amb l'ordinador ja seleccionat. S'observa a la figura 88.
- **Crear reserva:** Ens transporta a la vista formulari de reserves, amb l'ordinador ja seleccionat. S'observa a la figura 95.

5.3.5.4 Altres funcionalitats

- **Passar l'ordinador a llogat:** Quan es valida un préstec relacionat amb el l'ordinador, aquest automàticament passa a estat Llogat, impedit que aquest pugui tornar a ser llogat i apareixent els camps corresponents.
- **Passar l'ordinador a disponible:** Quan es finalitza el préstec, si el membre ha retornat l'ordinador, aquest passa a estat Disponible i els camps corresponents són amagats.
- **Passar l'ordinador a perdut:** Quan es finalitza el préstec, si el membre no ha retornat l'ordinador, aquest passa a estat Perdut i els camps corresponents són amagats.
- **Passar l'ordinador a reservat:** Quan es valida una reserva relacionada amb aquest objecte, l'ordinador passa a estat Reservat.

- **Passar l'ordinador a no reservat:** Quan es finalitza o cancel·la una reserva relacionada amb aquest objecte, l'ordinador passa a estat No reservat.

5.3.5.5 Informes

Els principals informes en pdf que es poden obtindre d'aquest model són:

- **Informe del llibre:**

My Company

IBM 5200:

RAM: 1KB	HDD: 5KB
Polzades: 10"	SO: UNIX
Data de llançament: 1975-04-26	



Correu electrònic: info@yourcompany.com Web: http://www.yourcompany.com
Pàgina: 1 / 1

Figura 75: PDF de l'ordinador IBM 5200 inserit al document. Font: elaboració pròpia.

- **Identificador del llibre:** Serà apegat a l'ordinador físic al qual identifica.

2022-06-06 17:52

My
Company

1 / 1

0001.IBM



0001.IBM

Figura 76: PDF de l'identificador l'ordinador IBM 5200 inserit al document. Font: elaboració pròpia.

5.3.6 Sucursals

Hereten del partner.

5.3.6.1 Vistes des del menú

Al model de sucursals disposem de dos formes diferents de mostrar els objectes: kanban i llista.

Kanban (per defecte):

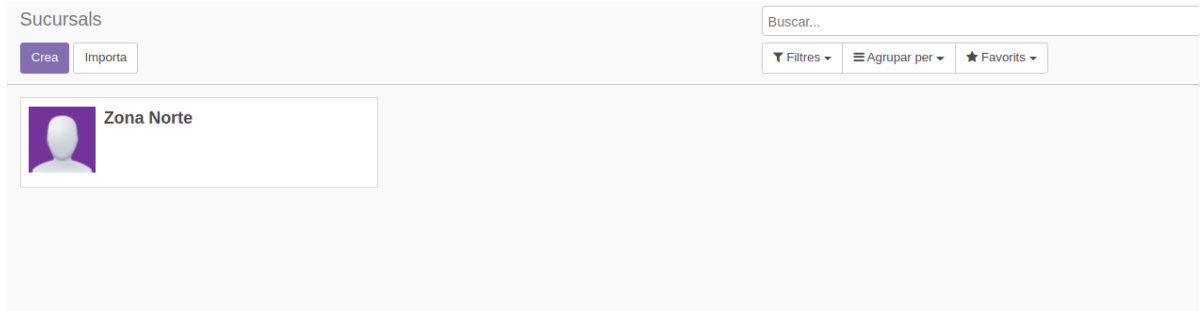


Figura 77: Captura de pantalla de la vista kanban de les sucursals. Originària d'una base de dades personal. Font: elaboració pròpia.

Llista:

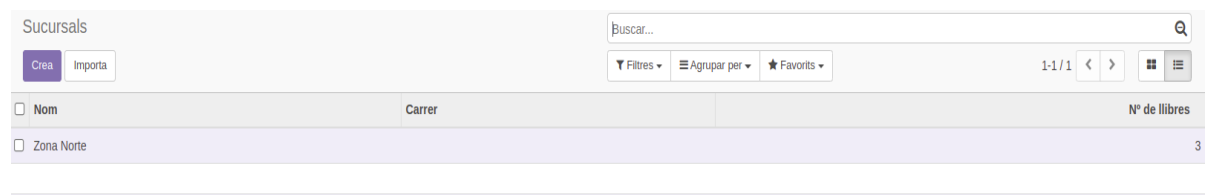


Figura 78: Captura de pantalla de la vista llista de les sucursals. Originària d'una base de dades personal. Font: elaboració pròpia.

5.3.6.2 Camps

Sucursal: Zona Norte

		Libres	Actiu
Nom	Zona Norte		
Codi	ZN		
Carrer			
Mòbil			
Correu electrònic			
Nº de llibres	3		
Idioma	English		

Figura 79: Captura de pantalla d'una sucursal del projecte Odoo11. Originària d'una base de dades personal. Font: elaboració pròpia.

Camps:

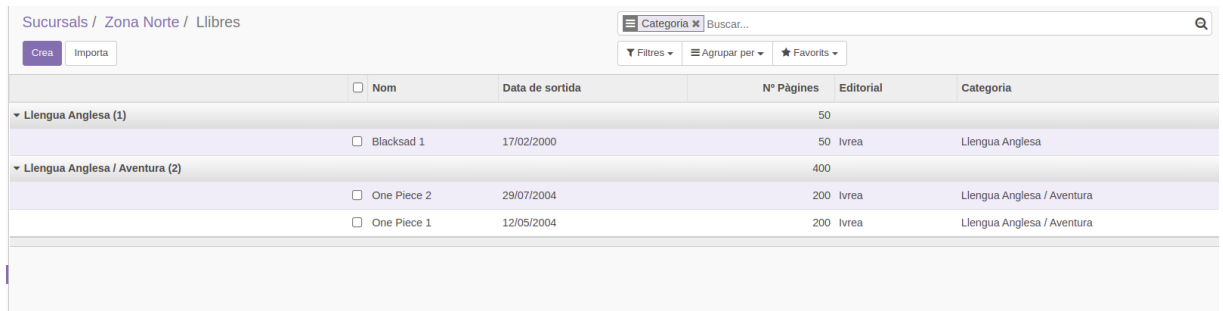
- **Nom:** Es tracta del nom de la sucursal.
- **Codi:** És el codi de la sucursal. Serà utilitzat en els noms complets dels productes.
- **Carrer:** Direcció on es troba la sucursal.
- **Mòbil:** Introduïm el mòbil de la sucursal, per a contactar amb ella.
- **Correu electrònic:** Camp on introduïm l'email, per a poder enviar correus a la sucursal.
- **Número de llibres:** Aquest camp es calcula automàticament. Mostra el número de llibres que té la sucursal.
- **Idioma:** El camp que relaciona la sucursal amb el model d'idioma. El funcionament és el mateix que als models d'autor i editorial, l'idioma amb què la seleccionem, serà l'idioma amb què enviarem els correus i es comunicarà el sistema amb la entitat.

5.3.6.3 Botons i funcions

Botons:

- **Actiu:** Permet arxivar una sucursal per tal que no siga utilitzada. Es pot desarxivar després.
- **Afegir llibres:** Ens transporta a una vista de llista dels llibres, filtrada per els llibres dels que disposa aquesta sucursal, i els llibres agrupats per la seua categoria.

Exemple:



<input type="checkbox"/>	Nom	Data de sortida	N° Pàgines	Editorial	Categoria
▼ Llengua anglesa (1)			50		
<input type="checkbox"/>	Blacksad 1	17/02/2000	50	Ivrea	Llengua anglesa
▼ Llengua anglesa / Aventura (2)			400		
<input type="checkbox"/>	One Piece 2	29/07/2004	200	Ivrea	Llengua anglesa / Aventura
<input type="checkbox"/>	One Piece 1	12/05/2004	200	Ivrea	Llengua anglesa / Aventura

Figura 80: Captura de pantalla dels llibres disponibles a la “Zona Norte”. Originària d’una base de dades personal. Font: elaboració pròpia.

5.3.6.4 Altres funcionalitats

- **Afegir llibres a la sucursal:** Cada vegada que des del formulari del llibre, es relaciona el llibre amb una sucursal, aquest llibre es vincularà a la sucursal, i per tant, apareixerà quan fem clic en el botó “Llibres”. També causarà que augmente la quantitat de llibres (Nº de llibres).
- **Afegir ordinadors a la sucursal:** Cada vegada que des del formulari de l’ordinador, aquest es relaciona amb una sucursal, quedarà vinculat a aquesta sucursal.

5.3.7 Membres

Hereten del partner.

5.3.7.1 Vistes des del menú

Al model de membres disposem de dos formes diferents de mostrar els objectes: kanban i llista.

Kanban (per defecte):

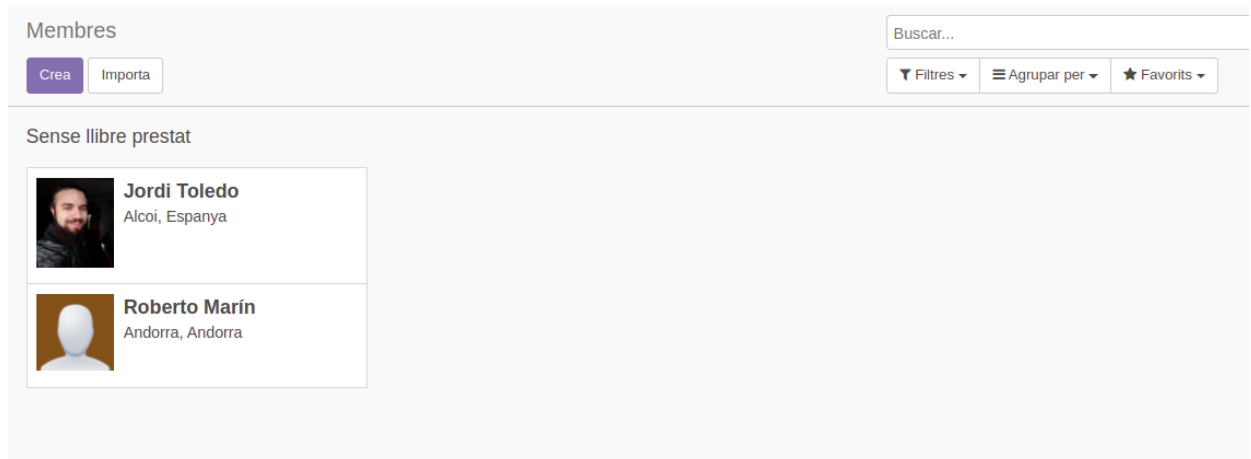


Figura 81: Captura de pantalla de la vista kanban dels membres. Originària d'una base de dades personal. Font: elaboració pròpia.

Llista:

The screenshot shows the 'Membres' interface in List view. At the top, there is a search bar labeled 'Buscar...' and three buttons: 'Crea', 'Importa', and 'Filtres'. Below the search bar, there are three more buttons: 'Filtres', 'Agrupar per', and 'Favorits'. The main content area displays a table with the following data:

<input type="checkbox"/>	Nom	Nº del membre	Ciutat	País	Data de registre
<input type="checkbox"/>	Jordi Toledo	NS0000007	Alcoi	Espanya	06/06/2022
<input type="checkbox"/>	Roberto Marín	NS0000008	Andorra	Andorra	06/06/2022

Figura 82: Captura de pantalla de la vista llista dels membres. Originària d'una base de dades personal. Font: elaboració pròpia.

5.3.7.2 Camps

Membre: Jordi Toledo

Perdonar càstig Crear Prèstec Enviar Email Sense llibre prestat Amb llibre prestat Castigat

Perdonar càstig Crear Reserva Sense ordinador prestat Amb ordinador prestat Castigat

Actiu

imatge

Nom Jordi Toledo
TIN 37513488B

Ciutat Alcoi
País Espanya

Data de naixement 16/06/1989
Mòbil 98563241
Correu electrònic jorditoledo@gmail.com

Nº Prèstecs 5
Temps de préstec 30
Nº de reserves 0

Nº del membre NS0000007
Data de registre 06/06/2022

Idioma English

Enviar un missatge Registrar una nota Seguint 1

Figura 83: Captura de pantalla d'un membre del projecte Odoo11. Originària d'una base de dades personal. Font: elaboració pròpia.

Camps:

- **Estat respecte llibre:** Aquest camp és un selector, i ens indica si el membre disposa o no d'un llibre llogat. Per defecte es troba en *Sense llibre llogat*. Potser: *Sense llibre llogat, Amb llibre llogat, o castigat*.
- **Estat respecte ordinador:** Aquest camp és un selector, i ens indica si el membre disposa o no d'un ordinador llogat. Per defecte es troba en *Sense ordinador llogat*. Potser: *Sense ordinador llogat, Amb ordinador llogat, o castigat*.
- **Imatge:** Aquest camp és la imatge d'avatar en la que s'identificarà el membre. Com les altres, millor que siga com a molt de 300 x 300 píxels.
- **Nom:** Es tracta del nom del membre.

- **TIN:** És tracta del codi del membre. És el camp equivalent al DNI, que l'identificarà. Cada TIN serà únic, i serà la cadena de caràcters que identificarà al llibre alhora de buscar el membre. Per exemple, quan en un altre model de dades relacionem un membre, aquest apareixerà amb el TIN i no amb el nom simple. A més, es tracta del camp a partir del qual es crea el codi de barres del membre, per a poder usar un lector de codi de barres i trobar immediatament el membre a la base de dades.

Exemple de cerca per codi de barres:

Membre	37513488B
Nom del membre	37513488B
Data de préstec	Crear i editar...

Figura 84: Captura de pantalla d'una cerca del membre Jordi Toledo. L'escaneig amb el lector de codi de barres ha reduït la cerca a un objecte, el que buscàvem. Font: elaboració pròpia.

- **Ciutat:** Ciutat on que resideix el membre.
- **País:** És un camp que relaciona el membre amb el model de res.country. El mateix funcionament que als altres models que hereten del partner, podem seleccionar tots els països del món, ja que tenim la taula al mòdul *base*.
- **Data de naixement:** Data a la qual va nàixer el membre.
- **Mòbil:** Introduïm el mòbil del membre, per a contactar amb ell.
- **Correu electrònic:** Camp on introduïm l'email, per a poder enviar correus al membre.
- **Número de préstecs:** Aquest camp es calcula automàticament. Mostra el número de préstecs que ha realitzat el membre en total.
- **Temps de préstec:** Un camp que indica quins són els dies que pot tindre el membre un producte prestat. Per exemple, per a aquest membre és 30, així que els seus préstecs duraran 30 dies.
- **Número de reserves:** Aquest camp es calcula automàticament. Mostra el número de reserves actives de les que disposa el membre. Com a màxim en pot tindre 3.
- **Número de membre:** Aquest camp es calcula automàticament, com hem indicat al punt 5.2.1. És el número que se li dona al membre. És únic per a cada membre i segueix un ordre (d'admissió). Per exemple, aquest membre és el 7, la qual cosa implica que és la setena persona en registrar-se com a membre.
- **Data de registre:** Aquest camp es calcula automàticament. Mostra la data a la qual el membre es va registrar al sistema.

- **Idioma:** És un camp que relaciona el membre amb el model de idioma. Té el mateix funcionament que tots els que hereten del partner, sent el camp que relaciona el membre amb una llengua. Aquesta llengua, serà la llengua amb la qual el sistema es comunicarà amb ell.

Camps ocults:

- **Data de càstig respecte llibre:** És la data en la que un membre entra en estat *Castigat* respecte al llibre. Serà utilitzada en una acció automàtica per a comprovar els dies que porta el membre amb la sanció.
- **Data de càstig respecte ordinador:** És la data en la que un membre entra en estat *Castigat* respecte a l'ordinador. Serà utilitzada en una acció automàtica per a comprovar els dies que porta el membre amb la sanció.

5.3.7.3 Botons i funcions

Botons:

- **Actiu:** Permet arxivar un membre per tal que no siga utilitzat. Es pot desarxivar després.
- **Enviar correu:** Ens permet enviar un correu al membre. Existeix una plantilla per on iniciar, amb el correu del membre com a destinatari:

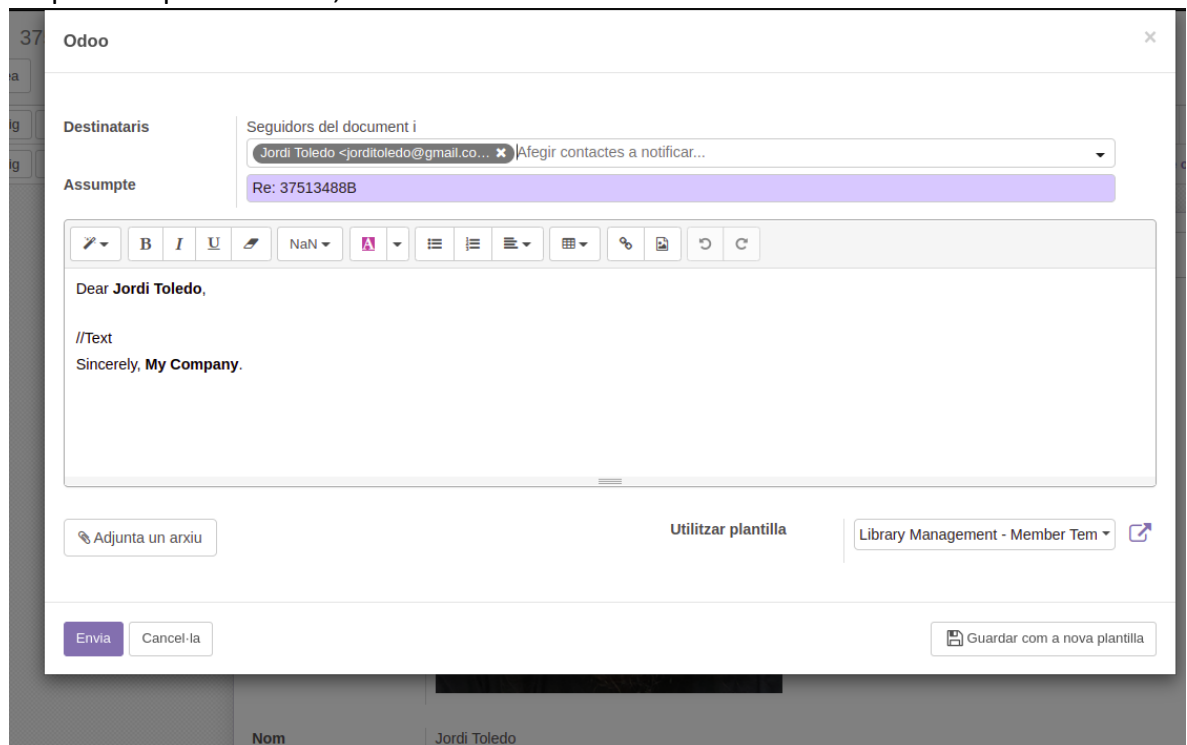


Figura 85: Captura de pantalla d'un correu base per a enviar a Jordi Toledo. Font: elaboració pròpia.

- **Perdonar càstig respecte llibre:** Si un membre es troba en estat castigat respecte llibres (cosa la qual li impedeix fer un préstec de llibres), al punxar en aquest botó, el càstig es desactiva i es passa a estat *Sense llibre llogat*.
- **Perdonar càstig respecte ordinador:** Si un membre es troba en estat castigat respecte ordinadors (cosa la qual li impedeix fer un préstec d'ordinadors), al punxar en aquest botó, el càstig es desactiva i es passa a estat *Sense ordinador llogat*.
- **Crear préstec:** Ens transporta a la vista formulari de préstecs, amb el membre ja seleccionat. S'observa a la figura 88.
- **Crear reserva:** Ens transporta a la vista formulari de reserves, amb el membre ja seleccionat. Es pot observar en la figura 95.

5.3.7.4 Altres funcionalitats

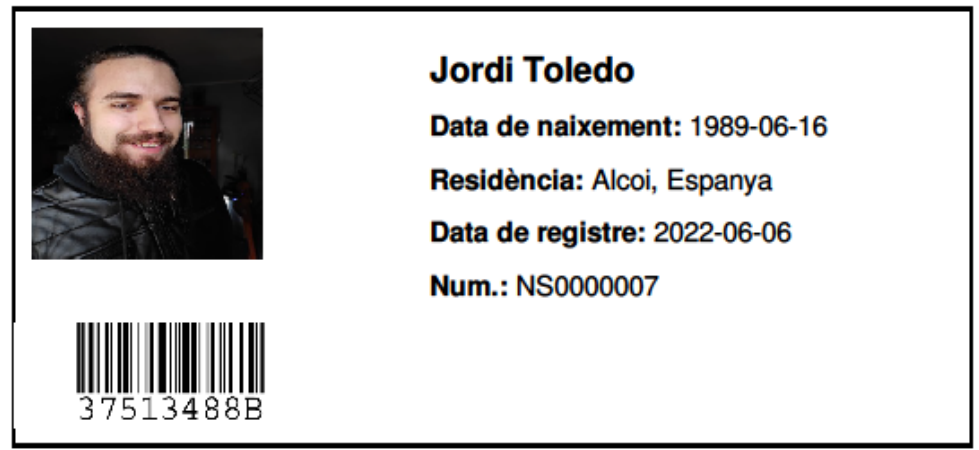
- **Passar el membre a “Amb llibre llogat”:** Quan es valida un préstec d'un llibre relacionat amb el membre, aquest membre automàticament passa a estat *Amb llibre llogat*, i causa que no pugui llogar cap altre llibre. A més, el número de préstecs del membre augmenta en 1.
- **Passar el membre a “Sense llibre llogat”:** Quan es finalitza el préstec, si el membre ha retornat el llibre a temps, aquest passa a estat *Sense llibre llogat*, i ja pot tornar a fer préstecs de llibres.
- **Passar el membre a “Castigat (respecte llibre)”:** Quan es finalitza el préstec, si el membre no ha retornat el llibre a temps, aquest passa a estat *Castigat*, i ja no pot tornar a fer préstecs de llibres.
- **Passar el membre a “Amb ordinador llogat”:** Quan es valida un préstec d'un ordinador relacionat amb el membre, aquest membre automàticament passa a estat *Amb ordinador llogat*, i causa que no pugui llogar cap altre ordinador. A més, el número de préstecs del membre augmenta en 1.
- **Passar el membre a “Sense ordinador llogat”:** Quan es finalitza el préstec, si el membre ha retornat l'ordinador a temps, aquest passa a estat *Sense ordinador llogat*, i ja pot tornar a fer préstecs d'ordinadors.
- **Passar el membre a “Castigat (respecte ordinador)”:** Quan es finalitza el préstec, si el membre no ha retornat l'ordinador a temps, aquest passa a estat *Castigat*, i ja no pot tornar a fer préstecs d'ordinadors.
- **Llevar càstig:** Existeix una acció automàtica, la qual s'executa cada dia. El que fa es comprovar els càstigs. L'acció calcula el temps que porta un membre amb el càstig, independentment de si és de llibre o ordinador. És una de les tres accions automàtiques nomenades al punt 5.2.2. Si fa 5 dies des de què fou posat, el càstig es perdona i el membre passa a estat *Sense llibre llogat*, podent tornar a llogar llibres, o *Sense ordinador llogat*, podent tornar a llogar ordinadors.

- **Augmentar reserves:** Quan es valida una reserva vinculada a un membre, el número de reserves d'aquest membre augmenta en 1 (si té un valor de 3 ja no permetrà afegir més i no es realitzarà la reserva). Quan es finalitza o cancel·la la reserva, aquest número disminueix en 1.

5.3.7.5 Informes

L'informe PDF que es pot obtindre d'aquest model és el carnet del membre:

My Company



Correu electrònic: info@yourcompany.com Web: <http://www.yourcompany.com>

Pàgina: 1 / 1

Figura 86: Carnet del membre Jordi Toledo. Font: elaboració pròpia.

5.3.8 Préstecs

5.3.8.1 Vistes des del menú

Al model de préstecs disposem d'una única forma de mostrar els objectes, llista.

<input type="checkbox"/>	Prèstec	Producte	Libre prestat	Ordinador prestat	Membre	Estat del préstec
<input type="checkbox"/>	OS000052	Libre	[ZN]One Piece 1(0000001.OP1.EO)		37513488B	Finalitzat
<input type="checkbox"/>	OS000051	Libre	[ZN]One Piece 2(0000002.OP2.EO)		89463750J	Finalitzat
<input type="checkbox"/>	OS000050	Libre	[ZN]One Piece 1(0000001.OP1.EO)		37513488B	Finalitzat
<input type="checkbox"/>	OS000049	Ordinador		[ZN]IBM 5200(0001.IBM)	89463750J	Finalitzat
<input type="checkbox"/>	OS000048	Ordinador		[ZN]IBM 5200(0001.IBM)	37513488B	Finalitzat
<input type="checkbox"/>	OS000047	Ordinador		[ZN]IBM 5200(0001.IBM)	37513488B	Finalitzat
<input type="checkbox"/>	OS000046	Libre	[ZN]One Piece 1(0000001.OP1.EO)		37513488B	Finalitzat

Figura 87: Captura de pantalla de la vista llista dels préstecs. Originària d'una base de dades personal. Font: elaboració pròpia.

5.3.8.2 Camps

Préstec: OS000053

Validar

Finalitzar Enviar per Email

Crear Validat Finalitzat

Actiu

Préstec	OS000053
Producte	Libre
Libre prestat	[ZN]One Piece 1(0000001.OP1.EO)
Nom del llibre	One Piece 1
Membre	37513488B
Nom del membre	Jordi Toledo
Data de préstec	28/06/2022
Dies	30

Enviar un missatge Registrar una nota

Seguint 1

Figura 88: Captura de pantalla del préstec OS000053. Originària d'una base de dades personal. Font: elaboració pròpia.

Camps:

- **Estat:** Aquest camp és un selector, i ens indica l'estat en el que es troba el préstec. En aquest cas, els estats segueixen una transició en la que una vegada passen d'estat, no poden tornar a l'anterior. Per defecte es troba en *Creat*. Potser: *Creat*, *Validat*, o *Finalitzat*.
- **Préstec:** Aquest camp es calcula automàticament, seguint la seqüència indicada al punt 5.2.2. És el nom que identifica a al préstec, i és únic. Aquest es tracta d'un camp que es calcula automàticament. L'ordre és creixent, és a dir, per exemple, aquest préstec és el número 52 que s'ha fet a la biblioteca.

- **Producte:** Indica el tipus de producte sobre el qual es fa el préstec. Pot ser un llibre o un ordinador. En funció del producte que es trie, es desbloquejarà el selector corresponent:
 - **Si es tria llibre:**
 - **Llibre prestat:** És un selector, on es tria el llibre que es vol prestar. Ací s'aplica la lectura del codi de barres per a filtrar, indicada al punt 5.3.1.2
 - **Nom del llibre:** S'introdueix automàticament, i és el nom del llibre prestat.
 - **Si es tria ordinador:**
 - **Ordinador prestat:** És un selector, on es tria l'ordinador que es vol prestar. Ací s'aplica la lectura del codi de barres per a filtrar, indicada al punt 5.3.5.2.
 - **Nom de l'ordinador:** S'introdueix automàticament, i és el nom de l'ordinador prestat.
- **Membre:** És un selector, on es tria el membre que efectua la reserva. Ací s'aplica la lectura del codi de barres per a filtrar, indicada al punt 5.3.7.2.
- **Nom del membre:** S'introdueix automàticament, i és el nom del membre que efectua la reserva.
- **Data de préstec:** S'introdueix automàticament, i és la data en la qual el préstec és creat.
- **Dies:** S'introdueix automàticament, i és el camp que ens indica els dies que té el client per a tornar el producte. Coincideix amb el temps de préstec del model de membre, i apareix quan es relaciona el préstec amb un membre.

- **Camps ocults:**
 - **Data de retorn:** S'introdueix automàticament, i és la data en la qual el préstec deu finalitzar, és a dir, la data màxima en la qual un membre pot tornar el producte (tot i que es deixen dos dies més per si coincideix amb el cap de setmana o festius). Aquest camp apareix quan el préstec es valida.

- **Perdut:** Aquest camp és un anomenat “check”. La seua funció és, abans de passar el préstec a finalitzat, marcar la casella si el membre ha perdut el producte, per tal que aquest passe automàticament a estat *Perdut* quan es finalitze el préstec.

Préstec	OS0000053
Producte	Libre
Llibre prestat	[ZN]One Piece 1(0000001.OP1.EO)
Nom del llibre	One Piece 1
Membre	37513488B
Nom del membre	Jordi Toledo
Data de préstec	28/06/2022
Dies	30
Data de retorn	28/07/2022
Perdut	<input type="checkbox"/>

Figura 89: Captura de pantalla del préstec OS0000053 validat. Originària d'una base de dades personal. Font: elaboració pròpia.

- **Data en la que es retorna:** S'introdueix automàticament, i és la data en la qual el préstec finalitza realment, és a dir, quan ha passat a estat *Finalitzat*. Aquest camp apareix quan el préstec es valida.

Préstec	OS0000053
Producte	Libre
Llibre prestat	[ZN]One Piece 1(0000001.OP1.EO)
Nom del llibre	One Piece 1
Membre	37513488B
Nom del membre	Jordi Toledo
Data de préstec	28/06/2022
Dies	30
Data de retorn	28/07/2022
Data en la que es retorna	28/06/2022

Figura 90: Captura de pantalla del préstec OS0000053 finalitzat. Originària d'una base de dades personal. Font: elaboració pròpia.

5.3.8.3 Botons i funcions

- **Actiu:** Permet arxivar un préstec per tal que no siga utilitzat. Es pot desarxivar després.
- **Enviar correu:** Ens permet enviar un correu al membre que ha efectuat el préstec. S'activa únicament quan el préstec es troba en estat *Validat*. Existeix una plantilla per on iniciar, amb el correu del membre com a destinatari:

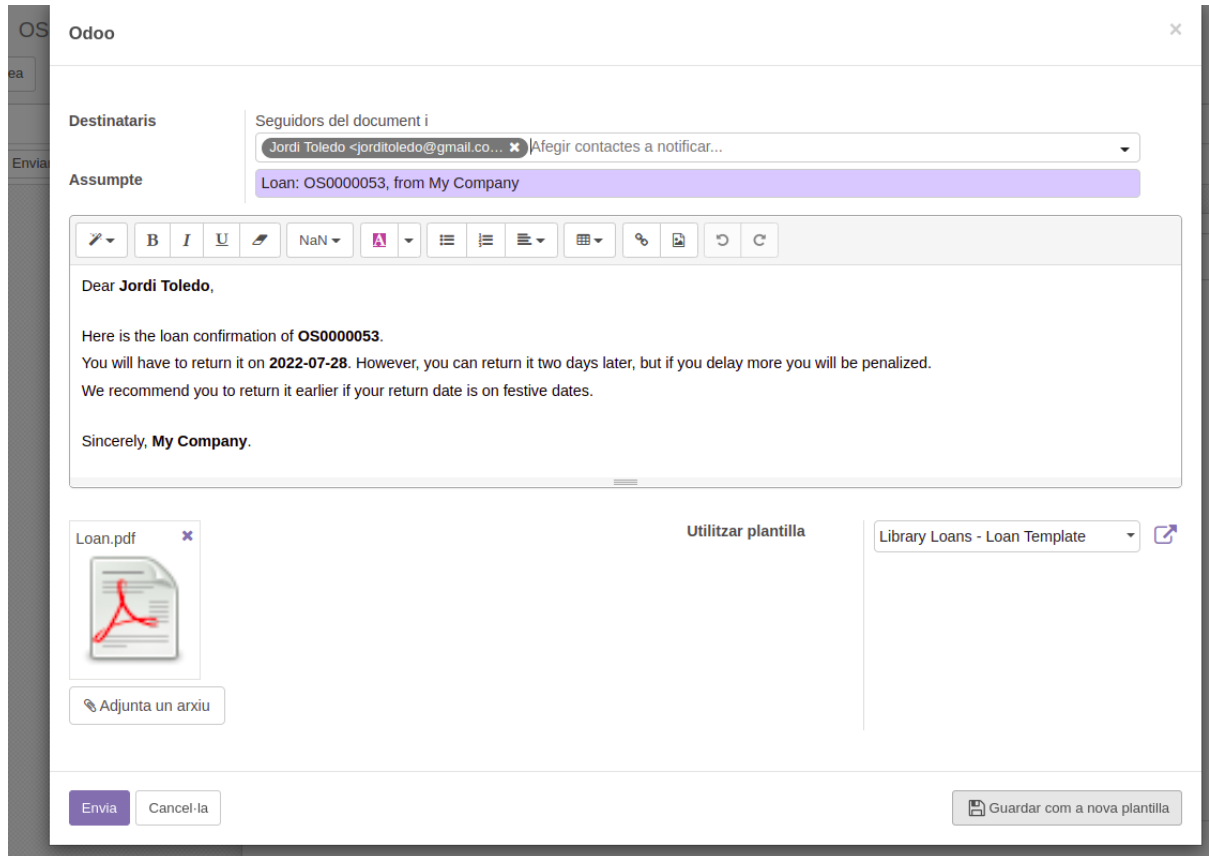


Figura 91: Captura de pantalla d'un correu base per a enviar a Jordi Toledo, del préstec OS0000053. S'adjunta l'informe del préstec. Font: elaboració pròpia.

- **Validar:** Per defecte un préstec es troba en estat Creat quan es guarda. Al punxar en el botó de *Validar*, el préstec passa a estat *Validat* i apareix la data de fi del préstec i el check de Perdut, així com l'opció d'enviar un email. També passa l'estat del membre de *Sense llibre/ordinador llogat*, a *Amb llibre/ordinador llogat* i passa el producte a estat *Llogat*.
- **Finalitzar:** Al punxar en el botó de *Finalitzar*, el préstec passa a estat *Finalitzat*. També, com hem dit anteriorment, els estats canvien: l'estat del membre respecte al producte llogat, pot passar de *Amb llibre/ordinador llogat* a *Sense llibre/ordinador llogat* si ha tornat el producte a temps, o a *Castigat* si no ho ha fet. El producte passarà d'estat *Llogat* a estat *Disponible*, si el membre l'ha retornat, o *Perdut* si el membre no l'ha retornat (s'indica marcant la casella de perdut).

5.3.8.4 Altres funcionalitats

- **Informar sobre el fi dels préstecs:** Es tracta d'una acció automàtica, que s'executa cada dia. El que fa és comprovar els préstecs que es troben encara en estat *Validat*. L'acció calcula els préstecs que queda 2 dies per a que s'arribe a la data de fi del préstec i envia un correu a cada un dels membres que formen part d'aquests préstecs informant de la situació. És una de les tres accions automàtiques nombrades al punt 5.2.2.



Figura 92: Captura de pantalla d'un correu enviat a Roberto Marín, recordant que queden 2 dies per a finalitzar el préstec. Font: elaboració pròpia.

5.3.8.5 Informes

L'informe PDF que es pot obtenir d'aquest model és el del préstec:

My Company

OS0000053:

Llibre: One Piece 1	Membre: Jordi Toledo
Data d'alquiler: 2022-06-28	Data de retorn 2022-07-28

Correu electrònic: info@yourcompany.com Web: <http://www.yourcompany.com>

Pàgina: 1 / 1

Figura 93: PDF del préstec OS0000053 inserit al document. Font: elaboració pròpia.

5.3.9 Reserves

5.3.9.1 Vistes des del menú

Al model de préstecs disposem d'una única forma de mostrar els objectes, llista.

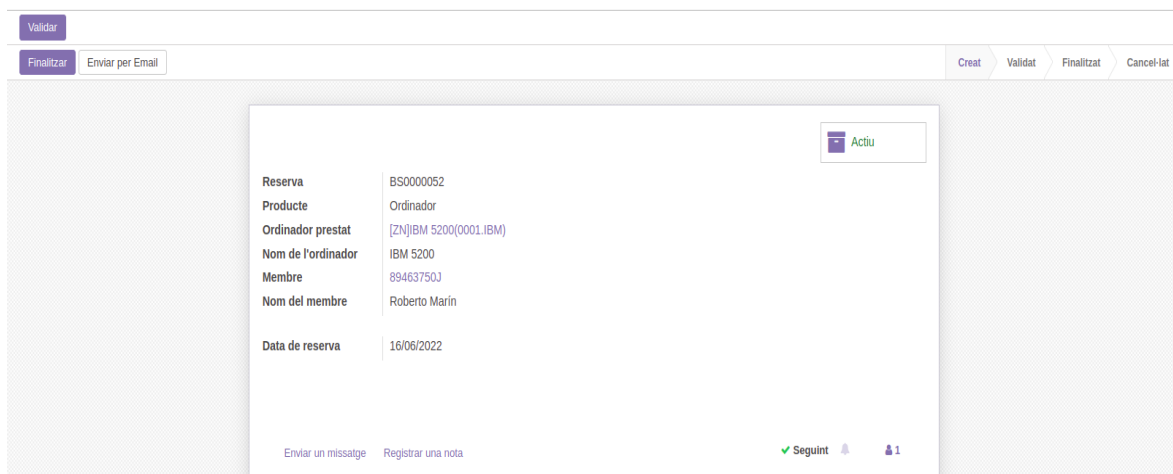


Reserva	Producte	Llibre prestat	Ordinador prestat	Membre	Estat de reserva
BS0000053	Ordinador		[ZN]IBM 5200(0001.IBM)	89463750J	Cancel·lat
BS0000052	Ordinador		[ZN]IBM 5200(0001.IBM)	89463750J	Finalitzat

Figura 94: Captura de pantalla de la vista llista de les reserves. Originària d'una base de dades personal. Font: elaboració pròpia.

5.3.9.2 Camps

Reserva: BS0000052



Reserva: BS0000052
Producte: Ordinador
Ordinador prestat: [ZN]IBM 5200(0001.IBM)
Nom de l'ordinador: IBM 5200
Membre: 89463750J
Nom del membre: Roberto Marín
Data de reserva: 16/06/2022

Figura 95: Captura de pantalla de la reserva BS0000052. Originària d'una base de dades personal. Font: elaboració pròpia.

Camps:

- **Estat:** Aquest camp és un selector, i ens indica l'estat en el que es troba la reserva. Igual que el préstec, es segueix una transició, en la que una vegada es passa d'estat, ja no es pot tornar enrere. Per defecte es troba en *Creat*. Potser: *Creat*, *Validat*, *Finalitzat* o *Cancel·lat*.
- **Reserva:** Aquest camp es calcula automàticament, És el nom que identifica a la reserva, i és únic. Igual que el nom del préstec, es calculen automàticament en ordre creixent. Per exemple, aquesta reserva és la número 52 que s'ha fet a la biblioteca.

- **Producte:** Indica el tipus de producte sobre el qual es fa la reserva. Pot ser un llibre o un ordinador. En funció del producte que es trie, es desbloquejarà el selector corresponent:
 - **Si es tria llibre:**
 - **Llibre prestat:** És un selector, on es tria el llibre que es vol reservar. Ací s'aplica la lectura del codi de barres per a filtrar, com s'ha indicat al punt 5.3.1.2.
 - **Nom del llibre:** S'introdueix automàticament, i és el nom del llibre reservat.
 - **Si es tria ordinador:**
 - **Ordinador prestat:** És un selector, on es tria l'ordinador que es vol reservar. Ací s'aplica la lectura del codi de barres per a filtrar, com s'ha indicat al punt 5.3.5.2.
 - **Nom de l'ordinador:** S'introdueix automàticament, i és el nom de l'ordinador reservat.
 - **Membre:** És un selector, on es tria el membre que efectua la reserva. Ací s'aplica la lectura del codi de barres per a filtrar, com s'ha indicat al punt 5.3.7.2.
 - **Nom del membre:** S'introdueix automàticament, i és el nom del membre que efectua la reserva.
 - **Data de reserva:** S'introdueix automàticament. És la data en la que es crea la reserva.
- Camps ocults:**
- **Data d'expiració reserva:** S'introdueix automàticament una vegada la reserva es valida. És la data en la que la reserva finalitza, que es de 15 dies més que la data de reserva.

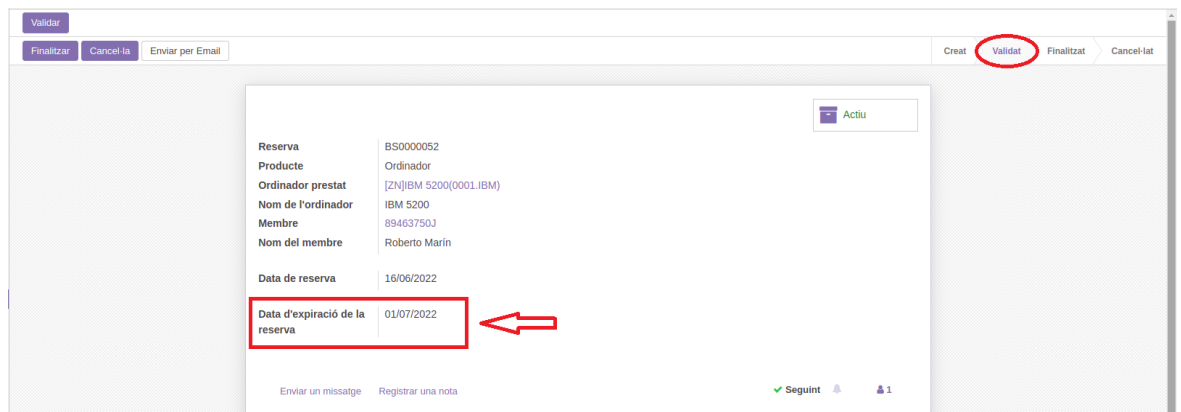


Figura 96: Captura de pantalla de la reserva BS0000052 validada. Originària d'una base de dades personal. Font: elaboració pròpia.

5.3.9.3 Botons i funcions

Botons:

- **Actiu:** Permet arxivar una reserva per tal que no siga utilitzada. Es pot desarxivar després.
- **Enviar correu:** Ens permet enviar un correu al membre que ha efectuat la reserva. S'activa únicament quan la reserva es troba en estat *Validat*. Existeix una plantilla per on iniciar, amb el correu del membre com a destinatari:

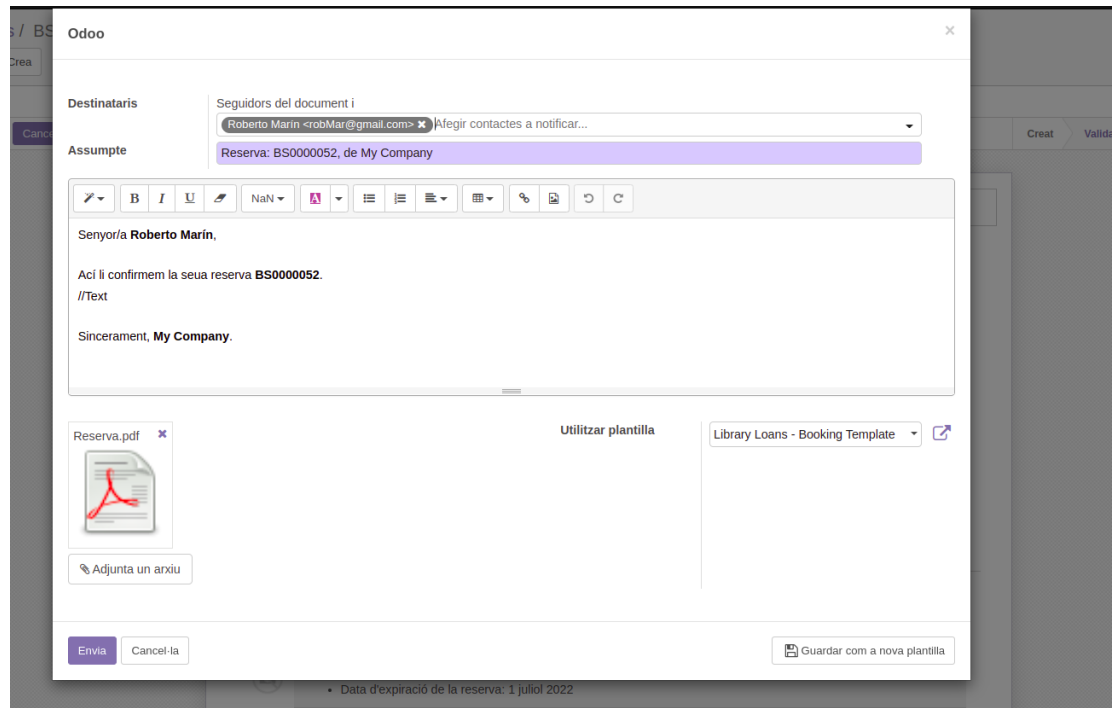


Figura 97: Captura de pantalla d'un correu base per a enviar a Roberto Marín, de la reserva BS0000052. S'adjunta l'informe de la reserva. Font: elaboració pròpia.

- **Validar:** Per defecte una reserva es troba en estat *Creat* quan es guarda. Al punxar en el botó de *Validar*, la reserva passa a estat *Validat* i apareix la data de fi de reserva, així com l'opció d'enviar un email. També, com hem dit anteriorment, augmenta el número de reserves del membre i passa el producte a estat *Reservat*.
- **Finalitzar:** Al punxar en el botó de *Finalitzar*, la reserva passa a estat *Finalitzat* i apareix l'opció de crear un préstec corresponent a eixa reserva. També, com hem dit anteriorment, disminueix el número de reserves del membre i passa el producte a estat *No Reservat*.
- **Cancel·lar:** Al punxar en el botó de *Cancel·lar*, la reserva passa a estat *Cancel·lat*. També, com hem dit anteriorment, disminueix el número de reserves del membre i passa el producte a estat *No Reservat*.

- **Crear préstec:** Ens transporta a la vista formulari de préstecs, amb el membre i producte ja seleccionats. Eixa vista es troba a la figura 88. Aquesta opció s'activa únicament quan la reserva passa a estat *Finalitzat*.

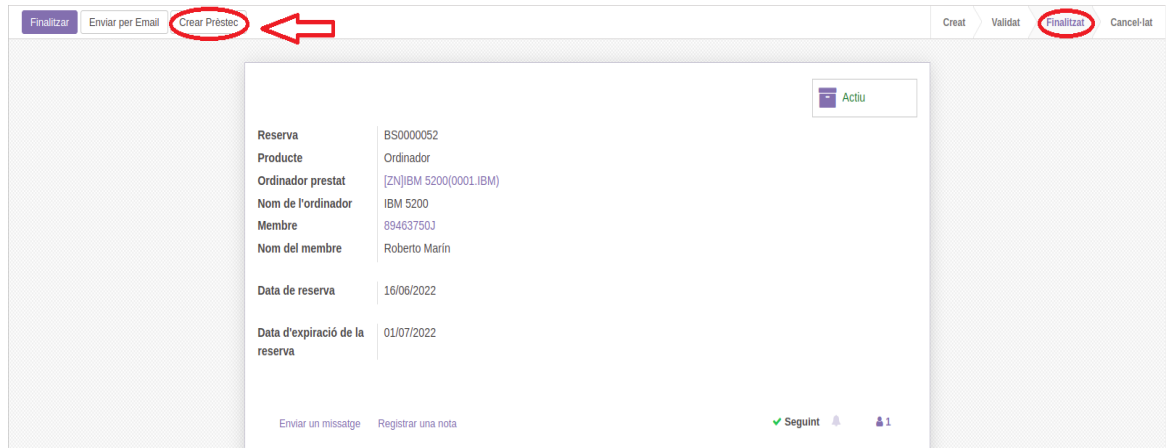


Figura 98: Captura de pantalla de la reserva BS0000052 finalitzada. Originària d'una base de dades personal. Font: elaboració pròpia.

5.3.9.4 Altres funcionalitats

- **Cancel·lar reserves:** Es tracta d'una acció automàtica, la qual s'executa cada dia. El que fa es comprovar les reserves que es troben encara en estat *Validat*. L'acció calcula les reserves que tenen ja quinze dies d'antiguitat, i les passa a estat *Cancel·lat*. També es passa el producte a estat No Reservat i es descompta una reserva del número de reserves del membre a qui està assignada. És una de les tres accions automàtiques indicades al punt 5.2.2.

5.3.9.5 Informes

L'informe PDF que es pot obtenir d'aquest model és el del préstec:

My Company

BS0000052:

Ordinador: IBM 5200	Membre: Roberto Marín
Data de reserva: 2022-06-16	Data de fi: 2022-07-01

Correu electrònic: info@yourcompany.com Web: <http://www.yourcompany.com>

Pàgina: 1 / 1

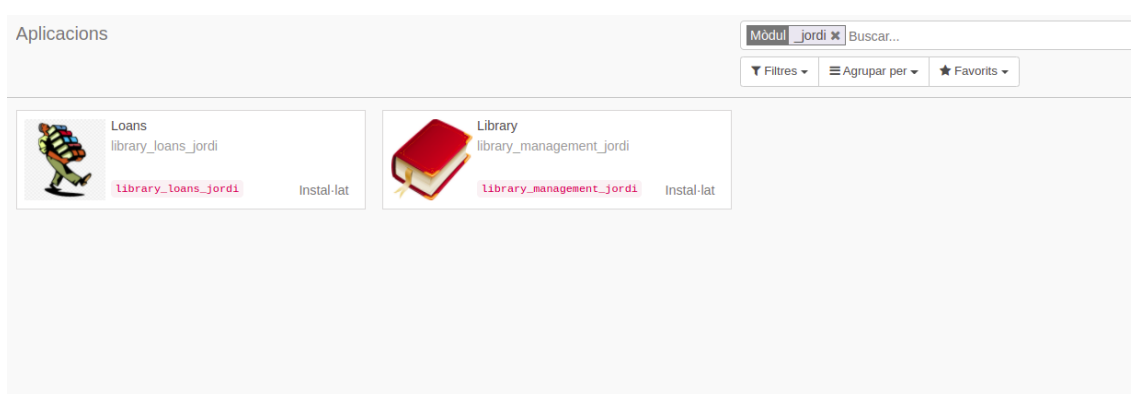
Figura 99: PDF de la reserva BS0000052 inserida al document. Font: elaboració pròpia.

6. Implementació de la solució

Una vegada que s'ha desenvolupat el programari i es disposa del mòdul complet, és hora de posar en pràctica allò que s'ha programat. Per a executar Odoo, aquest deu estar instal·lat en un servidor de l'organització, i si es vol accedir a mòduls especials (com el que s'ha desenvolupat), aquests repositoris deuen estar a la carpeta d'Odoo. Partint de que ja tenim Odoo al servidor de la biblioteca i corrent, primer procedirem a instal·lar els dos mòduls.

6.1 Implantació de l'ERP en la biblioteca

Per tal d'instal·lar correctament els mòduls al nostre ERP, en Odoo, hem de dirigir-nos a les Aplicacions. Ací tindrem per a instal·lar tots els repositoris dels que disposem a Odoo.



*Figura 100: Captura de pantalla dels mòduls a les aplicacions de Odoo.
Font: elaboració pròpia.*

Els mòduls disposen d'una imatge, i una descripció sobre el funcionament que tenen, al fitxer README.

6.2 Manteniment

Una vegada està ja implementat a la biblioteca els mòduls i l'ERP es troba funcionant, entra la fase de manteniment. Segons Solbyte (2022), aquesta fase consisteix en introduir noves funcionalitats al software, moltes de les quals poden ser sol·licitades directament per el client, i altres que tracten de corregir errors. En aquest cas, es farà un manteniment constant dels mòduls, i això ha donat lloc a la creació de dues funcionalitats noves:

- **Filtres d'autors i editorials:** Quan entrem en el menú d'objectes d'un model de dades, aquest ens mostra tots els objectes corresponent a eixe model. Per exemple, si punxem en el menú, als Llibres, ens mostrarà tots els llibres. Aquests poden ser més de 1000, i per això existeix un cercador. No obstant això, Odoo permet el desenvolupament de filtres de cerca, i això és el que s'ha fet per als autors i les editorials. Ha sigut implementat al mòdul de gestió principal.

- **Més de 10 llibres:** Per als autors, s'ha creat un filtre de cerca que ens mostra els autors que han escrit més de 10 llibres.

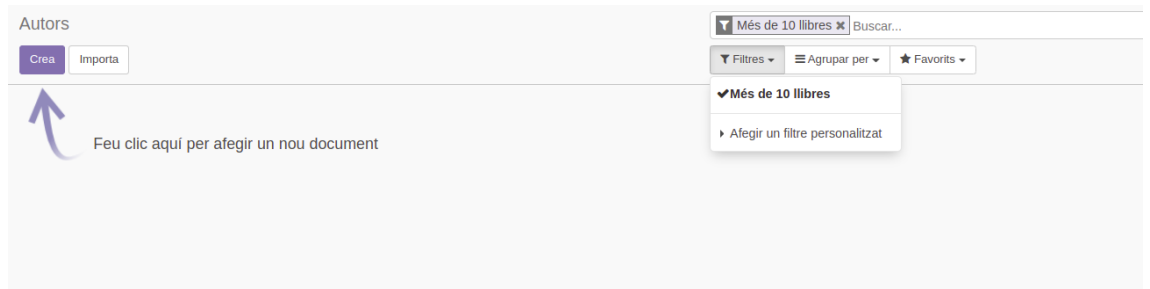


Figura 101: Captura de pantalla del filtre dels autors. Originària d'una base de dades personal. Font: elaboració pròpia.

- **Més de 50 llibres:** Per a les editorials, s'ha creat un filtre de cerca que ens mostra les editorials que han publicat més de 50 llibres.

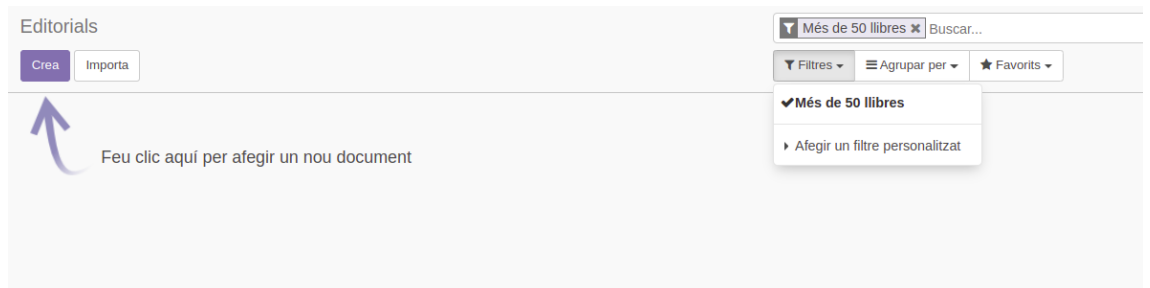


Figura 102: Captura de pantalla del filtre de les editorials. Originària d'una base de dades personal. Font: elaboració pròpia.

- **Informe d'estadístiques:** S'ha afegit un altre botó al menú, Estadístiques.

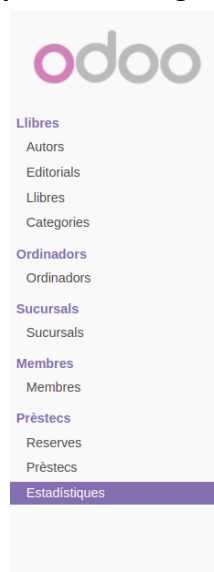
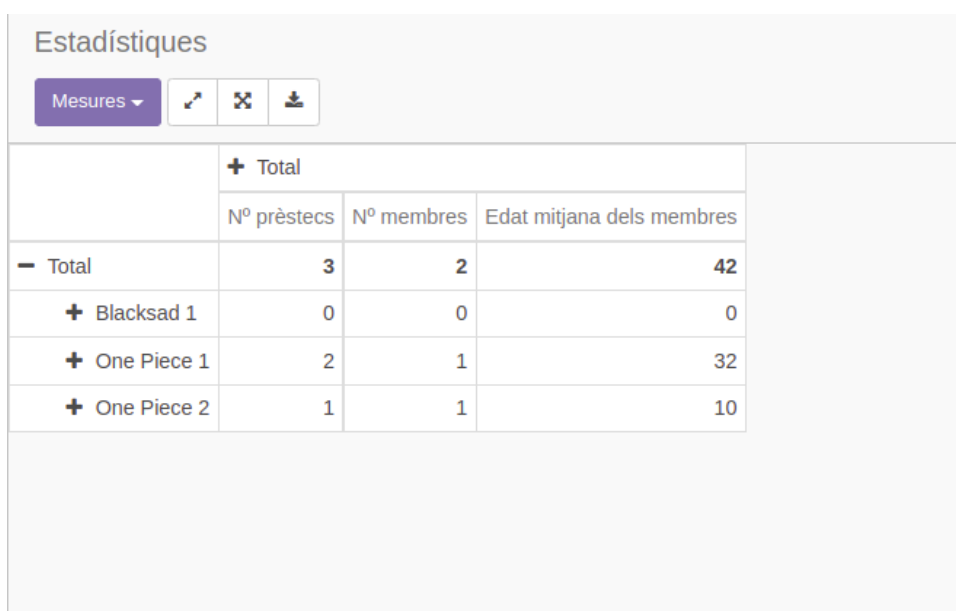


Figura 103: Captura de pantalla nou menú. Originària d'una base de dades personal. Font: elaboració pròpia.

Aquest botó el que fa és transportant-nos a una taula dinàmica, que s'actualitza constantment a mesura que anem afegint objectes a la base de dades. Aquesta taula ve d'un fitxer nou de vistes que s'ha creat (library_statistics.xml), i obté les dades dels fitxers .py. Aquesta funcionalitat ha sigut inclosa al mòdul de préstecs. Ens mostra per a cada llibre:

- N° de préstecs: Les vegades que ha sigut prestat.
- N° de membres: El número diferent de membres que han llogat el llibre.
- Edat mitjana dels membres: Representa l'edat mitjana dels diferents membres que han llogat el llibre. Si únicament l'ha llogat un membre, es mostrarà l'edat d'eixe membre.



Estadístiques			
Mesures ▾	↗	✕	↓
	+ Total		
	Nº préstecs	Nº membres	Edat mitjana dels membres
- Total	3	2	42
+ Blacksad 1	0	0	0
+ One Piece 1	2	1	32
+ One Piece 2	1	1	10

Figura 104: Captura de pantalla la taula dinàmica. Originària d'una base de dades personal. Font: elaboració pròpia.

6.3 Temps de vida del producte

Una vegada s'ha implantat el nostre producte i està en funcionament, resulta interessant tractar de fer una predicció sobre de quin tipus de producte es tracta segons la matriu BCG, el temps de vida que té, i quina rendibilitat pot tindre, ja que si bé el model de la biblioteca no busca obtindre ingressos, els desenvolupadors del software sí.

La matriu BCG, segons Unir (2022), consisteix en una ferramenta que utilitzen les empreses per a estudiar la rendibilitat d'un producte i determinar les millors estratègies de venda.

La matriu BCG es compon de quatre tipus de productes:

- Interrogants: Productes amb una quota de mercat baixa però que ofereixen una perspectiva de creixement elevat al mercat. Pot evolucionar a producte estrella.
- Estrelles: Productes amb una alta quota de mercat i que també ofereixen una perspectiva de creixement elevat del mercat. Normalment són els productes que anteriorment eren interrogants, i que li donen a la empresa una elevada participació en el mercat. És on més inverteix la empresa.
- Vaques: Productes amb una alta quota de mercat, però ja assentats al mercat. Serveixen com a fonts d'ingressos per a la empresa, ja que no cal invertir massa.
- Gossos: Productes que ja no tenen una alta quota de mercat i no generen ingressos degut a la poca participació en el mercat que tenen.

El nostre producte, els mòduls per a la gestió de biblioteques serien en aquest moment un producte interrogant, ja que és una nova oportunitat de mercat i no gaudiria d'una elevada quota de mercat.

 Mòduls d'odoo.	 Mòdul per a gestió de biblioteques.
	

Figura 105: Matriu BCG que indica la situació del mercat dels mòduls d'Odoo i del desenvolupat al treball. Font: elaboració pròpia.

Si tot funcionés correctament i les biblioteques municipals li donen un bon ús, es podria pensar en estendre la funcionalitat a biblioteques privades i en sort que passés a ser un producte estrela. Si es complira, la quota de mercat que tindria seria alta i tindria una elevada participació en el sector, de manera que els mòduls poden tindre futur.

Finalment, quan el producte arribés a ser un producte vaca, els ingressos que ens donés podrien ser utilitzats per a repetir la jugada, creant un mòdul en Odoo per a un altre negoci que el pugui necessitar, per exemple per a gestionar un gimnàs. Així, com a desenvolupador software, es podria gaudir de productes innovadors i que generen beneficis econòmics centrant-se únicament en explotar Odoo, demostrant que tant aquest producte com futurs poden ser molt rentables i tindre un gran futur.

7. Conclusions i línies futures

La realització d'aquest TFG ha sigut un treball laboriós, que ha portat uns 5 mesos de desenvolupament, però el resultat final ha sigut una aplicació que compleix amb els propòsits per els quals va ser dissenyada, i més.

Ha sigut un projecte software que ha inclòs totes les etapes. Començant per la identificació d'una problemàtica resoldre, com és el desenvolupament d'un mòdul de gestió bibliotecària a Odoo per què puga ser implantat a biblioteques. Seguint per la identificació de requeriments, on s'han valorat fins a 6 solucions possibles. S'han dissenyat les interfícies, l'estructura de la base de dades i la lògica, i després s'ha desenvolupat el projecte.

El resultat final és un parell de mòduls que complementats poden ser utilitzats amb una implantació Odoo a una biblioteca. Es tracta d'un programari que pot ser utilitzat per una empresa u organització per tal de ser comercialitzat per a un client, i oferint manteniment sobre aquest.

Personalment, es tracta d'un projecte que des de que fou plantejat, s'ha desenvolupat i tractat amb tota la cura i passió possible per tal que funcionara. És el primer gran projecte que he realitzat, i espere que siga el primer de molts. Programar amb Odoo ha sigut una nova experiència per a mi, ja que si bé he après a programar al llarg de la titulació d'Enginyeria Informàtica, mai havia programat amb el framework d'Odoo. No obstant això, aprendre noves maneres de programar, noves maneres d'interpretar problemes i altres punts de vista per a implementar solucions sempre es positiu.

Finalment, com a línies futures per als mòduls, sempre es poden aplicar millores que demane una biblioteca en particular. Seria una bona idea incloure una web per al mòdul, on incloure les col·leccions de llibres que es disposen, per exemple. També es podria buscar una forma de vincular les compres de la biblioteca amb els productes, ja que quan es realitza una compra d'un producte, no s'afegeix el producte a l'ERP i es té que crear. Es tracta de millores que faciliten més la comoditat del bibliotecari. A més, la idea és desenvolupar més mòduls per a implementacions Odoo en altres negocis, com un mòdul de gestió de gimnasos o de tallers tèxtils, on es posaria en pràctica tot allò que s'ha après al desenvolupar aquest projecte, i així aprofitar l'oportunitat de negoci que dona l'Odoo Development.

8. Bibliografia

Delgado Quintero, S. (2022). Aprende Python.

Martínez, L., Forés, B., Puig, A., Lapiedra, R., Tomás, J. V. & Ramiro, P. J. (2021). ODOO. Sistema de información en la empresa. Manual básico para estudiantes. *Editorial Universitat Jaume I*.

Odoo 11. Carpeta amb els fitxers base d'Odoo 11 a Github.

< <https://github.com/odoo/odoo/tree/11.0> >[Consulta: 2 de juny de 2022]

Odoo Mates (2022). Tutorials sobre desenvolupament software a Odoo.

< <https://www.youtube.com/channel/OdooMates> >[Consulta: 2 de juny de 2022]

Solbyte (2022). 5 fases del proceso de desarrollo software.

< <https://www.solbyte.com/blog/desarrollo-de-software/> >[Consulta: 9 de juny de 2022]

Tomás-Miquel, J. V., Expósito-Langa, M., & Capó-Vicedo, J. (2008). Los sistemas ERP en la práctica. *Editorial UPV*.

Unir (2022). Consulta sobre la búsqueda: Matriz BCG.

< <https://www.unir.net/empresa/revista/matriz-bcg/> >[Consulta: 6 de juny de 2022]

Viquipèdia (2022). Consulta sobre la búsqueda: ERP.

< <https://ca.wikipedia.org/wiki/ERP> >[Consulta: 2 de juny de 2022]

Viquipèdia (2022). Consulta sobre la búsqueda: Odoo.

< <https://ca.wikipedia.org/wiki/Odoo> >[Consulta: 2 de juny de 2022]

Nota: S'adjunta tot el codi i fitxers desenvolupats en l'entrega amb un fitxer .zip.