

Laia Domingo Ciurana

Interoperabilitat entre Plataformes E-commerce i ERP

TREBALL DE FI DE GRAU

dirigit per Antoni Martínez Ballesté

**Doble Titulació de Grau en Administració i Direcció d'Empreses
i Tècniques de Desenvolupament d'Aplicacions Web i Mòbils**



UNIVERSITAT ROVIRA I VIRGILI

**Tarragona
(juny 2024)**

Resum

Aquest treball de Final de Grau es centra en desenvolupar una aplicació que faciliti la interoperabilitat entre plataformes de comerç electrònic, sistemes de gestió empresarial, i diversos ERP. Es pretén automatitzar el procés d'importació i exportació de les dades entre dues plataformes, per tal de poder solucionar les limitacions actuals entre les plataformes a causa de la falta de compatibilitat directa. Això permetrà una gestió més eficient i reduirà el temps i els errors que es pot provocar amb les transferències manuals de les dades.

Resumen

Este trabajo de Final de Grado se centra en desarrollar una aplicación que facilite la interoperabilidad entre plataformas de e-commerce, sistemas de gestión empresarial y varios ERP. Se pretende automatizar el proceso de importación y exportación de los datos entre dos plataformas, para poder solucionar las limitaciones actuales entre las plataformas a causa de la falta de compatibilidad directa. Esto permitirá una gestión más eficiente y reducirá el tiempo y los errores que se puede provocar con las transferencias manuales de los datos.

Abstract

This Final Degree work is focused on developing an application which facilitates the interoperability between e-commerce platforms, business management systems, and several ERP. It is intended to automate the process of importing and exporting data between two platforms, in order to solve the current limitations between the platforms due to the lack of direct compatibility. This will allow a more efficient management and reduce the time and errors that can be caused by manual data transfers.

Taula de contingut

<u>1</u>	<u>INTRODUCCIÓ</u>	4
1.1	PROGRAMARI RETAIL	4
1.2	OBJECTIUS DEL TREBALL	6
<u>2</u>	<u>FONAMENTS</u>	7
2.1	PRESTASHOP.....	7
2.2	HIOFFICE	11
2.3	DATA BRIDGE	18
2.4	INTEROPERABILITAT.....	19
<u>3</u>	<u>DISSENY</u>	22
3.1	FUNCIONALITATS	22
3.2	DADES D'ENTRADA I SORTIDA	24
3.3	DIAGRAMA DE FLUX DE DADES.....	26
<u>4</u>	<u>IMPLEMENTACIÓ</u>	29
4.1	LLENGUATGE I PLATAFORMA.....	29
4.2	ARQUITECTURA DEL SISTEMA	30
4.3	DETALL D'IMPLEMENTACIÓ	36
<u>5</u>	<u>PROVES</u>	38
<u>6</u>	<u>IMPLANTACIÓ</u>	41
6.1	REQUISITS MÍNIMS	41
6.2	CONFIGURACIÓ.....	41
6.3	PRESSUPOST	42
<u>7</u>	<u>CONCLUSIONS</u>	43
7.1	ASSOLIMENT DELS OBJECTIUS	43
7.2	DESAFIAMENTS TROBATS.....	44
7.3	MILLORES I TREBALL FUTUR.....	46
7.4	APRENENTATGES.....	46
	<u>REFERÈNCIES BIBLIOGRÀFIQUES</u>	47

1 Introducció

1.1 Programari retail

En les últimes dècades el comerç electrònic ha fet una gran expansió, sobretot en els darrers anys, impulsat per la pandèmia del COVID-19. És considerat un dels sectors que mou més diners en tot el món, i aquesta situació es preveu que creixerà de manera important en les pròximes dècades a causa de la implementació de noves tecnologies i la penetració d'Internet cada vegada més elevada.

El comerç electrònic es pot definir com totes aquelles transaccions comercials que es duen a terme utilitzant Internet com a mitjà. El comerç electrònic es pot dur a terme gràcies a les aplicacions especialitzades en aquest sector.

Les aplicacions de comerç electrònic estan fetes per tal de facilitar les operacions comercials a través d'Internet, oferint una gran quantitat de funcions essencials que permeten gestionar als comerciants els negocis de manera més eficient. Algunes de les funcions més importants que ofereixen les aplicacions de comerç electrònic són el control del stock, les formes de pagament, el màrqueting i promocions i les anàlisis.

El control del stock permet rastrejar i gestionar les existències en temps real, assegurant de tal manera que els nivells d'stock s'actualitzen de manera automàtica quan es realitzen les vendes. A més, la majoria de les aplicacions ofereixen la funcionalitat de notificar al comerciant quan l'inventari es troba a baixos nivells, permetent d'aquesta manera la reposició dels productes.

Una altra funcionalitat essencial és la integració amb les passarel·les de pagament. Els mètodes de pagament són una part essencial de l'experiència del comerç electrònic. A mesura que el sector ha anat creixent, els mètodes a través l'usuari paga s'han anat diversificant. De manera global els mètodes de pagament més típics són les targetes de dèbit i crèdit, aquests mètodes permeten als clients fer els pagaments de forma ràpida i còmoda, s'ofereix aquest tipus de pagament a través de serveis com PayPal i Stripe. Altres avui en dia, també incorporen Bizum, una solució de pagament ràpida i segura que permet pagaments instantanis (també entre particulars) a través dels dispositius mòbils. També hi ha l'opció del pagament a través de transferències bancàries, permetent al client moure els diners del seu compte bancari al compte bancari de l'empresa. Pel que fa a la seguretat de cada un d'aquests pagaments depèn de cada entitat bancària, ja que cada una d'elles és responsable d'establir els seus propis límits.

En termes de màrqueting i promocions, les aplicacions de comerç electrònic solen incloure eines per tal de crear i gestionar descomptes, ofertes especials i codis promocionals. A més, la majoria d'elles integren serveis de màrqueting per correu electrònic, el que permet enviar promocions i actualitzacions directament al client, mantenint d'aquesta manera una comunicació efectiva amb el client i poder fonamentar la lleialtat del client.

Finalment, les capacitats d'anàlisi són essencials, són eines que ofereixen informes detallats sobre les vendes, el rendiment dels productes i el comportament dels clients. Les empreses poden analitzar aquestes dades per tal d'identificar patrons de compra i conèixer de manera més profunda els seus clients.

1.1.1 Plataformes Online

Per crear i gestionar el comerç electrònic, existeixen diverses plataformes en línia. Dues de les més destacades són *PrestaShop* i *WordPress*, aquesta última quan es combina amb el plugin *WooCommerce*.

PrestaShop és una plataforma de comerç electrònic de codi obert que permet als usuaris crear i gestionar les botigues online d'una manera senzilla i flexible. Entre les funcionalitats que ofereix es troba la gestió de productes i les seves combinacions, que facilita afegir, editar i categoritzar els productes, així com gestionar l'inventari. La plataforma també permet una àmplia personalització a través de mòduls i temes, cosa que permet a l'usuari la creació d'una pàgina web única i adaptada a les necessitats específiques de cadascun. A més, ofereix eines avançades de màrqueting que ajuden a optimitzar la visibilitat de la pàgina en motors de cerca i gestionar campanyes de màrqueting.

Per altra banda, *WordPress*, és una solució potent per fer webs que, gràcies al "plugin"¹ *WooCommerce*, poden esdevenir webs per a comerç electrònic. És conegut per la seva interfície molt amigable i la seva comunitat de suport, el que facilita la utilització fins i tot per aquells que no tenen experiència tècnica. *WooCommerce*, el plugin combinat amb *WordPress*, afegeix funcionalitats específiques de comerç electrònic permetent als usuaris gestionar les botigues online de manera fàcil. La plataforma, igual que *PrestaShop*, permet agregar temes i plugins fent la pàgina el més personalitzada i única possible. També inclou eines de màrqueting integrades, que ajuden a donar visibilitat en línia.

1.1.2 Software Utilitzat en Botiga Física

Avui en dia, en les botigues físiques del sector *retail* o comerç al detall, es fa ús de diferents solucions de software per gestionar les operacions diàries de manera eficient. Aquests sistemes ajuden al punt de venda i a més a més permeten una gestió integral del negoci.

Els sistemes de punt de venda (o TPV, terminals punt de venda) són necessaris per a gestionar les diferents transaccions. Aquests sistemes faciliten el processament de les vendes, la gestió d'inventaris i la generació de rebuts, així com les factures a client o els albarans de compra, entre altres. Els sistemes de punt de venda, permeten també diversos mètodes de pagament com poden ser les targetes de crèdit i dèbit, i actualitzen l'inventari a temps real a mesura que es van fent vendes a la botiga física.

A més de la solució explicada ara, la majoria de les botigues físiques utilitzen sistemes de gestió empresarial, coneguts com a ERP (Enterprise Resource Planning). Aquests sistemes integren totes les àrees de negoci en una única plataforma, proporcionant d'aquesta manera una visió global i centralitzada de les operacions.

Els sistemes ERP permeten a les botigues gestionar de manera més eficient tots els aspectes del seu negoci, des del subministrament o les factures de venda o albarans de

¹ *WordPress*, com la majoria de sistemes CMS (de l'anglès Content Management System) es pot configurar i ampliar amb un seguit d'afegitons o plugins.

compra fins a les finances i la comptabilitat, ja que per exemple fan de manera automàtica els assentaments comptables de l'empresa. D'aquesta manera, faciliten la planificació i el control dels recursos, millorant l'eficiència operativa i la presa de les decisions. Aquests sistemes permeten reduir el marge d'error i millorar la coherència de la informació.

1.2 Objectius del treball

Un dels problemes més habituals avui en dia en el sector del programari de comerç és la interoperabilitat entre les diferents plataformes de comerç electrònic i els diferents sistemes de gestió empresarial. Avui en dia no hi ha gairebé cap aplicació que permeti una interoperabilitat amb d'altres. Aquesta falta d'interoperabilitat en el sector té una sèrie d'inconvenients un dels principals és la quantitat de processos manuals que s'han de seguir per tal de transferir les dades d'una plataforma a l'altra, la duplicació de dades i l'augment de risc d'error.

El present Treball de Fi de Grau intenta abordar aquesta problemàtica a través del desenvolupament d'una aplicació destinada a facilitar l'exportació i importació de dades entre dues plataformes. Això s'assoleix adaptant les dades amb la fi de què es puguin transferir de manera eficient i sense errors. Aquesta aplicació hauria de permetre superar les limitacions actuals causades per la falta de compatibilitat directa, que comportarà una gestió més eficient i ràpida de la informació. Per fer la prova de concepte d'una manera més específica, per aquest treball s'han escollit dues plataformes: PrestaShop i HiOffice. S'han escollit aquestes dues perquè PrestaShop, avui en dia, és una de les plataformes més utilitzades en el comerç electrònic. D'altra banda, HiOffice és un sistema de gestió al qual he tingut accés a tota la informació, requisit necessari per a poder dur a terme el treball.

Els objectius d'aquest treball són els següents.

- El primer de tots és analitzar el model de dades i el funcionament de les dues plataformes.
- El segon és dissenyar un model d'interoperabilitat, buscant una bona solució per a la transferència de dades entre les plataformes
- El tercer, fer una prova de concepte de la meva proposta que sigui capaç de connectar dades entre plataformes diferents, més específicament entre PrestaShop i HiOffice.

2 Fonaments

En aquesta secció s'introdueixen els coneixements necessaris sobre les tecnologies que tracta el treball.

2.1 PrestaShop

Prestashop és un sistema de gestió de continguts (CMS) lliure i de codi obert i gratuït, que està orientat al comerç electrònic. Aquest està escrit amb PHP i amb suport per MySQL.

La companyia PrestaShop va ser fundada l'any 2007 per Igor Schlumberger i Bruno Leveque, el seu objectiu era revolucionar el comerç electrònic, i van estudiar com a proporcionar l'e-commerce a tothom, des de petites empreses com grans.

2.1.1 Història

La primera versió de l'aplicació es va llençar l'any 2008, en el moment que es llença la versió 1.0, en un sol mes es van aconseguir més de mil descàrregues.

L'any 2009, llencen les següents versions al mercat, la 1.1 i 1.2, aquell any van aconseguir posicionar-se entre les deu primeres mundials en millors plataformes de eCommerce Open Source.

L'any 2010, aconsegueixen guanyar dos premis, el primer premi de codi obert de Packt Publishing i el segon de l'Open Source Award a la "Millor solució de comerç electrònic".

Cada any, que ha anat passant han anat fent noves actualitzacions i afegint millores a la seva plataforma, l'any 2012 cal destacar que van introduir la possibilitat d'una multi botiga, a part d'incorporar noves millores.

Actualment, tenen un total de 150.000 botigues actives, 3.500 mòduls i està disponible a 60 països.

2.1.2 Arquitectura

PrestaShop segueix el patró d'arquitectura de Model – Vista – Controlador (MVC), el que facilita la separació de lògica de negoci, la presentació i la interacció amb l'usuari.

L'arquitectura de PrestaShop està separada per dues seccions lògiques, el Front Office que representa la part pública de la botiga i el Back Office que és on els comerciants gestionen les seves botigues.

Les dues seccions, les podem separar en dues parts diferents: el front-end i el back-end, com a la majoria d'aplicacions web. El front end és la part on bàsicament s'executa el browser.

2.1.2.1 Back-End

El back-end és on s'executa el servidor, en ell hi ha alguns elements que són comuns per les dues seccions lògiques, aquets elements són la base de dades, el Core Business i els mòduls.

Base de Dades: PrestaShop, està orientada a la Base de Dades, de tal manera que totes les dades es guarden allà, tant si són del Front Office com del Back Office, per la plataforma la base de dades és l'única font de veritat.

Core Business: és el conjunt principal de codi que gestiona que PrestaShop sigui PrestaShop, també és conegut com a lògica empresarial, aquest inclou mòduls, controladors, classes d'ajuda i altres entitats semblants. Per ser més concrets, el Core Business representa l'epicentre de PrestaShop, que conté la totalitat del codi essencial pel funcionament de la plataforma.

Tal com s'esmenta, els controladors estan dins del Core Business, aquests generalment generen pàgines HTML, però en alguns casos poden generar JSON o XML. L'estructura d'aquestes pàgines està definida per Themes, els quals transformen les dades proporcionades pel controlador a HTML, i això s'aplica tant pel Front Office com pel Back Office.

PrestaShop proporciona dues interfícies API:

- **BO API:** utilitzat per proporcionar informació a pàgines basades en VueJS.
- **Web Services:** utilitzat per integrar serveis a tercers. El Web Service de PrestaShop utilitza l'arquitectura REST per tal que estigui disponible en la quantitat més gran de plataformes possibles, ja que el protocol HTTP i el format XML/JSON són acceptats per la majoria de les plataformes, per no dir totes.

La diferencia és que mentre el Web Service pot generar dades XML o JSON, la BO API és només per JSON. En el treball es farà ús de la interfície de la API Web Service.

Mòduls: Són paquets independents i opcionals que poden ampliar i personalitzar PrestaShop de diverses maneres. Aquests mòduls interactuen amb el Core, o bé connectant-se a punts d'extensió ubicats en tot el codi, o substituint components principals amb les seves pròpies versions.

La pila de core business

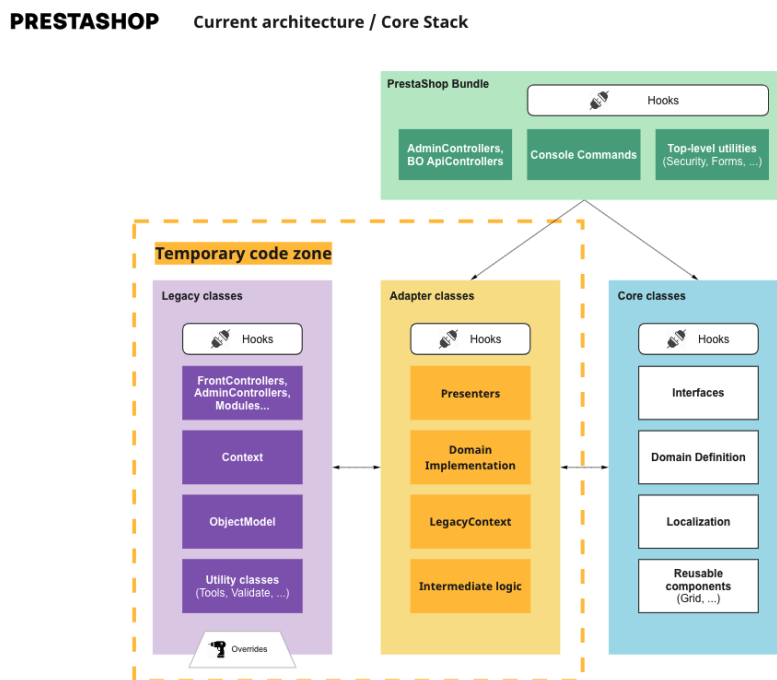
Els controladors seran diferents en el Back-Office i en el Front-Office, però quasi tot el codi PHP de PrestaShop és compartit amb aquestes dues seccions. Aquest codi es divideix en quatre sistemes lògics: Legacy code, Core Code, Adapter code i Symfony code.

El subsistema Legacy inclou el codi antic que no utilitza espais de noms i que ha sigut heretat de versions anteriors de software. Aquest codi gradualment està sent substituït pel subsistema Core, el qual utilitza espais de noms i es basa en els principis SOLID, ja que utilitzar la tècnica d'espais de noms per organitzar el codi, és més eficient i evita conflictes de noms entre diferents parts del programa, per tant, aquest canvi representa una millora a la seva organització.

El subsistema Adapter actua com a pont cap a les classes heretades, que solen ser estàtiques, per poder injectar-les a les classes del nucli.

I per últim, el subsistema Symfony, també anomenat PrestaShop Bundle, és un conjunt de Symfony que conté funcionalitats específiques de Symfony com poden ser controladors, vistes, formularis, etc.

La interacció entre els diferents subsistemes funciona de la següent manera, sempre que el subsistema Core necessita alguna cosa del subsistema Legacy, en comptes d'utilitzar el subsistema Legacy, delega la tasca al Adapter i aquest utilitza el subsistema Legacy.



*Imatge 2.1.2.1.1 La pila empresarial bàsica
Font: PrestaShop*

Controladors

PrestaShop està basada en el Model - Vista - Controlador (MVC), tal com s'ha esmentat, on s'encarreguen de gestionar les sol·licituds i retornar respostes. Els controladors es divideixen en dues grans famílies: aquells que gestionen les sol·licituds en el Front Office i aquells que gestionen en el Back Office.

Poden pertànyer tant el sistema Legacy com el PrestaShop Bundle. Els primers es fa referència com "Legacy Controllers" i els altres com "Symfony Controllers" i aquests només estan presents en el Back Office.

Mòduls

Els mòduls, com ja s'ha esmentat són paquets independents. El sistema de mòduls proporciona complements per afegir funcionalitats, es basa en punts d'extensió anomenats Hooks, que són el principal camí per la integració de mòduls. Els mòduls poden connectar-se a ells per proporcionar o modificar característiques.

Hi ha dos tipus de hooks:

- **Display hooks:** integrats principalment, però no exclusivament, en plantilles. Permeten als mòduls injectar contingut que es mostrarà en algun lloc de la pàgina.
- **Action hooks:** permeten que els mòduls siguin informats d'alguna cosa que succeeix en el sistema i opcionalment alterar el comportament del sistema modificant les dades

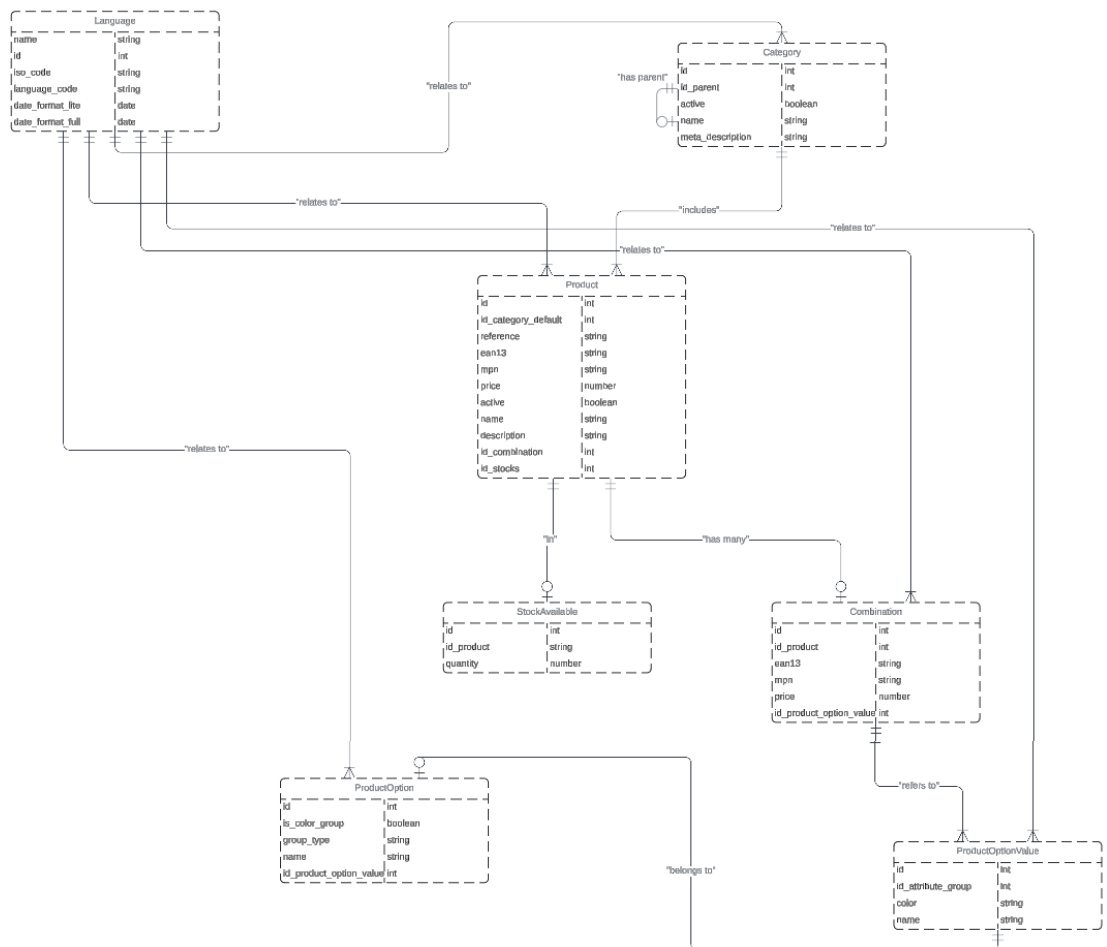
2.1.2.2 Front-End

Les implementacions poden variar molt depenent dels themes, ja que alguns estan basats en HTML i depenen d'una petita quantitat d'script, en canvi, n'hi ha d'altres que estan més avançats i depèn bàsicament de JavaScript.

Com en els controladors, hi ha dos tipus de themes en PrestaShop:

- **Front Office:** els themes defineixen la seva aparença, normalment PrestaShop ve amb un theme predeterminat. Els themes de front office funcionen sobre controladors heretats, aquests estan basats en plantilles Smarty. Tots ells integren una llibreria de JavaScript central que s'anomena *Core.js*, que inclou jQuery 3, una biblioteca de JavaScript que facilita la manipulació d'esdeveniments, la creació d'animacions i interacció amb els servidors a través de AJAX.
- **Back Office:** pel Back Office, hi ha dos themes: el per defecte, i el nou. N'hi ha dos, ja que els Legacy Controllers estan basats en Smarty, i els Symfony Controllers estan basats en Twing, per tant, hi ha un theme per cada un d'ells, els primers utilitzat el theme per defecte i els de Symfony utilitzen els nous. Cal comentar que els controladors de manera progressiva van migrant cap a Symfony, ja que les templates també es mouen del theme per defecte al nou.

2.1.2.3 Estructura de la base de dades



2.2 HiOffice

HiOffice, està englobat dins de les solucions HioPos Solutions, és una companyia que desenvolupa solucions per les empreses i establiments dedicats als sectors de l'hostaleria i retail, HiOffice més en concret és el software de gestió empresarial, que centralitza les dades. HioPos va néixer el 2010, juntament amb A&K Holding, Grup ICG, un grup creat l'any 1985, fundada per Andreu Pi Pocorull, on el seu objectiu principal era fabricar solucions tecnològiques per tal de fer la vida més fàcil a les empreses i usuaris.

2.2.1 Història

El grup ICG, va començar el seu camí amb la fabricació de software propi. Al 1962 es van reinventar i van començar a desenvolupar software de forma industrial.

L'any 1994, aprofitant l'aparició del primer Windows, van desenvolupar el seu primer software visual i tàctil per a l'hostaleria. Al cap d'un any, també van desenvolupar el primer terminal de punt de venda tàctil en el món, gràcies aquesta innovació i al ser els primers a fer-ho els hi va obrir les portes cap a l'expansió internacional de la seva marca i productes.

Al pas dels anys van anar desenvolupant i millorant les seves tecnologies. L'any 2010, decideixen crear HioPos Solutions, on l'objectiu principal era desenvolupar una plataforma cloud pròpia amb tecnologia d'última generació, gràcies a la seva experiència en el sector van tindre a l'abast tots els coneixements i eines necessàries per a respondre a les necessitats del mercat.

HioPos inclou un conjunt d'aplicacions web i serveis, concretament són:

- HioScreen: aplicació de pantalles de cuina.
- SelfQuiosk: aplicació de Kiosk, s'executa en un dispositiu Android.
- Analytics: aplicació web d'estadística
- HiOffice: Aplicació web, solució ERP BackOffice.
- PortalRest: aplicació web que permet als usuaris finals realitzar comandes.
- Bridge d'exportació i importació: serveix per poder realitzar exportacions i importacions de dades del cloud.

2.2.2 Arquitectura

HioPos es basa en una arquitectura de servidors cloud. El cloud de HioPos, s'ha construït a partir dels avantatges del cloud públic i del privat. Aquest cloud utilitza els recursos a nivells de hardware que ofereixen els cloud públics, però garanteix la seguretat i confidencialitat de les dades que ofereixen els clouds privats.

Els avantatges de fer-ho d'aquesta manera són:

- S'utilitzen màquines virtuals amb diferents rols.
- Varis *datacenters*, privats i públics,
- Gran escalabilitat
- Independència de tercers.
- Seguretat i confidencial garantida
- Total flexibilitat.

La seguretat implementada per tal que les comunicacions entre servidors, aplicacions i dispositius siguin segures es fa ús del protocol de seguretat SSL, més concretament TLS.

2.2.2.1 Arquitectura de les solucions cloud

HioPos esta composta per diferents aplicacions, on tota la seva informació es guarda al núvol. La infraestructura cloud de HioPos es divideix en tres grups principals: CloudLicense, CloudClient, Backups.

CloudLicense

És el core de la infraestructura cloud. Totes les funcions de manteniment de la infraestructura es realitzen des d'aquí (afegir nous servidors, manteniment de clients i botigues, configuració de punts de venda, contractacions de productes, etc.) CloudLicense està format per quatre servidors d'aplicacions i dos servidors de base de dades. Tres servidors d'aplicacions i un servidor de la base de dades estan ubicats en les instal·lacions de HioPos, aquests disposen d'accés a internet. Els altres servidors, un d'aplicacions i un de bases dades, estan ubicats en una localització externa.

El servidor de base de dades, ubicat a HioPos, està amb replicació amb el servidor extern a temps real. Tots els servidors de CloudLicense treballen contra la base de dades màster, en el cas d'error, la base de dades replicada passa a ser màster i els servidors d'aplicacions treballarien contra aquesta base de dades, en els dos servidors de base de dades es realitzen backups.

CloudClient

En aquesta part del cloud es troben emmagatzemades les bases de dades dels clients i les diferents aplicacions que es poden executar, entre elles HiOffice.

El servidor CloudClient i el seu replicat, sempre estan emmagatzemats en ubicacions diferents, per tal que quan es trobi amb una intenció greu el servidor CloudClient s'activa el seu replicat perquè sigui ell qui proporcioni el servei, els proporciona als usuaris treballar de forma transparent contra el servidor replicat.

Hi ha diversos models disponibles:

- Model estàndard: en aquest model les aplicacions, serveis i les bases de dades estan ubicades en el mateix servidor. Les bases de dades estaran replicades a temps real contra un altre servidor en una localització externa.
- Model de separació servidor d'aplicacions i dades: en aquest model les aplicacions i serveis estan separats de la base de dades en diferents servidors però el mateix datacenter. Les bases de dades estan replicades a temps real contra un altre servidor extern.
- N servidors d'aplicacions: en aquest model, les aplicacions i serveis estan separats de la base de dades en diferents servidors, existeixen diferents servidors d'aplicacions. Com el model anterior, les bases de dades estan replicades a temps real amb un altre servidor extern.

El servidor de recursos, també anomenat CloudImage, s'encarrega d'emmagatzemar i donar serveis a tots aquells recursos estàtics, com per exemple, les imatges als servidors clients.

Aquest servidor el que permet és centralitzar tots els fitxers estàtics de totes les aplicacions d'un client en un mateix servidor, poder migrar clients entre servidors CloudClient sense la necessitat de migrar tots els recursos estàtics i allibera espai dels servidors CloudClient.

Backups

Aquests servidors estan emmagatzemats en un lloc diferent dels servidors CloudClient, guarda les còpies de seguretat i incrementals de totes les bases de dades dels clients, bases de dades de CloudLicense i bases de dades dels servidors de recursos.

2.2.2.2 Aplicacions web i Android

HioPos, tal com s'ha esmentat compta amb diferents aplicacions que s'executen sobre Android i web, i en aplicacions web es compta en diferents aplicacions de negoci, entre elles HiOffice.

S'utilitza Angular, és un framework open source desenvolupat per Google, per tal de facilitar la creació i programació d'aplicacions web. Utilitza un patró Model – Vista – Controlador (MVC).

2.2.2.3 Estructura de la Base de Dades

Es basa en una base de dades relacional, és aquella que emmagatzema i proporciona accés a punts de dades relacionals entre si. La base de dades es basa en el model relacional.

En la base de dades relacional cada fila de la taula té una clau primària, és un registre que garanteix la unicitat de cada entrada a través de l'assignació d'un identificador únic a cada registre. Per establir relacions entre els registres de diverses taules es fa ús de l'identificador únic esmentat, en aquest context s'anomena clau forana, aquesta permet vincular les dades de manera coherent i eficient entre les taules relacionades.

Les dades, en aquest context, s'organitzen en dimensions o també anomenades entitats, i atributs. Cada una de les dimensions, té un conjunt de registres, amb els seus atributs corresponents, de tal manera que la dimensió és la representació lògica d'una entitat. Les principals dimensions de Hiopos Cloud són els articles, documents, costos, stocks, tresoreria i contactes, aquest últim inclou els clients, proveïdors i venedors. El model es complementa amb les interrelacions entre les diferents dimensions, proporcionant una visió integrada de les diverses entitats i les seves relacions dins del sistema Hiopos Cloud. En aquest treball només es definiran aquelles dimensions amb les quals es treballaran.

Articles

L'esquema està compost per diferents dimensions, per tal d'evitar una exposició excessivament detallada sobre les relacions i dimensions que integren la dimensió d'articles, s'ha optat per descriure els atributs més significatius de les diverses dimensions que la conformen.

El nucli central de l'esquema 2.2.2.4.1 resideix en la dimensió Article. En el moment de la seva creació, es genera un nou codi d'article, aquest actua com a clau primària, mentre que els atributs principals associats aquest article es desen en aquesta entitat.

A continuació es procedeix a explicar els principals atributs i relacions amb l'entitat Article.

Classificació d'Article

En HiOffice existeixen quatre nivells de classificació: Departament (nivell 1), Secció (nivell 2), Categoria (nivell 3) i Subcategoria (nivell 4). Cada un d'aquests nivells únicament pot pertànyer a un dels seus nivells superiors.

Tal com es pot observar en l'esquema 2.2.2.4.1, cada un d'aquests nivells esmentats està dotat de la clau forana dels seus nivells superiors als quals està relacionat, mentre que al mateix temps, cada clau primària dels nivells està vinculada amb la clau forana de l'entitat Article. Cal destacar que un article no necessàriament ha d'incloure tots els nivells, ni pot no tenir cap nivell assignat.

Tipus d'article intern

En el cas de retail, un article pot tindre talles, colors, utilitzar stocks, etc. Aquesta informació es guarda dins l'entitat Tipus Article Intern, el valor de l'atribut Tipus Article Intern és predeterminat pel mateix programa el tipus d'article intern que serà, en el cas d'aquest treball sempre serà talles i colors. L'entitat es relaciona amb l'entitat Article a través de la clau primària Codi Tipus Article Intern.

Subarticle

La dimensió Subarticle conte els diferents formats d'un article. Aquesta dimensió es relaciona amb la dimensió Article mitjançant la clau forana Codi Article.

Codi de Barres

El codi de barres es guardarà en l'entitat Subarticle, tot i que, és important destacar que un mateix article pot tindre diversos codis de barres. Per aquest motiu, s'ha implementat l'entitat Codi de Barres, la qual emmagatzemarà aquests codis addicionals.

Preus i tarifes

Existeix la particularitat que un mateix article pot disposar de diversos preus, motiu pel qual es defineixen les tarifes de venda. A un sol article se li poden associar diverses tarifes, cadascuna amb el seu preu corresponent. L'associació d'un subarticle amb la seva tarifa corresponent es duu a terme a través de les entitats Preu de Venda. Aquestes entitats inclouen la clau primària del subarticle, i dins d'elles es troben la tarifa i el seu preu respectiu.

Estoc i magatzem

En cas que un article disposi d'estoc, les dades relacionades amb aquest es guardaran en l'entitat estoc, establint-se una relació amb el magatzem corresponent. A més a més, l'entitat Magatzem, es relaciona, a la seva vegada, amb l'establiment corresponent.

L'estructura presentada fins ara representada és d'un article bàsic, però en el context del sector retail existeixen diverses tipologies d'articles amb estructures més complexes. En aquests negocis, les necessitats específiques requereixen la definició dels articles amb una o més característiques o paràmetres, en els quals poden combinar-se entre si per distingir un mateix producte. Aquestes característiques fan referència a la talla i color d'un article, la combinació dels quals genera un article únic.

El programa de HiOffice permet assignar un grup de talles i colors a l'article.

La creació d'aquests tipus d'article segueixi la mateixa estructura abans esmentada en el procés de creació de l'article en si. Es coneix que un article té talles i colors gràcies a l'entitat Tipus d'Article intern, explicada anteriorment. Un cop l'usuari selecciona que el tipus d'article intern serà talles i colors automàticament es determina en l'entitat Article els atributs "Utilitza dimensió u" i "Utilitza dimensió dos", indicant que utilitzen Talla i utilitzen Color.

Dimensió – Valors – Grup

Un cop el sistema reconeix que el Tipus d'Article Intern correspon a talles i colors, s'involucren quatre dimensions diferents. La primera és Dimensió, la qual conte el Codi Dimensió, un per la Talla i un pel Color. La segona dimensió és Dimensió Valors, la qual inclou la clau forana de Codi Dimensió i estableix els valors associats a Talles i els Colors amb el seu corresponent Codi Dimensió Valor. A més a més, aquests atributs es relacionen amb l'entitat Subarticle mitjançant la clau forana Codi Dimensió Valor. La tercera dimensió, anomenada Dimensió – Grup, estableix una relació entre cada dimensió de l'entitat Dimensió amb un grup de dimensions. Aquest grup de dimensions, referit com a grup de talles i grup de colors, i requereixen una assignació d'un grup en el mateix programa. Cada un dels grups té el seu Codi Grup Dimensió corresponent. Cadascun dels grups disposa d'un Codi Grup Dimensió corresponent, els quals es relacionen amb l'entitat Article com a clau forana. Així, el grup es vincula amb l'article, mentre que els valors de cada grup s'associen amb el Subarticle. La quarta entitat, Dimensió – Valor – Grup, estableix simplement una relació entre Dimensió – Grup i Dimensió Valors, amb l'objectiu de definir talles i colors dels articles. A través d'aquesta relació, s'estableix que les talles de l'article seran les definides a Dimensió Valors, així com els colors dels articles seran els definits en aquesta dimensió.

Subarticle

Dins d'aquesta dimensió, es registrarà cada article al qual ja s'hagi assignat el valor del grup de la talla i el valor del grup del color.

Preus i estoc

L'article, té un preu de venda, i aquest valor es guarda en la dimensió Preu Venta, que a la mateixa vegada està relacionat amb la dimensió Tarifa Venta. El preu de compra de l'article, es guardarà en les dimensions Preu de Compra i Tarifa de compra.

Documents

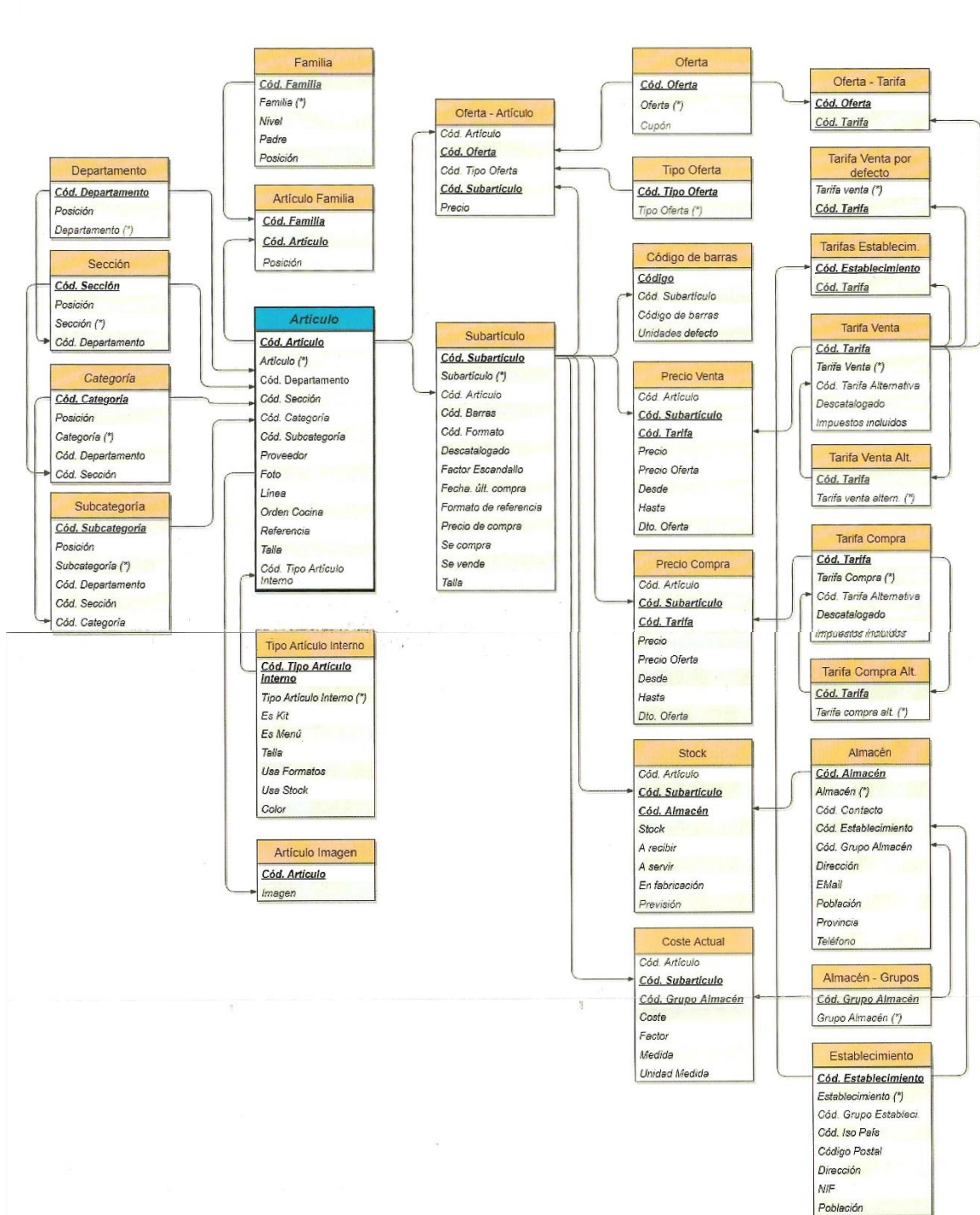
Un document representa el justificant d'una operació comercial que utilitzen les empreses per a realitzar transaccions en les seves operacions mercantils. Els documents mercantils no negociables serveixen a l'empresa per demostrar la realitat d'una operació davant d'un tercer. HiOffice trasllada el document en paper en digital. Els documents que es poden extreure de HiOffice es formen a partir d'un Dashboard i d'on aquests Dashboards també es pot crear un totalment nou mostrant la informació que es desitja.

Dashboard

El dashboard està format per un conjunt de blocs que estan relacionats entre si per tal de formar una pantalla amb informació que tingui sentit propi. Aquest es capaç de mostrar documents de diversos tipus, no únicament factures o justificants, per exemple, poden ser la font de dades per exportar una informació en concret.

Els blocs són estructures que mostren informació provinent d'un origen de dades, es poden enllaçar els blocs entre si per tal de definir una dependència. L'origen de dades del dashboard, ens ofereixen informació emmagatzemada en la base de dades, s'obté de les dimensions esmentades anteriorment, que ja estan relacionades i tenen un sentit propi les dades. Les dimensions ens permeten accedir a la informació de la base de dades. Una dimensió és un conjunt d'atributs que representen una entitat.

2.2.2.4 Esquema de la Base de Dades



2.3 Data Bridge

Data Bridge serà una aplicació que es desenvoluparà per tal de solucionar i facilitar la interoperabilitat entre diferents aplicacions. L'objectiu principal de l'aplicació serà oferir integracions amb diferents sistemes i aplicacions, amb l'objectiu de crear processos de manera automàtica i estalviar temps als usuaris.

L'aplicació principalment està orientada cap el sector empresarial en comptes de l'usuari en particular. La seva funció principal consisteix en la importació, transformació i exportació de dades des d'un servei A cap un servei B i viceversa, Data Bridge s'encarregarà de tractar les dades. Se li dona aquest enfocament a causa que s'ha detectat que les empreses de manera habitual, requereixen solucions especialitzades per gestionar i transferir les dades de manera eficient i segura entre les diferents plataformes. Data Bridge serà una eina indispensable per optimitzar processos interns i millorar la interoperabilitat entre els serveis i garantir la integritat i coherència de la informació en entorns empresarials dinàmics.

En aquest treball en concret, es centrarà en dos serveis que utilitzen el sector de la venda al detall o també anomenat el sector Retail, PrestaShop i HiOffice, ja explicats i anomenats anteriorment.

Estarà desenvolupada de tal manera que per configurar l'aplicació serà necessari tindre alguns coneixements informàtics perquè per la seva configuració és necessari omplir certs camps que requereixen aquets coneixements. Aquets camps són:

- **Camps de configuració extra**, per les diferents funcionalitats serà necessària informació extra per poder executar-les. En el cas del servei implementat en aquest treball, en aquest camp extra serà necessari per exemple les API Keys de les dues plataformes, per tal de poder establir connexió entre les diferents API's.
- **Mappings**, és un camp que permet mapejar les dades de la plataforma A amb la B, i a la inversa. De tal manera que gràcies aquest camp sabem quina és l'equivalència de cada camp entre les dues plataformes

En la base de dades de la plataforma Data Bridge, trobem els diferents clients que estan subscrits a la plataforma, dins de cada un d'ells els serveis que tenen contractats, amb les diferents plataformes amb les quals volen treballar per importar i exportar les seves dades. I les dades necessàries per executar aquestes funcionalitats.

Dins d'aquesta Base de Dades també trobem les relacions de cada servei creat a la plataforma amb la funcionalitat a la qual pertany, i els seus respectius mappings.

2.4 Interoperabilitat

La interoperabilitat, de manera general, es refereix a la capacitat dels sistemes per intercanviar informació i utilitzar la informació que s'intercanvia. En el context empresarial, fa referència a la "capacitat d'interacció entre empreses". La interoperabilitat es pot donar a terme entre diferents nivells de les empreses que interactuen i els seus sistemes empresarials relacionats. Quan la interoperabilitat no s'aconsegueix, es converteix en un problema, i aquests problemes poden influir en el rendiment i els resultats de les xarxes de les empreses (da Silva Serapião Leal, Guédria, & Panetto, 2020).

Per tal d'evitar aquests problemes, les empreses han de saber a nivell d'interoperabilitat quines són les seves fortaleses i les seves debilitats. Les empreses es poden beneficiar de l'Avaluació Interoperabilitat (INAS), on aquesta ajuda a identificar els principals problemes d'interoperabilitat i les seves possibles solucions.

En la literatura de la interoperabilitat s'han proposat diferents marcs per tal de descriure-la.

La European Interoperability Framework (EIF), considera quatre nivells principals d'interoperabilitat:

- **La capa semàntica:** aquesta engloba la sintaxi de la informació, és a dir, que el format de les dades a transmetre s'entengui i la semàntica de la informació, el seu significat.
- **La capa tècnica:** s'encarrega de les aplicacions i les infraestructures relacionades amb els sistemes i serveis. Aquesta inclou especificacions d'interfícies, serveis d'interconnexió, serveis d'integració de dades, presentació i intercanvi de dades i protocols de comunicació.
- **La capa organitzacional:** fa referència a la manera en què els sistemes alineen els seus processos, responsabilitats i expectatives per arribar als objectius marcats mútuament.
- **La capa legal:** es refereix als problemes legislatius relacionats amb l'alineament de funcions empresarials superiors o polítiques governamentals, expressades normalment en forma d'elements legals i regles comercials.

El Framework for Enterprise Interoperability (FEI) defineix tres barreres de la interoperabilitat:

- **Barreres conceptuals:** fa referència a la modelització en un nivell d'abstracció alta, és a dir, als obstacles que poden sorgir a causa de les diferències en que es modelen els conceptes i processos, com els models d'una empresa.
- **Barreres tecnològiques:** és la falta d'un conjunt estàndard compatible que permetin l'ús de tècniques informàtiques per a compartir i intercanviar la informació.
- **Barreres organitzatives:** són aquelles incompatibilitats de l'estructura organitzativa, les normes del negoci i les tècniques de gestió que s'implementen en les dues empreses.

Un altre punt a esmentar, és les diverses perspectives, és a dir àmbits o àrees, on es pot aplicar l'intercanvi o interacció de la informació. Les que proposa la FEI en son quatre:

- **Preocupació empresarial:** es refereix a treballar de manera coordinada en els diferents nivells dins de l'organització.
- **Preocupació pels processos:** l'objectiu és fer que diversos processos funcionin de manera conjunta.
- **Preocupació pels serveis:** el propòsit principal és identificar, compondre i operar conjuntament amb diverses aplicacions.
- **Preocupació per les dades:** trobar i compartir informació de diverses bases de dades com a fonts, que poden residir en diferents dispositius, sistemes operatius i sistemes de gestió de bases de dades

Per tal que les empreses puguin evitar els problemes d'interoperabilitat és necessari que la interoperabilitat entre els diferents sistemes empresarials millori de manera continuada. Un dels principals objectius del INAS és que les organitzacions siguin conscients de la seva situació, per tal de millorar la seva capacitat.

2.4.1 Tipus d'avaluació

Hi ha tres tipus d'interoperabilitat:

- **L'avaluació de la potència:** és aquella que avalua la interoperabilitat d'un sistema cap el seu entorn. L'objectiu principal és mesurar la capacitat d'un sistema per adaptar-se.
- **L'avaluació de la compatibilitat:** avalua la interoperabilitat entre dos sistemes, el més crucial és analitzar l'estat actual de tots dos sistemes per poder identificar els conflictes que poden crear problemes i els que no.
- **L'avaluació del rendiment:** avalua les interaccions en el temps d'execució, considera els costos provocats per la implementació de les aplicacions, la duració entre el moment en què es sol·licita la informació i el moment en el qual s'utilitza la informació sol·licitada, la qualitat de l'intercanvi, la qualitat d'ús, i la qualitat de conformitat.

2.4.2 Tipus d'interoperabilitat

La interoperabilitat es pot aplicar a diferents nivells:

- **Interoperabilitat tècnica:** és aquella que a través de llenguatges i eines permeten l'intercanvi telemàtic de bytes. Permet la interconnexió a nivell de comunicació de protocols i intercanvi d'informació entre equips informàtics.
- **Interoperabilitat sintàctica:** fa referència a la transmissió de documents/ missatges entre sistemes. Permet l'intercanvi d'informació fent ús d'un format estandarditzat.

- **Interoperabilitat semàntica:** manté el significat del que es vol traduir entre sistemes i organitzacions. Permet l'intercanvi d'informació, a través d'un vocabulari comú i compartit, per tal d'evitar inexactituds en la interpretació del significat dels continguts que es comparteixen. De tal manera que pertany a les metadades, els models de representació de la informació i vocabularis. (Berners et al., 2001).
- **Interoperabilitat organitzativa:** reflecteix les regles de negoci, és la capacitat que tenen dues organitzacions per tal de col·laborar, compartir informació i processos d'una manera efectiva.

2.4.3 Mecanismes de mesura

Hi ha dos tipus principals:

- **Mesures qualitatives:** són principalment subjectives. Normalment avalua a través d'una escala de variables lingüístiques, com per exemple Òptim, per qualificar el sistema. S'utilitza majoritàriament en els models de maduresa, que estan dissenyats per avalua la qualitat d'un domini concret basant-se en un conjunt de criteris mes o menys exhaustius.
- **Mesures quantitatives:** avaluen a partir de nombres numèrics per caracteritzar la interoperabilitat. En general, l'escala de qualificació va del 0 al 100%. De manera habitual s'apliquen en les avaluacions compatibilitat i rendiment (da Silva Serapião Leal, Guédria, & Panetto, 2020).

3 Disseny

Ja que per fer el disseny és necessari centrar-se en plataformes existents, en aquest apartat es suposarà que la plataforma A correspon a HiOffice i la plataforma B fa referència a PrestaShop

3.1 Funcionalitats

L'aplicació consta de dos grans mòduls. El primer mòdul fa referència a les dades que transformem de la plataforma A, per introduir a la plataforma B. El segon, és aquell que transformem les dades de la plataforma B, per introduir a la A.

En el primer mòdul hi ha cinc grans funcionalitats, aquest mòdul és una sola funció que conté les següents funcionalitats:

- **Creació d'atributs:** En el sector retail, hi ha el concepte anomenat combinacions, són aquells productes que tenen més d'una variació. Tant poden ser de colors, talles, o altres atributs. Aquesta funcionalitat ofereix el servei d'obtenir tots els atributs que ofereix la plataforma de HiOffice i importar-los a la plataforma de Prestashop. Aquesta funcionalitat es capaç tant de crear els grups d'atributs, talla i color, per exemple, com els atributs de cada un dels grups. En aquesta funcionalitat, inclou la funció de fer un Update a tot aquells atributs que ja existeixen a la plataforma de PrestaShop, per tal d'assegurar-se que l'usuari sempre tingui les dades actualitzades.

En la plataforma Prestashop aquests atributs es guarden amb un *id* concret, aquest *id*, no es pot gestionar exteriorment, a la plataforma de Prestashop s'assigna de manera incremental internament sense possibilitat de ser manipulat.

- **Creació de categories:** Per tal de classificar els productes relacionats entre si que s'organitzen segons la seva naturalesa, les seves característiques o ús, s'utilitza el concepte de categories. La funcionalitat obté totes les categories creades a HiOffice i les subcategories de cada una d'elles, i transformar-les perquè la plataforma de PrestaShop les reconegui i poder-les importar en aquesta plataforma. Tal com els atributs, aquestes categories s'identifiquen per un *id* únic sent no manipulable de manera externa.
- **Creació i actualització de productes:** Aquesta funcionalitat s'encarrega de carregar tota la informació dels productes simples, els productes simples són aquells que no existeix cap combinació d'aquell producte, existents a HiOffice i importar-los a PrestaShop. Els camps importants són:
 - Nom de l'article
 - Referència.
 - Codi de barres, anomenat ean13.
 - Categoria
 - Subcategoria
 - Marca del producte
 - Preu

Aquesta funcionalitat ofereix un altre funcionalitat dins d'ella. En el cas que el producte ja estigui creat a la plataforma de PrestaShop, actualitzarà les dades del producte, per assegurar-se de què l'usuari tingui les dades correctes sempre, a part d'actualitzar totes les característiques mencionades prèviament, també actualitzarà l'stock del producte en cas que sigui diferent, o crearà l'stock associat en aquell producte, en cas que es creï per primer cop el producte.

- **Creació i actualització de les combinacions:** Les combinacions, esmentades, anteriorment, són les diferents variants que pot tindre un producte. Aquesta funcionalitat s'encarregarà d'exportar els productes de HiOffice i crear aquells productes que posseeix combinacions i crear les seves respectives combinacions a PrestaShop.

Les característiques que es crearan a cada combinació del producte, són les mateixes esmentades anteriorment per la creació del producte. Cal destacar que en aquesta funcionalitat, crearà el producte principal que té combinacions i les seves combinacions. Aquesta funcionalitat, també ofereix la funcionalitat que en el cas, que el producte o les seves combinacions estiguin creades es farà una actualització de les seves característiques. A més d'actualitzar les seves característiques també es farà una actualització o creació de l'Stock en cas que sigui diferent entre les dues plataformes.

- **Creació d'idiomes:** aquesta funcionalitat ofereix el servei d'exportar tots els idiomes que es poden trobar a la plataforma A i importar-los a la plataforma B. En aquest cas s'importen les característiques de l'idioma que són:
 - Nom de l'idioma
 - Codi ISO
 - Codi de la llengua.

Aquesta funcionalitat s'ha creat, ja que és necessària per poder executar la funció del segon bloc, que es anomenada a continuació.

Per tal de poder incorporar categories o combinacions en un producte, és imprescindible que estiguin prèviament creades a la plataforma de PrestaShop. Un cop introduïdes, mitjançant el seu identificador, s'estableix la relació indicant que aquest producte correspon a una categoria específica o que disposa d'unes combinacions determinades. S'ha desenvolupat una taula de relacions a la plataforma de HiOffice, la qual assenyala la interoperabilitat semàntica. En les funcionalitats de creació d'atributs, categories i idiomes, s'efectua una importació a la plataforma de HiOffice en aquesta taula de relacions, indicant l'identificador creat a PrestaShop que es correspon amb HiOffice.

Per verificar l'existència dels atributs o categories, es procedeix a partir d'aquesta taula de relacions. Es poden donar tres situacions diferents: en el cas que el camp de l'identificador de PrestaShop estigui buit significarà que la relació no ha estat establerta mai i es crearà de nou. La segona situació es dona quan el camp no està buit, però l'identificador que figura a la taula ja no es troba a PrestaShop, i per tant es considera com un atribut o categoria nova. Finalment, si el camp no està buit i l'identificador es troba a PrestaShop, es realitza una actualització de la descripció de les categories o atributs.

Respecte a la verificació de l'existència dels productes o combinacions, s'efectua a partir del codi de l'article o combinació de HiOffice. Aquesta elecció es deu al fet que, en crear un producte nou a PrestaShop, s'importa el codi de l'article o combinació a PrestaShop en un camp buit. Aquesta metodologia s'ha adoptat atès que no existeix cap característica única en els articles o combinacions.

En el segon mòdul, es troba una funcionalitat:

- **Creació de la descripció:** En el sector ecommerce, hi ha la descripció dels diferents productes que ofereix la botiga. En aquesta funcionalitat, a causa que la plataforma B, té una gran capacitat per oferir aquest servei, s'exporta des d'aquesta plataforma i s'importa a la plataforma A. Per a cada una de les descripcions existents dels productes de la plataforma B, amb els seus respectius idiomes s'importen a la plataforma A, associant la descripció en cada un dels productes amb el seu idioma.

3.2 Dades d'entrada i sortida

Totes les funcionalitats esmentades, com a dada bàsica, necessiten connexió a les dues API's amb les que es treballa, a l'API de la plataforma A i a l'API de la plataforma B.

Del primer mòdul, les dues primeres funcionalitats, creació d'atributs i creació de categories, les seves dades necessàries per la seva funcionalitat i les seves dades d'entrada de sortida són equivalents, l'única diferència és els recursos a l'hora de realitzar el GET a la plataforma A, i el POST a la plataforma B són diferents.

En aquestes dues funcionalitats, a part de l'esmentada, no és necessària cap dada més per les seves funcionalitats. A partir de la connexió amb l'API de la plataforma A, s'exporta un fitxer JSON personalitzat que serà l'entrada de les dades, aquest JSON es fa a partir d'un dashboard personalitzat, i aquest dashboard personalitzat s'alimenta de la base de dades del client. Tenim un dashboard per cada una de les funcionalitats, un d'ells ens mostra tots els atributs creats en la plataforma i l'altre totes les seves categories, i les subcategories que trobem dins de cada categoria. En la plataforma B, es fa un GET dels respectius recursos, atributs i categories, i aquest ens retorna un fitxer XML, que serà l'altra dada d'entrada de la funció, el qual es converteix per poder tractar les dades. Finalment, un cop s'han tractat les dades, s'esbrina aquells que s'han de crear o fer UPDATE a la plataforma, en el cas que s'hagi de crear es construirà un fitxer un XML on farà un POST a l'Api de la plataforma B, d'altra banda si s'ha d'actualitzar, es crearà un altre tipus de fitxer XML que farà un UPDATE a l'API.

En les funcionalitats de la creació de producte i creació de combinacions, les dades necessàries per a la seva execució, igual que les seves dades d'entrada i sortida són les mateixes.

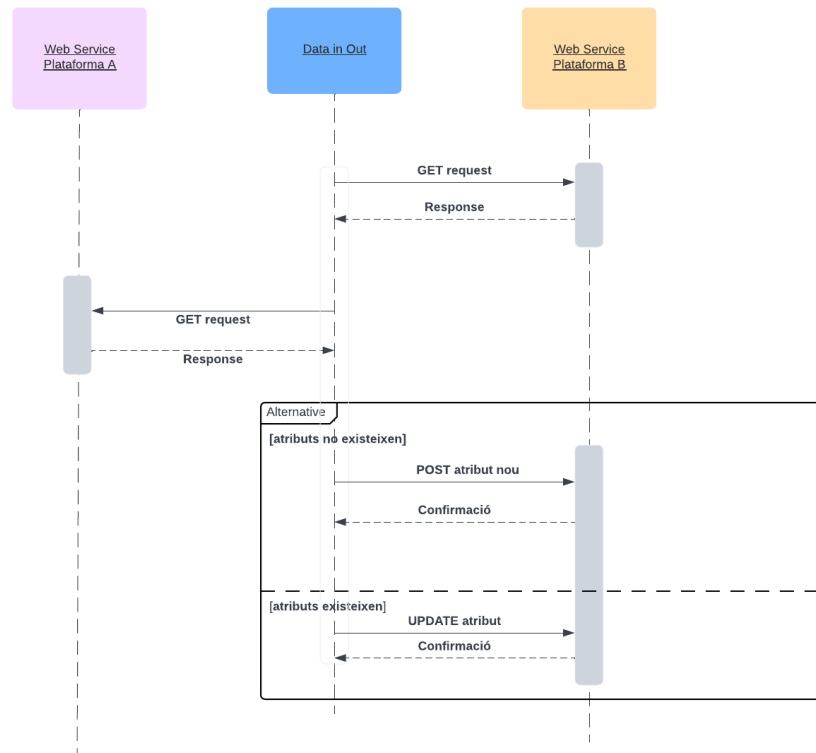
A part de la connexió amb les dues API's, també es necessari el mapping creat en la plataforma de Data Bridge. Aquest mapping s'aconsegueixen fent un GET a la Base de Dades de l'aplicació de Data Bridge, també es fa el respectiu GET a l'API de la plataforma A per tal de saber tots els productes i combinacions que es tenen creats, l'API retorna un fitxer JSON, creat a partir d'un dashboard que s'alimenta de la Base de Dades de la plataforma A.

Aquesta crida a l'API serveix per a les dues funcionalitats, ja que el fitxer JSON ens dona tots els articles i en cada article té un subarray amb les seves respectives combinacions. Es fa un GET a l'API de la plataforma B per obtenir els productes i les seves combinacions en el cas d'aquesta plataforma, es faran dues crides, una per cada un dels recursos. A més, també es fa un GET per obtenir l'stock dels productes i les combinacions. Un cop es fan les comparacions tant d'una funcionalitat com de l'altre, es construeixen els diferents fitxers XML per importar les dades a la plataforma B, en ella es farà un POST o UPDATE . En la funcionalitat de la creació de productes, es podran arribar a crear dos fitxers XML diferents: un XML per la creació de productes, i un altre per l'actualització de les dades del producte. Per altra banda, en la funcionalitat de creació de combinacions s'arribaran a crear quatre fitxers XML diferents: un serà per la creació del producte, un altre XML per la creació de les combinacions d'aquest producte, un altre XML per actualitzar les dades del producte, i un altre XML per l'actualització de les combinacions del producte. A més per tal de crear o actualitzar l'stock del producte i combinacions, es crearà un altre fitxer, ja que la plataforma B, el considera un altre recurs. En aquest cas, el de l'stock, es crearan dos fitxers XML diferents, un per actualitzar l'stock dels productes senzills, i un altre XML per les combinacions del producte.

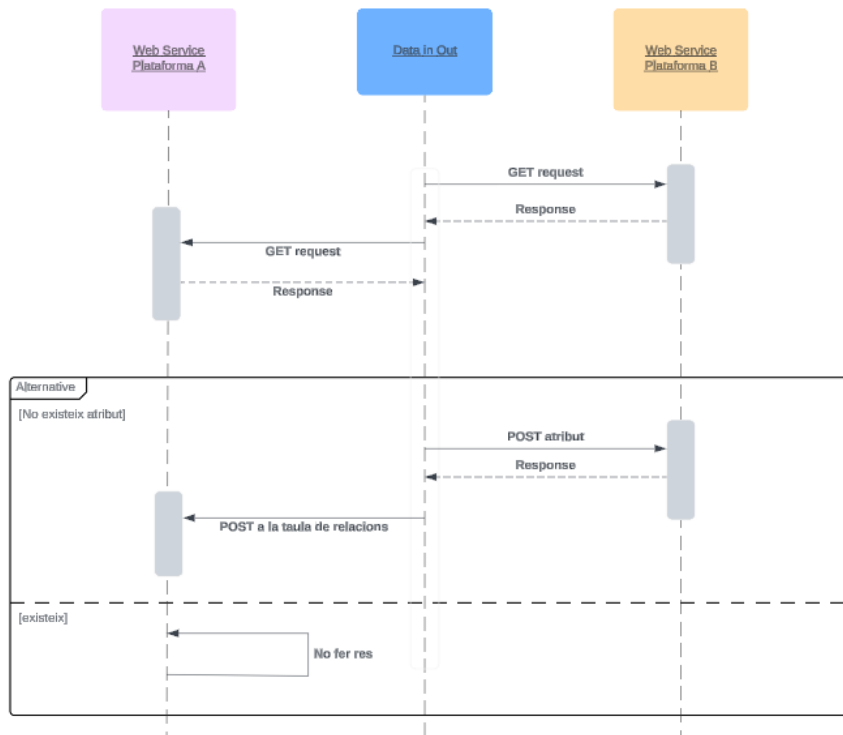
En el segon mòdul, la funcionalitat de la creació de la descripció, necessita, prèviament, l'execució de la funcionalitat de creació d'idiomes. Si no es fa així, les diferents descripcions no es podran relacionar amb els seus idiomes corresponents i l'exportació no es durà a terme de manera correcta. Com a dades d'entrada, es realitza un GET a l'API de la plataforma A, la qual retorna un fitxer JSON que conté els codis dels articles, la descripció de cada article amb el seu idioma i, finalment, la relació entre l'*id* de la plataforma A i l'*id* de la plataforma B. També es realitza un GET a l'API de la plataforma B, la qual retorna un fitxer JSON amb el *mpn*, element necessari per identificar els articles, i les seves descripcions respectives de cada article. Un cop s'han tractat les dades, es crea un fitxer JSON que s'importa a la plataforma A.

3.3 Diagrama de flux de dades

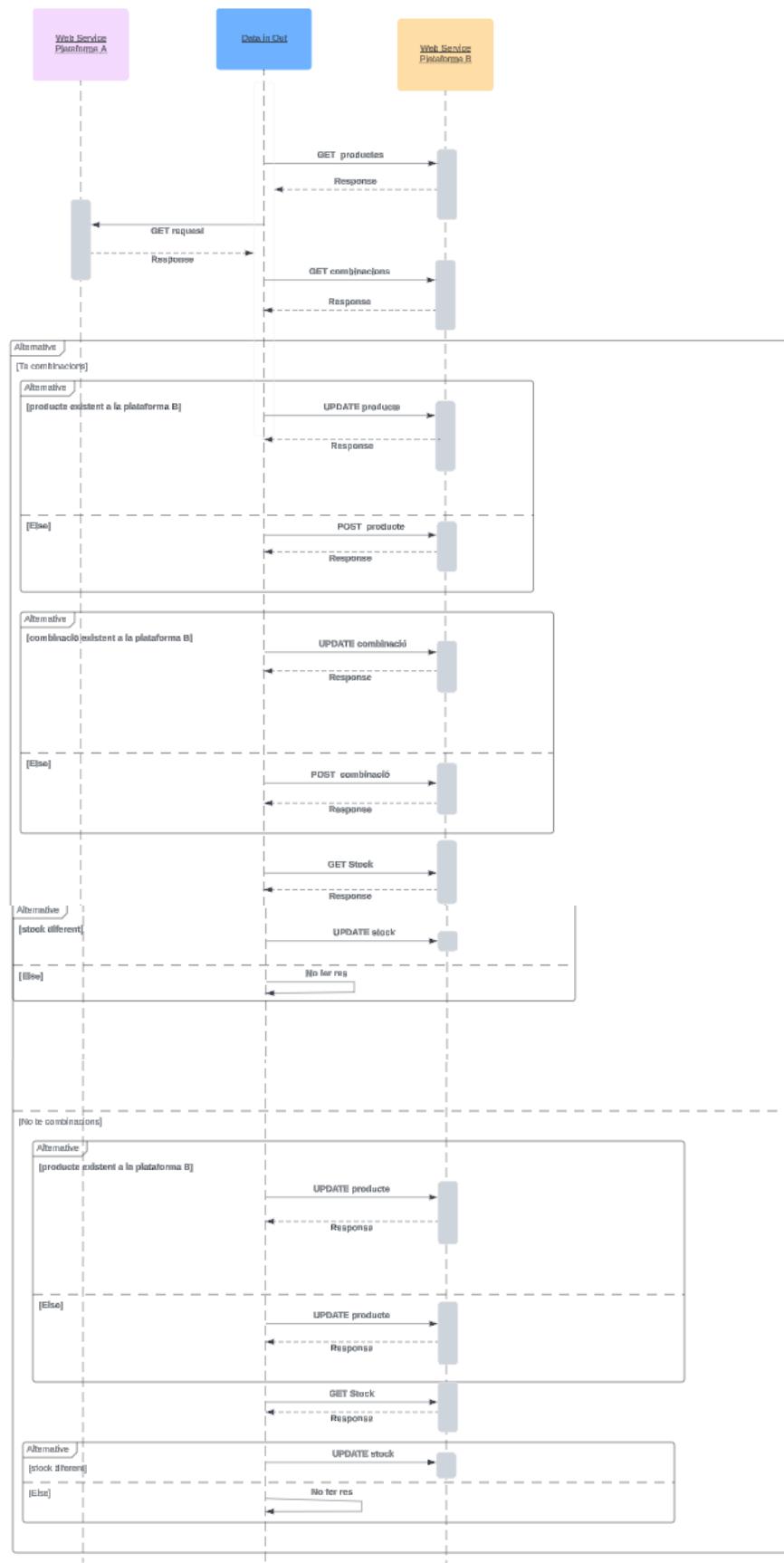
3.3.1 Creació d'atributs



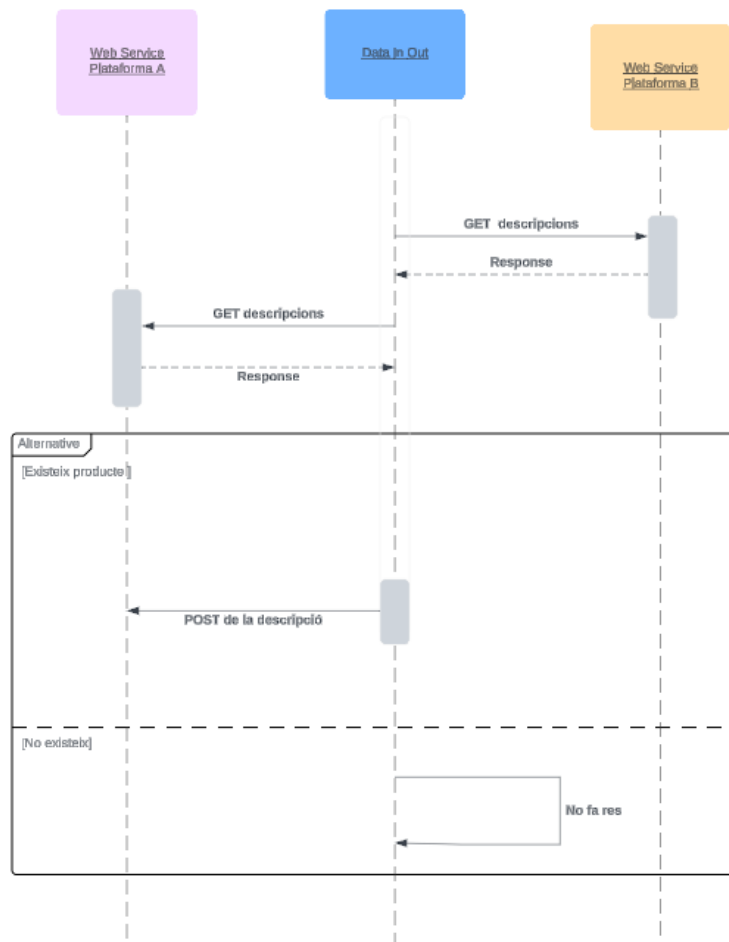
3.3.2 Creació de categories



3.3.3 Creació i actualització de Productes i Combinacions



3.3.4 Creació Descripció



4 Implementació

En aquesta secció s'expliquen detalls relacionats amb la implementació.

4.1 Llenguatge i plataforma

4.1.1 Laravel

La plataforma escollida per tal de desenvolupar l'aplicació és Laravel. Laravel és un framework de PHP de manera gratuïta i de codi obert, va ser creat per Taylor Otwell. Segueix el patró arquitectònic Model – Vista- Controlador (MVC) i està destinat al desenvolupament d'aplicacions web i mòbil.

Laravel inclou diferents eines que faciliten la construcció d'aplicacions web, fent el procés més ràpid i com a resultat dona un codi ben estructurat i fàcil de mantindre. Facilita tasques comunes utilitzades en el desenvolupament de projectes web, proporciona autenticació pròpia, encaminament, gestor de sessions, emmagatzematge en caixet, contenidor d'injecció de dependències i una gran quantitat de components més comunament utilitzats. Fa ús de Eloquent, un ORM (Object- Relational Mapping) que simplifica la interacció amb la base de dades. Permet definir models i realitzar consultes de manera intuïtiva sense tindre la necessitat d'escriure consultes SQL directament.

Funciona des de la línia d'ordres amb Artisan, nom que se li atorga a la interfície per comandes per executar funcionalitats.

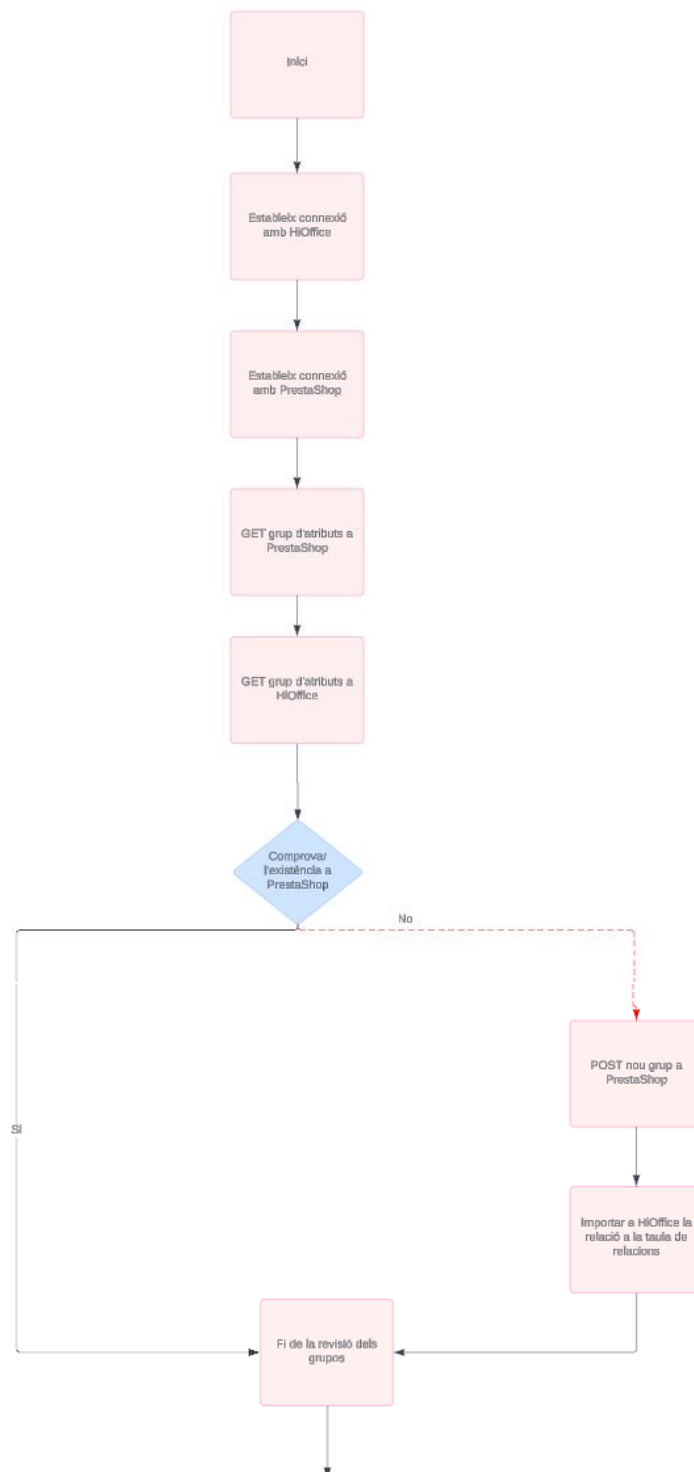
Els serveis principals que ofereix són els següents:

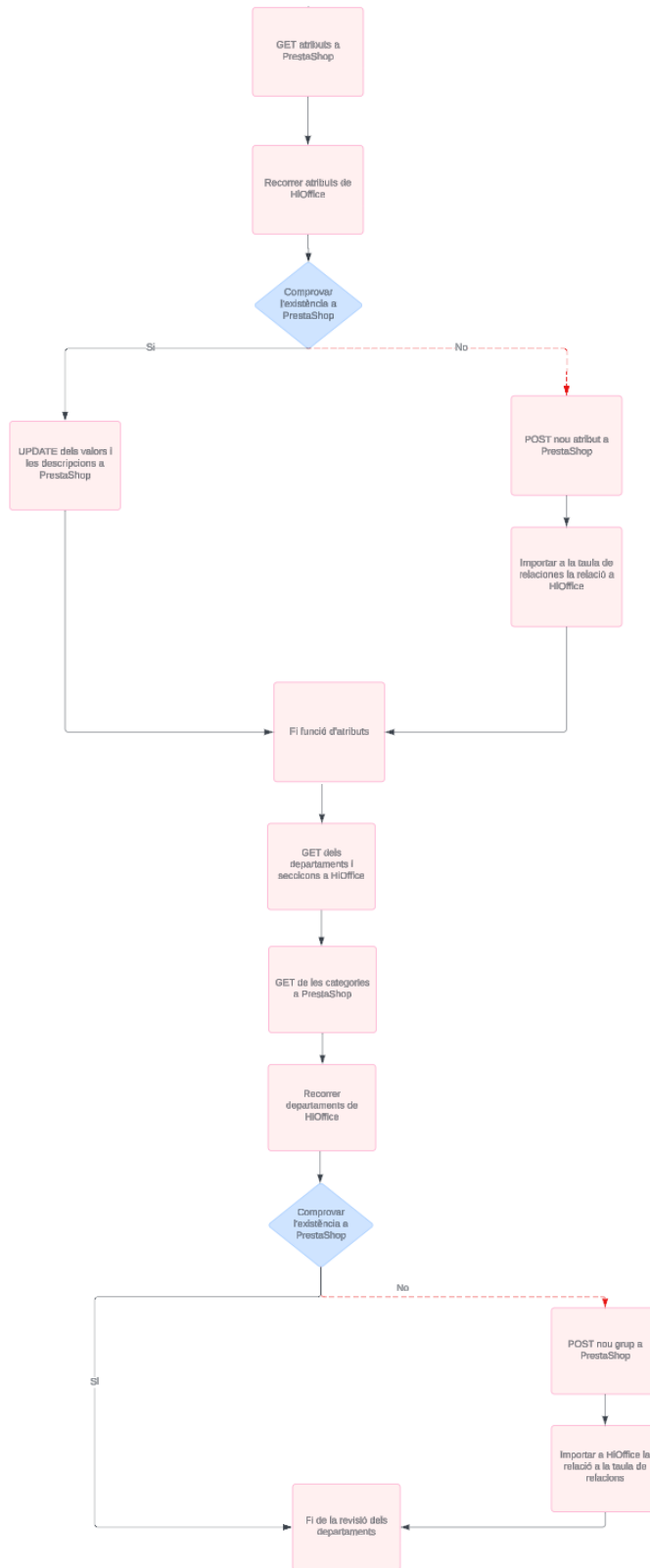
- **Arquitectura de carpetes avançades:** utilitza una estructura de carpetes definida on promou la separació d'arxius i un ordre correcte en el desenvolupament de projectes.
- **Entorns de desenvolupament flexibles:** ofereix una gran quantitat d'opcions per crear entorns de desenvolupament. Facilitat als desenvolupadors a trobar una alternativa que s'adapti a les seves necessitats o preferències.
- **Seguretat:** Laravel ofereix un alt nivell de seguretat, amb mecanismes de hash i salt per encriptar les dades.
- **Integració amb altres serveis:** és capaç d'integrar-se amb altres serveis.

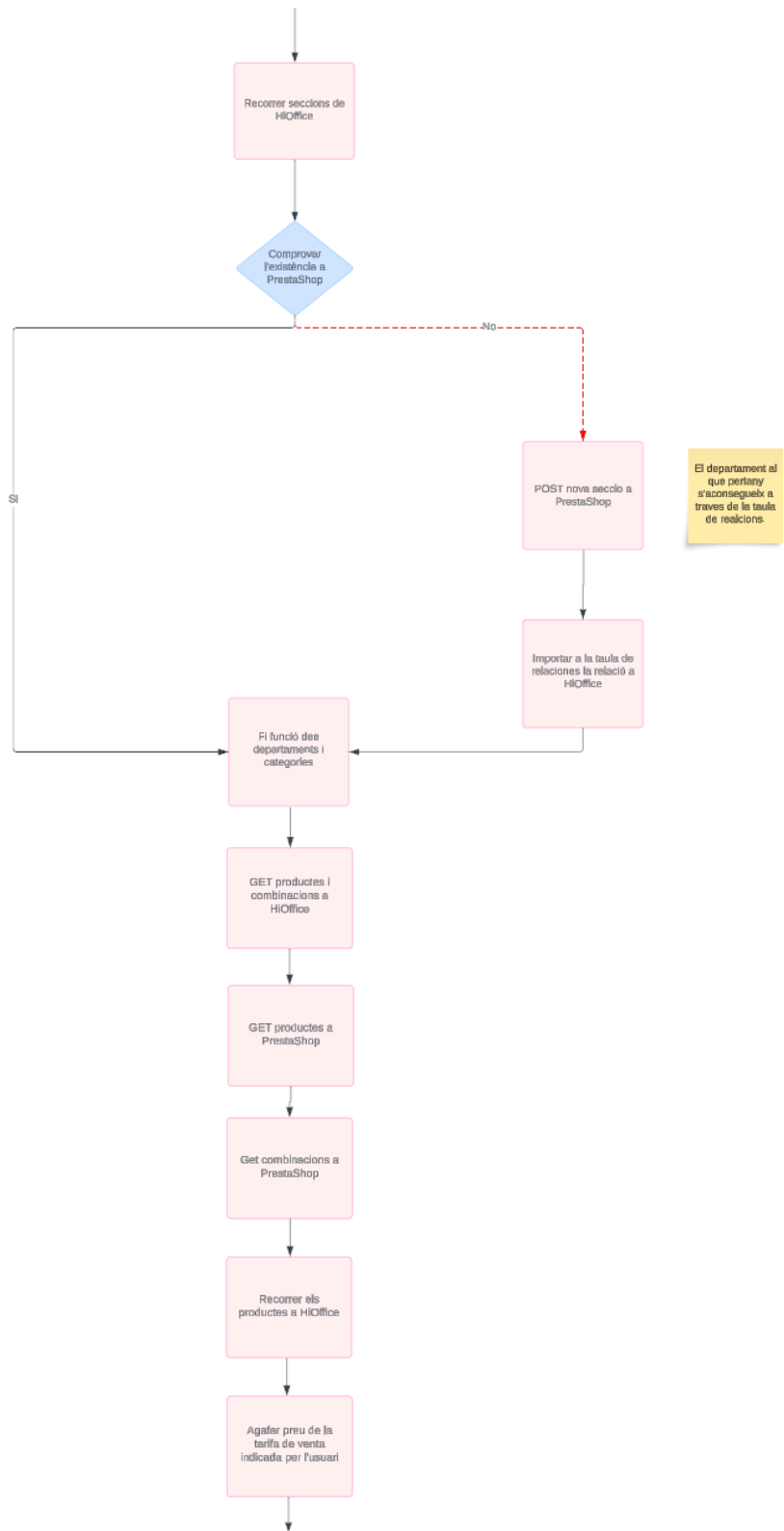
Per al desenvolupament de l'aplicació, s'ha considerat que el framework més adequat seria el de Laravel, per totes les funcionalitats esmentades. Per la seva eficiència en el desenvolupament, ja que és elegant en la seva sintaxi, per la seguretat que ofereix per tal de protegir la informació que es transfereix d'una plataforma a l'altre. A més, a causa de la facilitat que té en integrar altres serveis, es beneficiós per connectar-se entre les dues plataformes de manera eficient i sense causar cap inconvenient. Per últim, la seva escalabilitat fa que més fàcil la gestió d'un gran volum de dades i el tràfic.

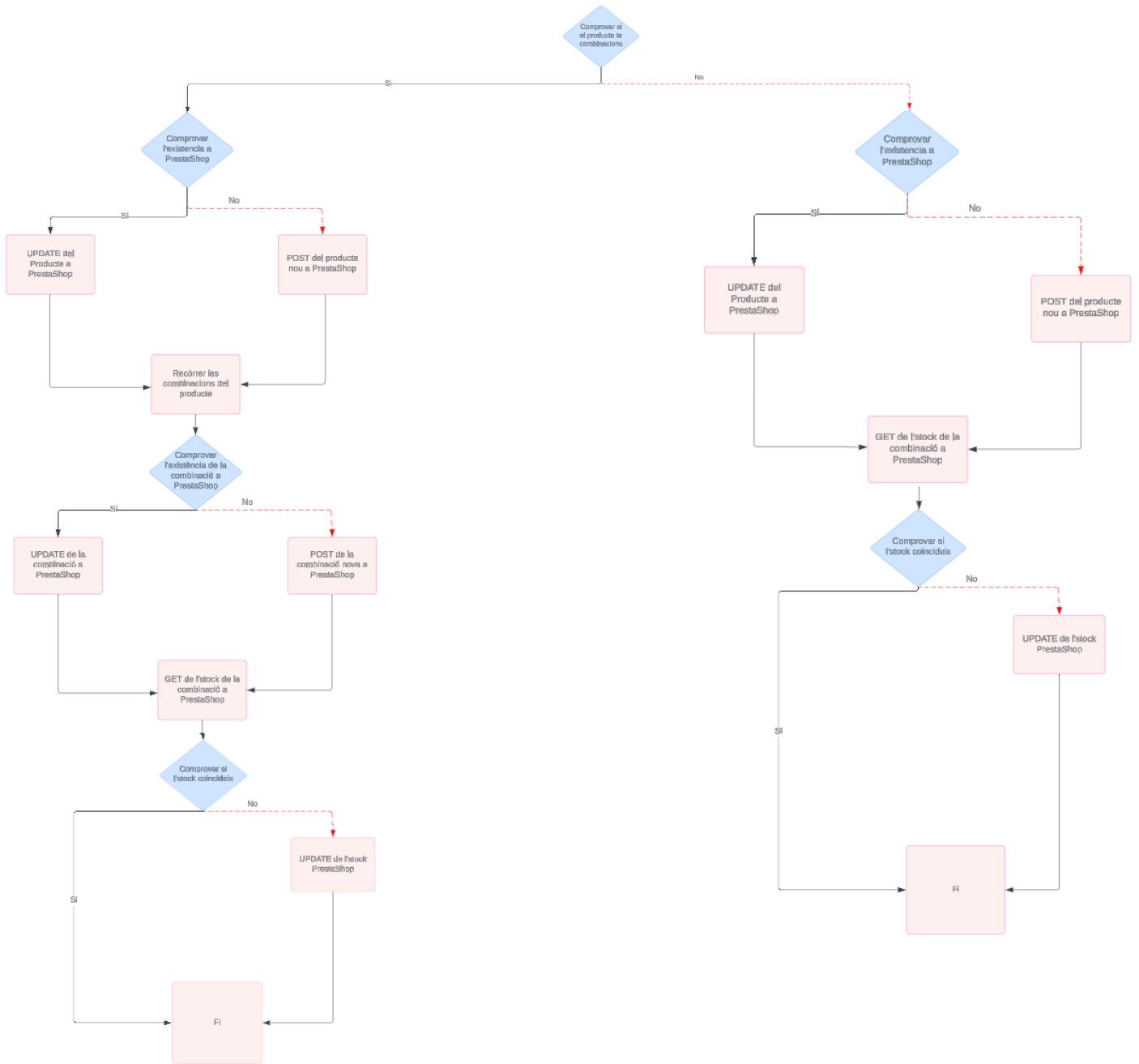
4.2 Arquitectura del sistema

4.2.1 Estructura del mòdul u



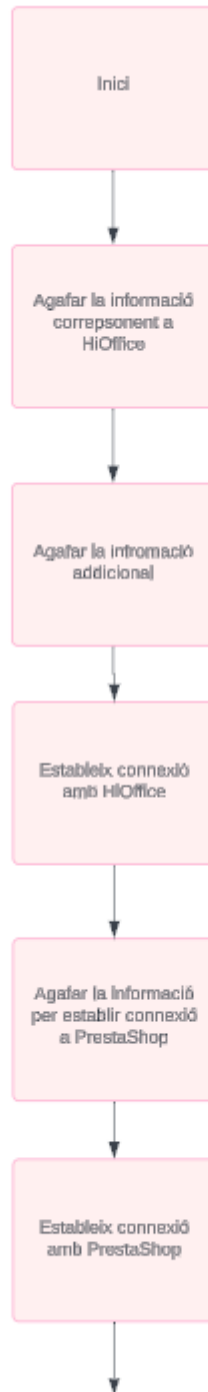


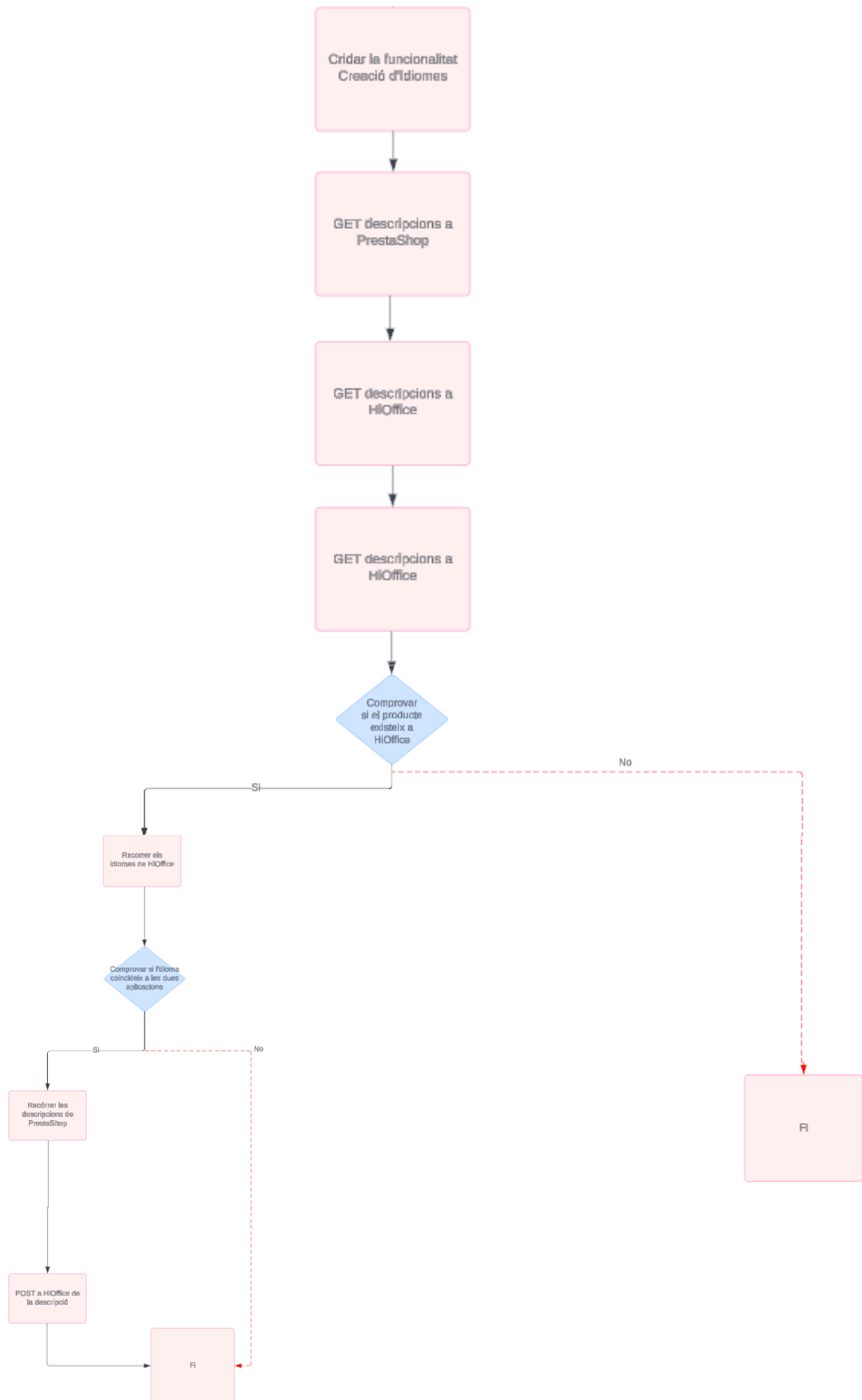




4.2.2 Estructura del mòdul dos

4.2.2.1 Estructura Creació descripció





4.3 Detall d'implementació

En aquest apartat es detallarà els processos específics que es duen a terme dins de l'aplicació per tal d'executar totes les seves funcionalitats de manera correcta i el tractament de les dades de cada una de les plataformes.

4.3.1 Establiment de connexió amb la API de HiOffice

```
public static function hioPosApiLogin($hioPosConnection)
{
    $responseLogin = Http::get($hioPosConnection->url, [
        'email' => $hioPosConnection->email,
        'password' => $hioPosConnection->password,
        'isoLanguage' => $hioPosConnection->iso_language,
    ]);

    $hioPosXMLLogin = (array) simplexml_load_string($responseLogin->body());

    return $hioPosXMLLogin['customerWithAuthTokenResponse'];
}
```

4.3.2 Establiment de connexió amb la API de PrestaShop

```
$prestashopAPI = new PrestaShopWebservice ($prestashopConnection ->
url,$prestashopConnection -> api_key, false);
```

4.3.3 GET d'un recurs de PrestaShop

```
$prestashopGroupsAttribute = json_decode(json_encode($prestashopAPI->
get(['resource' => 'product_options', 'display' => '[id]'))));
```

A causa de que la resposta de la API és amb XML, es codifica i descodifica en JSON per tal de poder tractar les dades.

4.3.4 POST a la PrestaShop

```
$response = $webservice->edit([
    'resource' => 'products',
    'id' => (int)$product ->id_presta,
    'putXml' => $xmlString,
]);
```

4.3.5 Comprovació de l'existència d'un article

```
$product = $prestaProductsCombinations->firstWhere(function($value,
int $key) use ($article) {
    return $value->mpn == $article->cod_articulo;
});
```

4.3.6 Creació del fitxer d'importació per crear la relació a la taula de relacions

```
$datatoImport = [
    'idTipusTransformacio' => $idTransformacio,
    'tipoTransformacio' => $attribute->dimension,
    'idHiOffice' => $attribute->cod_atributo,
    'descriptionHiOffice' => $attribute->valor,
    'idPresta' => $dataPresta ->idPresta,
    'descriptionPresta' => $dataPresta ->
    descriptionPresta,
];
```

Es fa una comprovació per tal de saber si la relació ja ha estat creada prèviament, en el cas que ja estigui creada es farà un UPDATE dels valors de la taula, i sinó es farà un POST.

```
if($attribute ->codTaulaRelacio != 0){
    $datatoImport['idTransformacioTaula'] = $attribute->
    >codTaulaRelacio;

    $jsonData = json_encode($datatoImport, JSON_PRETTY_PRINT);
    file_put_contents('data.json', $jsonData);

    HioposController::hioPosApiImportation($serviceExtraConf->
    >udpdaterelacions_id, $hioPosXMLLogin, 'data.json');
}else{
    $jsonData = json_encode($datatoImport, JSON_PRETTY_PRINT);
    file_put_contents('data.json', $jsonData);
    HioposController::hioPosApiImportation($serviceExtraConf->
    >taularelacions_id, $hioPosXMLLogin, 'data.json');
}
```

5 Proves

En aquest apartat l'objectiu és descriure les proves realitzades per verificar el correcte funcionament de les funcions desenvolupades en l'aplicació. Aquestes proves asseguraran que les funcions de creació i actualització dels atributs, categories, productes, combinacions, idiomes, descripcions i imatges operen correctament entre les dues plataformes.

5.1.1 Metodologia

Per tal de fer les proves s'han seguit els següents passos: inclou la preparació de l'entorn de proves, l'execució de les funcions i la validació d'aquestes. A continuació es descriu de manera més detalladament cada un d'aquests passos.

5.1.1.1 Preparació de l'entorn

El primer pas consisteix en la preparació de l'entorn. Per garantir que les proves fossin representatives poden d'aquesta manera reproduir tots els escenaris possibles i reals, es va seleccionar bases de dades de proves tant per la HiOffice com en la plataforma B.

A HiOffice, s'utilitza una base de dades de prova que conté un conjunt de dades representatives, inclou diferents productes, tant en combinacions com sense, les categories i atributs més comuns que es poden trobar en el sector retail. Aquesta base de dades està dissenyada per simular un entorn real per tal de poder avaluar com les funcions de l'aplicació gestionen les dades.

En PrestaShop, es configura una base de dades de prova, aquesta serà configurada un cop totes les funcions que s'executen de HiOffice cap a PrestaShop, s'hagin executat. En aquesta base dades es tindrà els productes, les combinacions, les categories i els atributs de HiOffice. En aquesta plataforma s'afegiran imatges i les descripcions, per tal de tindre l'entorn de prova preparat per poder comprovar el correcte funcionament de les funcions.

5.1.1.2 Execució de les funcions

Un cop es té preparat l'entorn de proves, es procedeix a l'execució de cada una de les funcions. Per cada una de les funcions es van realitzar els següents passos:

1. Creació de les dades a HiOffice, es creen les dades a HiOffice com poden ser els productes o les categories
2. Transferència de les dades, s'utilitza l'aplicació desenvolupada per a transferir les dades des de la plataforma de HiOffice a PrestaShop.
3. Recepció i validació de PrestaShop, es verifiquen que les dades han sigut creades o actualitzades correctament a la plataforma.

En el cas de la transferència de les dades de PrestaShop a HiOffice es segueixen els mateixos passos esmentats.

Creació dels Atributs

Per comprovar la creació tant del grup d'atributs com dels atributs, es creen diferents atributs en la HiOffice. Seguidament s'executa la funció de transferència d'atributs en l'aplicació web i es verifica en la PrestaShop que els nous atributs i els seus valors s'han creat correctament.

Creació de Categories

La prova de creació de categories va consistir a crear diferents categories en la HiOffice, i transferir-les a la PrestaShop. Es verifica que totes les categories són creades de manera correcta a la PrestaShop, seguint la mateixa estructura que a la HiOffice.

Creació i actualització de Productes i Combinacions

Per tal de comprovar la creació i l'actualització s'executa de manera correcta, es creen productes a la HiOffice, i es transfereixen a través de l'aplicació a la PrestaShop, seguidament es fa la comprovació que tots els productes i les seves combinacions, en cas que en tinguin, han estat creats de manera correcta. Un cop tots estan a la PrestaShop, es fa alguna modificació algun d'aquests productes i es tornen a transferir les dades, per tal de comprovar si l'aplicació actualitza els canvis de manera correcta a la PrestaShop.

Creació dels idiomes

Per verificar la creació dels idiomes, es segueix el mateix esquema que en la resta de les funcions, es creen els idiomes a la HiOffice, i s'executa, un cop acabat l'execució es comprova que tots han estat creats a la PrestaShop.

Creació de les Descripcions

En aquest cas, la direcció és en direcció contrària. En aquest cas s'escriuen les descripcions a la PrestaShop, un cop creades, es transfereixen les descripcions a la HiOffice, es verifica que apareixen de manera correcta.

5.1.1.3 Resultat de les proves

Un cop executades les proves per cada una de les funcions, es documenten que totes les funcionalitats operen de manera correcta segons els requisits establerts. Qualsevol error que es va poder arribar a trobar durant el desenvolupament de les funcions, es van corregir per tal d'obtenir els resultats satisfactoris. La següent taula resumeix els resultats obtinguts.

Funció	Resultat	Observacions
Creació d'Atributs	Satisfactori	Tant el grup d'atributs, com els atributs es transmeten correctament amb tots els seus valors
Creació de Categories	Satisfactori	Les categories i les seves respectives seccions es creen correctament i mantenen l'estructura jeràrquica
Creació i Actualització de Productes i Combinacions	Satisfactori	La creació dels atributs i les combinacions s'executa de manera correcta, i les actualitzacions es transfereixen correctament
Creació d'Idiomes	Satisfactori	Els idiomes es repliquen correctament a la PrestaShop.
Creació de Descripció	Satisfactori	Les descripcions es transmeten correctament en cada un dels idiomes.

6 Implantació

Aquest apartat té l'objectiu de proporcionar una guia de com configurar i utilitzar l'aplicació web, dissenyada per facilitar la transferència d'informació entre diferents plataformes. L'aplicació s'ha desenvolupat per tal d'optimitzar la integració i sincronització de dades, permetent una gestió eficient i centralitzada de la informació. A través d'aquest manual, s'explicaran els passos necessaris per assegurar una correcta configuració i un ús òptim de l'aplicació, abordant aspectes clau com el registre en la plataforma, els requisits tècnics i la configuració de les connexions amb les API de les plataformes externes.

6.1 Requisits mínims

Per tal, d'assegurar el correcte funcionament de l'aplicació web és essencial primer de tot està registrat a la plataforma, és a dir, l'usuari ha d'estar registrat a la plataforma de l'aplicació web, per tal de saber les funcionalitats que està interessat a tindre, ja que només pot estar interessat en la integració de dades d'una plataforma a l'altre i no ambdues direccions.

Un altre requisit necessari, és disposar de connexió a Internet estable i fiable, pel fet que la interacció amb les APIs de les plataformes externes requereix una comunicació continua i sense interrupcions. Per tal, d'establir connexió amb les plataformes externes, és necessari les claus d'accés (API keys) de les dues plataformes. Aquestes claus són essencials per autenticar les sol·licituds que es fan, permetent així, l'intercanvi de dades entre l'aplicació i les plataformes.

A part de tots aquests requisits, també és necessària informació addicional específica per la configuració de la connexió, com per exemple l'URL de la PrestaShop, o especificar la direcció en les que van les dades, és a dir si van de la HiOffice a PrestaShop o a la inversa, i altres paràmetres necessaris que poden ser necessaris per a les plataformes externes.

6.2 Configuració

La configuració de l'aplicació web, per tal de connectar-la amb les plataformes externes, segueix una sèrie de passos detallats que assegurin una integració correcta i funcional.

El primer pas per configurar l'aplicació és accedir a ella a través de el seu URL, un cop està l'usuari a la pàgina d'inici, ha d'iniciar sessió utilitzant les seves credencials registrades. Un cop que l'usuari està registrat s'haurà d'indicar en quina direcció es volen les dades, és a dir, si la direcció es vol des de la HiOffice a PrestaShop o a la inversa, per si per exemple es vol l'exportació d'articles de la HiOffice a PrestaShop, o es vol les funcionalitats de la PrestaShop a HiOffice.

Per connectar l'aplicació amb les dues plataformes, és necessari introduir a l'aplicació la clau API. Aquest procés, l'usuari ha de ser capaç d'accedir al back end de les dues plataformes i obtindrà les claus APIs per tal de posar-les a l'aplicació. Un cop s'obté la clau API, ha de tornar a l'aplicació web i posar les claus en el camp corresponent, és important que la clau s'ingressa correctament on li toca, ja que de no ser així hi hauria errors de connexió.

A més de la clau API, és necessària informació addicional per a completar la configuració com pot ser l'URL de les plataformes o identificadors específics per tal de realitzar exportacions o importacions. Un cop s'han introduït totes les dades necessàries, s'ha de guardar aquesta configuració. S'haurà d'introduir cada quant vol l'usuari que s'executi les diferents funcions, i l'aplicació ja estarà llesta per ser utilitzada.

6.3 Pressupost

En aquest apartat es detallaran els costos associats al desenvolupament de l'aplicació a mida per tal de facilitar la interoperabilitat entre dues plataformes. Aquest pressupost inclou les despeses relacionades amb la planificació, disseny, desenvolupament, proves implantació i manteniment del sistema.

Tasca	Hores	Preu
<i>Planificació i anàlisi</i>		
Reunions inicials	5	75,00 €
Anàlisi de requisits	15	225,00 €
<i>Disseny del Back-End</i>		
Disseny del model d'interoperabilitat	25	375,00 €
<i>Desenvolupament del Back-End</i>		
Programació back-end i proves	60	900,00 €
Integració dels sistemes i proves	20	300,00 €
<i>Proves, implantació i validació</i>		
Preparació entorn	3	45,00 €
Proves d'integració	20	300,00 €
Ajustos finals	25	375,00 €
<i>Manteniment i suport</i>		
Manteniment (6 mesos)	30	450,00 €
Suport tècnic (6 mesos)	20	300,00 €
	Total	3.345,00 €

Aquest pressupost és una estimació basada en les hores necessàries per a cadascuna de les fases del projecte, a un preu de 15 euros per hora treballada. Els costos podrien variar en funció de les necessitats addicionals o possibles imprevistos que puguin sorgir durant el desenvolupament del projecte. Aquest pressupost també inclou una previsió de manteniment i suport tècnic durant sis mesos després de la seva implementació, per tal de poder ajudar a l'usuari en el cas que es trobés amb algun tipus de problema.

7 Conclusions

En aquest treball s'ha desenvolupat una aplicació web que esta destinada a facilitar la transferència d'informació entre dues plataformes diferents, tenint unes certes característiques cada una d'elles. El projecte ha inclòs des de la implementació del codi necessari per tal d'entendre com funcions, fins la configuració i les proves de les funcions desenvolupades, amb l'objectiu d'assegurar una total integració fluida i eficient de les dades.

Al llarg del desenvolupament s'han portat a cap diverses etapes. La primera de totes va ser la anàlisis dels requisits, es va fer un estudi de les necessitats i els requisits funcionals per la integració entre les dues plataformes. Aquest inclou la identificació dels tipus de dades que necessitaven ser transferides i les operacions que s'havien de portar a cap per tal del bon funcionament, seguidament es va dur a terme el desenvolupament de les funcionalitats, s'han implementat les diferents funcions esmentades i explicades durant el treball (creació i actualització de productes i combinacions i la creació dels atributs i les categories, dades que van des de la HiOffice cap a PrestaShop, i la creació i actualització de les descripcions des de la PrestaShop cap a HiOffice).

La següent etapa ha sigut la configuració de l'aplicació, posar les dades necessaris com per exemple les API keys, i altres paràmetres necessaris per tal de poder establir connexió entre les dues plataformes. Finalment, s'han realitzat diferents probes per cada una de les funcions implementades, utilitzant dues plataformes d'exemple, creant a cada una d'elles la informació necessària per assegurar que les transferències de dades es realitzen de manera correcta.

7.1 Assoliment dels objectius

Aquest treball, tal i com s'ha mencionat al principi, tenia de manera general l'objectiu de solucionar la problemàtica de la interoperabilitat entre les plataformes de comerç electrònic i sistemes de gestió empresarial a través d'una aplicació. Més concretament s'ha buscat crear una solució per la plataforma PrestaShop i el sistema de gestió empresarial HiOffice. Avaluant els objectius plantejats al principi d'aquest treball, es pot afirmar que s'han complert els objectius establerts.

En primer lloc, es volia analitzar el model de dades i el funcionament entre les dues plataformes. Aquest objectiu s'ha assolit, ja que s'ha realitzat un estudi de les estructures de dades de les dues plataformes i els mecanismes operatius de cada una d'elles.

El segon d'ells, era dissenyar un model d'interoperabilitat que sigues capaç de proporcionar una solució eficient per la transferència de dades entre les plataformes. L'objectiu s'ha complert satisfactòriament. L'aplicació que he desenvolupat permet la transferència de les dades entre les dues plataformes, en les dues direccions, permetent millorar de manera significativa l'eficiència i poder reduir les tasques manuals i el temps de les persones.

Finalment, es va dur a terme una prova de concepte de la proposta, podent demostrar la capacitat de l'aplicació per connectar les dades entre les dues plataformes. La prova ha validat la viabilitat de la solució plantejada, confirmant el seu funcionament.

7.2 Desafiaments trobats

Durant el desenvolupament del projecte, s'ha presentat diversos desafiaments que han requerit un estudi i una resolució tan efectiva com siga possible. En aquest apartat s'exposaran els obstacles que ha presentat la implementació per tal de poder transformar les dades. A través de la descripció dels desafiaments enfrontats i les estratègies utilitzades per resoldre'ls.

7.2.1 Desafiaments tècnics

En aquest apartat es presentaran els desafiaments tècnics. A continuació es presenten els desafiaments més significatius que han sorgit.

7.2.1.1 Relacionar els productes amb les seves associacions

Un dels reptes principals que va presentar el projecte es va centrar en com indicar a l'API de la PrestaShop les associacions dels articles, siguin les categories, els idiomes o les combinacions.

La PrestaShop, en la inserció de les associacions dels productes, opera exclusivament amb les identificacions (id's) de les associacions. En conseqüència, no es va poder utilitzar cap altre camp per identificar aquest id que retornava la plataforma. En el cas de voler associar per exemple, una categoria al producte de HiOffice a PrestaShop, va sorgir un impediment, ja que la creació d'una combinació implica associar-ho amb el id d'aquestes categories, de manera que no es podia obtenir d'una manera òptima quin id de la PrestaShop, pertanyia al id de la HiOffice.

Després d'analitzar diverses estratègies per implementar les associacions, la solució òptima que vaig trobar va ser crear una taula de relacions. Això va implicar assignar un identificador específic a cada tipus d'associació, com per exemple, assignar l'id 12 a tots els colors. En aquesta taula, a més de la identificació, es guarda la relació dels id's de les dues plataformes. Mitjançant una sol·licitud GET a aquesta taula, es pot accedir a l'id de la PrestaShop per a cada categoria, mida o color d'un determinat producte, mentre es recorre l'array de productes o subproductes. Això proporciona una manera eficient de gestionar i identificar les associacions necessàries per a una correcta integració entre les dues plataformes.

7.2.1.2 Tractar les dades

El com tractar les dades va ser un dels reptes principals a l'hora de desenvolupar l'aplicació.

Un dels principals desafiaments va ser en com gestionar les dades de la manera més òptima possible. Es va plantejar la qüestió de si era més adequat recórrer producte per producte i enviar cada un d'ells individualment a la plataforma de destinació, o si era més eficient per un enfocament global.

Aquesta decisió va requerir una anàlisi dels avantatges i inconvenients del que suposaria implementar cada una de les opcions. Finalment, es va decidir optar per enviar cada un dels productes de manera individual, ja que, és més fàcil controlar les dades, en aquesta decisió va influir el fet d'implementar en com crear les relacions posteriorment. Un altre factor que es va tindre en compte és les vegades que s'hauria de recórrer l'array, pel fet que si s'implementava de manera global, primer s'havia de fer una distinció dels productes que contenen combinacions i els que no, i seguidament, els que ja estan creats a la plataforma i els que no, de tal manera que s'hauria de recórrer l'array el doble de vegades. En canvi, amb la implementació de producte a producte, recórrer l'array de productes de HiOffice una sola vegada, i a més és molt més senzill i pràctic controlar les dades, a l'hora d'actualitzar-les o crear-les, ja que s'ha dissenyat pensant amb tots els diferents casos que es podria trobar a l'hora d'exportar i importar dades d'una plataforma a l'altre.

En el tractament de dades, un altre desafiament que es va presentar és la necessitat de definir de manera concreta i clara la direcció de la migració de les dades, garantint així, que l'exportació i importació de les dades sempre es realitzi en una direcció específica per evitar possibles errors i complicacions futures. Aquesta decisió va ser crucial per tal d'assegurar la integritat i coherència de les dades. La importància d'aquest aspecte és a causa de la complexitat en la manipulació i sincronització de les dades entre les dues plataformes. En el cas de permetre, per exemple, l'exportació i importació bidireccionals dels productes, es podrien generar diversos problemes com ara la possibilitat de duplicats, inconsistències en les dades, dificultats per identificar quina és la font d'origen i una gestió complicada dels registres.

Un altre dificultat presentada per aquest projecte, és l'estructura marcada per cada una de les API's de cada plataforma. Aquesta divergència ha complicat en ocasions la implementació de la solució òptima, ja que les estructures definides han establert un marc rígid, on la major part de les vegades, ha restringit la flexibilitat i l'eficiència del procés d'integració de les dades. La fixació de les estructures de dades per part de les API's han actuat com un obstacle per obtenir una integració harmoniosa i eficient, les diferències entre les estructures han requerit ajustos i adaptacions per tal de poder garantir la compatibilitat i la coherència en el flux de les dades. Tot i això, s'ha treballat per tal de superar aquestes limitacions de les estructures i desenvolupar estratègies per afrontar aquesta dificultat.

7.2.2 Desafiament de gestió

Un dels desafiament més significatius va ser la necessitat d'anticipar i abordar la possibilitat de situacions i casos que podrien sorgir durant la importació i exportació de les dades entre les dues plataformes. Va ser necessària una immersió des de la perspectiva de l'usuari final i la consideració dels diversos escenaris, ja que era possible que l'usuari no segueixi de manera estricta el flux de dades establert. Es va desenvolupar l'aplicació per tal d'anticipar i gestionar els diversos escenaris potencials.

Un dels altres desafiament va ser posar-se a la pell de l'usuari i comprendre les seves necessitats, objectius i comportaments per tal de donar prioritat a les preferències que pot arribar a tindre l'usuari. Es tractava d'identificar què era realment important per l'usuari i com es podria plasmar en la funcionalitat de les aplicacions. Per abordar aquest desafiament, es va interaccionar amb diferents empreses i professionals, amb l'objectiu de recopilar els aspectes més importants i crítics per l'usuari.

A partir d'aquesta interacció, es va proporcionar una base sòlida pel desenvolupament de l'aplicació, basant-se en tot moment en les preferències que es tenia en el mercat.

Quan es parla de les preferències, es fa referència, per exemple, quines dades són més pràctiques que s'importin d'una plataforma a l'altre o quines són les dades realment útils en transferir entre les dues plataformes.

7.3 Millores i treball futur

Una de les motivacions per realitzar aquest treball ha sigut la meva aspiració a poder-me dedicar al desenvolupament de programari per al sector retail. He identificat la problemàtica de la interoperabilitat entre diferents sistemes de gestió empresarial i plataformes de comerç electrònic. Aquest treball ha servit com a prova de concepte per abordar el problema.

Tot i que els objectius s'han assolit, executant el treball he identificat certes limitacions, que es podrien millorar en un futur. Una de les futures millores seria la cerca d'una manera encara més eficient de transferir les dades entre les dues plataformes, és a dir el temps que tarda en transferir les dades, ja que actualment la capacitat de les API i l'estructura de les dades entre les dues plataformes limita i provoquen retards quan es tracta una gran quantitat de dades. Optimitzar aquets processos i minimitzar el temps de transferència és una de les millores que es podrien fer per un futur.

Un altre millora del futur, seria l'expansió de que aquesta interoperabilitat es pogués aplicar més enllà de les dues plataformes actualment suportades. L'objectiu futur serà que la solució sigui més general i polivalent, de tal manera, que pugui ser utilitzada amb qualsevol plataforma de comerç electrònic i sistema de gestió empresarial del mercat.

En conclusió, els objectius inicials s'han complert amb èxit, ja que s'ha assolit poder desenvolupar una aplicació que solucioni el problema de la interoperabilitat entre les dues plataformes, d'una manera eficient i sent capaç de reduir el temps i els errors, però el projecte encara té un gran potencial de creixement i millora, per tal de poder adaptar la solució desenvolupada en entorns més reals i complexos. La investigació i desenvolupament continuaran per tal de perfeccionar l'aplicació i estendre aquesta funcionalitat a nivell més global de sistemes.

7.4 Aprenentatges

Durant el desenvolupament d'aquest treball, he pogut adquirir coneixements i he pogut aplicar les competències adquirides al llarg de la meva carrera a la universitat. Estudiant Administració i Direcció d'Empreses, he pogut entendre millor les necessitats i desafiaments als que s'enfronten les empreses en la gestió de dades i processos operatius. Mentre que amb el grau de Tècniques de Desenvolupament d'Aplicacions Web i Mòbil m'ha proporcionat totes les habilitats necessàries per dissenyar i implementa la solució presentada en aquest treball.

Referències bibliogràfiques

Coppola, M. (2023, junio 12). *Qué es WordPress, para qué sirve y cómo utilizarlo*. Hubspot.es. <https://blog.hubspot.es/website/guia-completa-wordpress>

da Silva Serapião Leal, G., Guédria, W., & Panetto, H. (2019). Interoperability assessment: A systematic literature review. *Computers in Industry*, 106, 111–132. <https://doi.org/10.1016/j.compind.2019.01.002>

Dorado, A. J., González Serrano, C., & Jiménez Builes, J. A. (2015). Model of Semantic interoperability between Learning Management Systems. *Informador técnico*, 79(1), 53. <https://doi.org/10.23850/22565035.137>

Evolución del comercio electrónico: fases y futuro. (s/f). Beetrack.com. Recuperado el 5 de junio de 2024, de <https://www.beetrack.com/es/blog/evolucion-del-comercio-electronico>

Hioffice. (2023, septiembre 18). Cide Intelligence Solutions; Software y hardware para empresas y profesionales. <https://cide.cat/hioffice/>

Historia de Prestashop. (s/f). Modulosprestashop.com. Recuperado el 5 de junio de 2024, de <https://www.modulosprestashop.com/blog/actualidad-prestashop/la-historia-y-evolucion-de-prestashop>

Martínez, G. (2015, mayo 3). *Qué es WordPress y sus características principales*. Webempresa. <https://www.webempresa.com/wordpress/que-es-wordpress.html>

Métodos de pago de e-commerce: instrucciones para elegir uno. (s/f). Stripe.com. Recuperado el 5 de junio de 2024, de <https://stripe.com/mx/resources/more/ecommerce-payment-methods>

PrestaShop. (s/f). *PrestaShop 8 documentation*. PrestaShop. Recuperado el 5 de junio de 2024, de <https://devdocs.prestashop-project.org/8/>