

DÉPARTEMENT D'INFORMATIQUE

## PROJET DE FIN D'ÉTUDES

MASTER SCIENCES ET TECHNIQUES  
SYSTÈMES INTELLIGENTS & RÉSEAUX

---

# CONCEPTION ET DÉVELOPPEMENT D'UNE PLATEFORME E-COMMERCE / PIM POUR LE GROUPE ARCELORMITTAL À BASE DE LA SOLUTION HYBRIS

---



LIEU DU STAGE : SQLI Oujda

**Réalisé par :**

- KHALDI Mohammed

**Encadré par :**

- Pr. CHAKER Ilham
- Mr. HASSAINEAbdelmounaim
- Mr. MEJDOUBI Abderrahmane
- 

**Soutenu le 13.06.2018 devant le jury composé de :**

- |                   |   |                |
|-------------------|---|----------------|
| - Pr. R. BEN ABOU | Faculté des Sciences et Techniques de Fès | (Président)    |
| - Pr. K. ZENKOUAR | Faculté des Sciences et Techniques de Fès | (Examinateur)  |
| - Pr. A. BEGDOURI | Faculté des Sciences et Techniques de Fès | (Examinatrice) |
| - Pr. I. CHAKER   | Faculté des Sciences et Techniques de Fès | (Encadrante)   |

**Année Universitaire 2017 – 2018**

# Dédicaces

*Je dédie ce présent mémoire ...*

*A mes chers parents,*

*Qui ont œuvré pour ma réussite, de par leur amour, leur soutien, tous les sacrifices consentis et leur précieux conseils, pour toute leur assistance et leur présence dans ma vie, reçois à travers ce travail aussi modeste soit-il, l'expression de mes sentiments et de mon éternelle gratitude. Puisse Dieu, le Très Haut, vous accorder santé, bonheur et longue vie et faire en sorte que jamais je ne vous déçoive.*

*A mon frère, ma sœur,*

*Je leur dédie ce travail en témoignage de mon grand amour et ma gratitude infinie.*

*A mes amis,*

*Avec qui j'ai partagé les moments de bonheur  
Et de malheur, des moments qui seront gravés dans ma*

*Mémoire.*

*A tous ceux qui m'aiment.*

*Mohammed KHALDI*

# Remerciements

A l'issue de mon stage de fin d'étude, je tiens en premier lieu à exprimer ma profonde gratitude à Dieu le tout puissant, le tout miséricordieux, Qui m'a donné grâce et bénédiction pour bien mener ce projet à terme dans de très bonnes conditions.

Mes plus sincères remerciements et reconnaissance vont à mon encadrant pédagogique Madame. **Ilham CHAKER** pour sa disponibilité et les remarques constructives qu'il m'a prodiguées tout au long de mon stage.

J'adresse aussi mes vifs remerciements à Mme **Hanae DADI**, Manager du Skill Center Java/BI et M. **MEJDOUBI Abderrahmane**, chef du projet AMDS, M. **Abdelmounaim HASSAINE**, chef du projet AMC, de m'avoir fait part de leur équipe dans le cadre de mon projet de fin d'étude. Je voudrais remercier également l'équipe de développement AMDS et AMC pour leur accueil chaleureux, leurs conseils et surtout l'ambiance amicale et amusante qui règne au sein de l'équipe. Je voudrais remercier spécialement M.**Hassane ETTIFOURI** pour son support technique côté Hybris et qui n'a jamais hésité de répondre à toutes mes questions. Ainsi que tout le corps professoral de la Faculté des Sciences et Techniques de Fès pour avoir assuré un environnement théorique et pratique convenable tout au long de la formation. Que toutes les personnes ayant contribuées de près ou de loin à l'aboutissement de ce travail, trouvent ici l'expression de mes sentiments les meilleurs.

# Résumé

Le présent rapport est l'aboutissement du travail effectué dans le cadre du projet de fin d'études au sein de l'équipe AMDS et AMC de l'agence SQLI Oujda, pour l'obtention du diplôme du master Systèmes intelligents et réseaux. L'objectif principal des deux projets auxquels j'ai participé consiste à la réalisation d'une plateforme e-commerce B2B et un PIM à base de la solution Hybris 6.3.

Dans un premier temps et pour monter en compétence, j'ai suivi une formation technique et fonctionnelle en Hybris et Spring. Cette formation m'a permis de se familiariser avec la plateforme Hybris, son principe et ses différentes fonctionnalités d'un point de vue technique et fonctionnel.

Par la suite, j'ai intégré l'équipe AMDS pendant deux mois avant de se déplacer vers l'équipe AMC où j'ai passé un mois.

Durant les deux projets, nous avons commencé par une analyse fonctionnelle et technique des spécifications du client afin de bien saisir la problématique, puis nous avons entamé l'étape de conception pour finir avec la mise en œuvre tout en respectant l'architecture JavaEE et les conventions de développement Hybris.

Pour la mise en place des deux projets, et pour assurer l'agilité et la réactivité avec le client, le choix est tombé sur Scrum comme méthodologie de gestion de projet, avec un cycle de développement itératif et incrémental.

*Mots clés : e-commerce, B2B, JavaEE, Hybris, Scrum*

# Abstract

This report is the result of the work done as part of the end-of-studies project within the AMDS and AMC team of the SQLI Oujda agency, to obtain the diploma of the Master's degree in Intelligent Systems and Networks. The main objective of the two projects is the realization of a B2B e-commerce platform and a PIM based on the Hybris 6.3 solution.

First, to improve our skills, I followed a technical and functional training in Hybris and spring framework. This training allowed me to become familiar with the Hybris platform, its principle and its various functionalities from a technical and functional point of view.

Afterwards, I joined the AMDS team for two months before moving to the AMC team where I spent a month.

During both projects, we started with a functional and technical analysis of the customer's specifications in order to understand the problem, and then we moved to the design to finish with the implementation while respecting the JavaEE architecture and the Hybris development agreements.

For the implementation of both projects, and to ensure agility and responsiveness with the customer, the choice fell on Scrum as a project management methodology, with an iterative and incremental development cycle.

**Keywords :** *e-commerce, B2B, JavaEE, Hybris, Scrum*

# Table des matières

Dédicaces .....	i
Remerciements.....	ii
Résumé.....	iii
Abstract.....	iv
Liste des figures .....	vi
Liste des tableaux .....	viii
Liste d'Acronymes et Abréviations.....	ix
Introduction générale.....	1
Chapitre 1 : Contexte général du stage.....	4
1 Présentation de l'organisme d'accueil.....	5
1.1 Le groupe SQLI.....	5
1.1.1 Présentation générale.....	5
1.1.2 SQLI en chiffre .....	5
1.2 SQLI Oujda.....	7
1.3 Le groupe SQLI et Hybris.....	8
2 Contexte du projet.....	9
2.1 Présentation du projet AMDS.....	9
2.2 Présentation du projet AMC.....	10
2.3 Pourquoi le client a choisi la solution Hybris ?.....	10
3 Conduite du projet .....	11
3.1 Présentation de la méthodologie Scrum.....	11
3.2 Implémentation de Scrum sous Jira.....	13
3.2.1 Présentation de Jira.....	13
3.2.2 Cycle de développement adopté.....	15

3.2.3	Personnalisation de Jira au projet.....	17
Chapitre 2 : Etude et analyse fonctionnelle .....		19
1	Analyse et spécification .....	20
1.1	Projet AMDS.....	20
1.1.1	Analyse de l'existant.....	20
1.1.2	Spécifications fonctionnelles .....	20
1.1.3	Analyse des besoins .....	27
1.1.4	Flux AMDS .....	31
1.2	Projet AMC.....	32
1.2.1	Analyse de l'existant.....	32
1.2.2	Spécifications fonctionnelles .....	33
1.2.3	Analyse des besoins .....	34
Chapitre 3 : Présentation technique .....		38
1	Architecture.....	39
1.1	Présentation de la plateforme Hybris.....	39
1.2	Architecture d'Hybris Commerce.....	39
1.2.1	Content.....	40
1.2.2	Orders.....	41
1.2.3	Channel.....	41
1.2.4	Commerce.....	41
1.2.5	Platform .....	42
1.3	Hybris Accelerators.....	42
1.4	ImpEx .....	43
1.5	Architecture technique du Distrishop.....	44
1.6	Architecture de développement.....	44
2	Plateforme Hybris .....	45

2.1	Les évolutions de la version 6.3 de Hybris .....	45
2.1.1	Hmc vs Backoffice .....	45
2.1.2	Accelerator B2C versus Accelerator B2B.....	47
3	Outils et technologies utilisés .....	47
3.1	Environnements de développement et outils.....	47
3.1.1	Eclipse .....	47
3.1.2	Jira.....	48
3.1.3	Confluence.....	48
3.1.4	Git .....	48
3.2	Frameworks utilisées .....	49
3.2.1	Spring.....	49
3.3	Web Service.....	49
Chapitre 4 : Réalisation et mise en œuvre.....		51
1	Configuration initiale du projet.....	52
1.1	Réunion de lancement .....	52
1.2	Vérification de l'environnement de travail .....	52
1.3	Découpage des fonctionnalités par itération .....	53
2	Mise en œuvre.....	53
2.1	Projet AMDS.....	53
2.1.1	Gestion des catalogues et navigation.....	53
2.1.2	Gestion des utilisateurs.....	56
2.1.3	Gestion des comptes clients .....	57
2.1.4	Stratégie d'affichage du détail d'un produit .....	58
2.1.5	Le processus de check-out.....	60
2.1.6	Gestion du contenu CMS.....	63
2.1.7	Gestion des prix .....	64

2.1.8	Import et conversion des medias via Cronjobs .....	64
2.2	Projet AMC.....	66
2.2.1	Création des tables contenant les informations SOFIE .....	66
2.2.2	Import des donnés.....	68
3	Etat d'avancement et perspective.....	69
3.1	Etat actuel .....	69
3.2	Les tests.....	69
3.3	Déploiement.....	69
3.4	Perspectives.....	70
	Conclusion générale.....	71
	Webographie.....	73
	Annexes.....	74

# Liste des figures

Figure 1:Effectif de SQLI.....	6
Figure 2 : Chiffre d'affaires 2017.....	6
Figure 3 : Répartition des agences.....	6
Figure 4 : Partenaires de SQLI .....	7
Figure 5 : Skill Centers de SQLI.....	8
Figure 6 : L'équipe AMDS / AMC de SQLI Oujda.....	13
Figure 7 : Interface Jira de SQLI.....	14
Figure 8 : Graphe de traitement d'une demande client sous Jira.....	15
Figure 9 : Cycle de développement d'un projet.....	16
Figure 10 : La Recherche par facette .....	21
Figure 11 : Détails d'un produit dans le Distrishop.....	24
Figure 12 : Processus de check-out dans le site .....	25
Figure 13 : Diagramme de cas d'utilisation Gestion des commandes.....	28
Figure 14 : Diagramme d'activité de la gestion des comptes clients.....	29
Figure 15 : Diagramme de séquence de la gestion des comptes clients.....	30
Figure 16 : Interface principale de l'application SOFIE.....	32
Figure 17 : Données SOFIE / PIM.....	33
Figure 18 : Diagramme de cas d'utilisation de gestion des produits AMC.....	35
Figure 19 : Processus d'exécution du Hot Folder.....	36
Figure 20 : Architecture d' Hybris.....	40
Figure 21 : Architecture générale de Hybris Commerce Suite.....	42
Figure 22 : Architecture technique du Distrishop.....	44
Figure 23 : Architecture de développement .....	45
Figure 24 : Vue générale du HMC.....	46
Figure 25 : Vue générale du Backoffice .....	47
Figure 26 : Les catalogues du site AMDS via Backoffice .....	54
Figure 27 : Menu de navigation du site.....	54
Figure 28 : Bas de page du site.....	55
Figure 29 : Diagramme de classe du bas de page .....	56

Figure 30 : Diagramme de classe de l'inscription - client non agence.....	57
Figure 31 : La classe Zip Code Model.....	58
Figure 32 : Page détail d'un produit.....	59
Figure 33 : Nouveaux attributs du ProductModel.....	60
Figure 34 : Schéma expliquant l'opération de data populating .....	60
Figure 35 : Les étapes de la création d'une commande dans le Distrishop.....	61
Figure 36 : Classe B2BUnit Model modifiée .....	62
Figure 37 : Client AMDS bloqué.....	62
Figure 38 : Processus de la création d'une page CMS .....	63
Figure 39 : Page panier du Distrishop .....	64
Figure 40 : Les deux cronjobs pour l'import et la conversion des médias.....	65
Figure 41 : Lancer un Cronjob depuis le backoffice.....	66
Figure 42 : Nouveaux nœuds.....	67
Figure 43 : Processus d'exécution du Hot Folder.....	68
Figure 44 : Menu de navigation sur Mobile.....	74
Figure 45 : Page Login et inscription sur Mobile .....	75
Figure 46 : Page Mon compte sur IPad.....	<b>Erreur ! Signet non défini.</b>

# Liste des tableaux

Tableau 1 : Modules logiciels internes.....	31
Tableau 2 : Services externes.....	31
Tableau 3 : Permissions des employés AMC.....	34
Tableau 4 : Récapitulatif des dates prévisionnelles des itérations du projet AMDS et AMC .	52
Tableau 5 : Découpage des fonctionnalités par sprint .....	53
Tableau 6 : Mapping des termes entre Hybris et les specs .....	57

# Liste d'Acronymes et Abréviations

<i>Abréviation</i>	<i>Désignation</i>
AMDS	<i>ArcelorMittal Distribution System</i>
AMC	<i>ArcelorMittal Construction</i>
B2C	<i>Business To Client</i>
B2B	<i>Business To Business</i>
CMS	<i>Content Management System</i>
CSV	<i>Comma-Separated Values</i>
DAO	<i>Data Access Object</i>
DTO	<i>Data Transfer Object</i>
ERP	<i>Enterprise Resource Planning</i>
HMC	<i>Hybris management console</i>
ImpEx	<i>Import / Export</i>
PDP	<i>Product Détails Page</i>
PLP	<i>Product List Page</i>
IDE	<i>Integrated Development Environment</i>
JSP	<i>Java Server Pages</i>
REST	<i>Representational State Transfer</i>
SVN	<i>Apache SubVersion</i>
UML	<i>Unified Modeling Language</i>
PIM	<i>Product Information Management</i>
SOFIE	<i>Système Organisationnel Fédérateur de l'Information en Europe</i>

# Introduction générale

Après avoir participé à la 12ème édition de la compétition E-Challenge organisée par SQLI Oujda et Rabat, j'ai passé les trois étapes de la compétition et j'ai décroché un stage PFE pré-embauche de 6 mois au sein de SQLI Oujda.

Mon stage de projet de fin d'études, effectué au sein de l'agence SQLI Oujda pour une période de six mois, s'est déroulé en deux grandes parties. J'ai travaillé dans un premier temps sur la réalisation d'un Distrishop AMDS à base de la plateforme Hybris puis j'ai participé au projet AMC.

- **AMDS (ArcelorMittal Distribution System)**

La vente en ligne représente aujourd'hui une grande partie des achats des ménages et des entreprises. Il est donc important d'être de la partie en ayant une boutique en ligne pour son entreprise. Il existe maintes raisons de vendre en ligne, citons-nous quelques-unes : Augmenter son chiffre d'affaires, élargir sa base des clients potentiels, agrandir son secteur géographique et ainsi de suite. D'où le besoin des sites e-commerce est en croissance.

C'est dans ce cadre que le groupe **ArcelorMittal** spécialisé dans la production sidérurgique a décidé de lancer le projet AMDS pour construire un site e-commerce qui sera connu par suite par le slogan « **Distrishop** ».

Ce site doit garantir une gestion efficace de l'ensemble des activités de vente en ligne, de plus il doit être facilement administrable et configurable selon les spécifications de l'entreprise, et il servira à vendre des produits massifs récupérables depuis le point de service le plus proche du client.

Ma contribution durant ce projet se résume dans la conception et le développement de plusieurs besoins client notamment : le processus d'inscription d'un client en tenant compte de son type, la stratégie d'affichage des produits et les prix dans la page qui liste l'ensemble des produits (PLP) ainsi que la page dédiée aux détails d'un produit spécifique (PDP), l'enrichissement du processus dédié à la création des commandes, l'import des médias et association avec leurs produits via des jobs (services) qui peuvent être lancés manuellement ou bien de façon périodique.

- **AMC (ArcelorMittal construction)**

AMC a décidé de mettre en œuvre un **PIM** à la base de la solution SAP Hybris pour compléter le paysage des applications AMC et disposer d'un référentiel de données commun afin de prendre en charge les processus fonctionnels d'AMC et éviter le risque d'avoir une désynchronisation des informations produits.

Actuellement AMC se base essentiellement sur et un ensemble d'applications informatiques autonomes développées en interne, d'où le risque d'avoir une incohérence des données.

L'usage d'un logiciel de **PIM** va permettre donc de centraliser la gestion de l'information liées aux produits vendus en vue d'en assurer une distribution efficace et précise vu que le **PIM** assure la consolidation et la gestion centralisée des informations et des attributs des produits.

Les informations stockées au niveau du PIM seront exploitées par la suite dans un web shop similaire à celui du AMDS.

Pour le projet AMC, mon intervention consiste à développer une grande partie qui concerne la création des nouvelles tables au niveau du PIM pour que son architecture soit la même que l'application utilisée actuellement par le client, ainsi que la mise en place d'une solution efficace pour permettre un import transparent des données dans le nouveau PIM.

Afin de mettre en relief la conduite et les objectifs, ce rapport est structuré comme suit :

Le premier chapitre sera consacré à la présentation de l'organisme d'accueil, ses secteurs d'activités ainsi que ses différents pôles. Je vais par la suite présenter le contexte métier et avant de clôturer le premier chapitre, une brève présentation de la méthodologie adoptée pour la réalisation de ces deux projets sera élucidée.

Une étude et analyse fonctionnelle fera l'objet du deuxième chapitre et dans lequel je vais exposer les différentes spécifications fonctionnelles, et les piliers nécessaires la réalisation des deux projets.

Par la suite, une analyse conceptuelle qui identifie les différents acteurs de notre système ainsi que les différents cas d'utilisation et les principaux scénarios possibles.

Après avoir exposé les spécifications fonctionnelles dans le chapitre précédent. Lors du troisième chapitre, dans un premier temps, je vais mener une étude technique décrivant l'architecture technique. Ensuite une présentation détaillée de la plateforme Hybris et ce qu'elle offre comme services et extensions, ensuite je vais clôturer ce chapitre par la présentation des différents outils et technologies choisis pour le développement.

Dans le quatrième chapitre et dernier, nous allons décrire, selon une approche itérative et incrémentale, les différentes phases par lesquelles on est passé pour mettre en œuvre la solution qui répond aux exigences des clients.

Enfin, je vais finir par une conclusion, suivie d'un ensemble de références webographies ainsi que plusieurs annexes destinées à illustrer davantage mon travail.

# Chapitre 1 : Contexte général du stage

Dans la première partie de ce chapitre et pour mieux circonscrire le contexte de déroulement du stage, une présentation de l'organisme d'accueil s'impose, ses secteurs d'activités ainsi que ses différents pôles, par la suite je vais situer les deux projets dans leurs contextes généraux tout en élucidant leurs objectifs pour finir avec une présentation du processus de développement adopté.

# 1 Présentation de l'organisme d'accueil

## 1.1 Le groupe SQLI

### 1.1.1 Présentation générale

Créé en 1990, SQLI est le partenaire de référence des entreprises dans la définition, la mise en œuvre et le pilotage de leur transformation digitale. Elle est organisée en agences de proximité, afin de conserver le maximum de réactivité face aux besoins de ses clients. Son positionnement unique lui permet de répondre de façon globale aux enjeux de performance :

- **Performance business** : e-commerce, marketing digital & social, mobilité e-communication ...
- **Performance de l'entreprise** : entreprise collaborative, poste de travail, solutions métier, intégration de S.I ...

Ses 2400 collaborateurs sont répartis en France (Paris, Lyon, Toulouse, Bordeaux, Rouen et Nantes), en Suisse (Lausanne et Genève), au Luxembourg, en Belgique (Bruxelles), aux Pays-Bas et au Maroc (Rabat et Oujda).

### 1.1.2 SQLI en chiffre

Le groupe SQLI suit un rythme de développement élevé au niveau des projets, des ressources humaines, et chiffre d'affaires.

- **21 ans de projets et missions de haut niveau**

SQLI assoit son développement sur une expertise technologique de pointe et sur sa politique intense de veille et R&D. En 2005, SQLI devient la 1ère SSII française à obtenir la certification CMMI niveau 3[1].

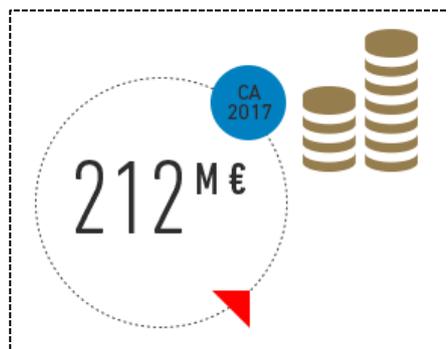
- **2400 collaborateurs en France et à l'étranger**

SQLI est déjà présente au Benelux (Belgique Nederland Luxembourg), en Espagne, en Suisse et au Maroc. Elle compte plus de 2400 collaborateurs en France et à l'étranger[1].



*Figure 1:Effectif de SQLI*

SQLI affiche un solide niveau de résultats, elle a réalisé un chiffre d'affaires de 212M€ en 2017, stable à périmètre constant grâce à une croissance au cours du second semestre[1].



*Figure 2 : Chiffre d'affaires 2017*

- **32 agences en France et à l'étranger**

Le Groupe SQLI est implanté en France à Paris, Rouen, Nantes, Bordeaux, Toulouse, Lyon et Lille... et présent en Suisse (Lausanne et Genève), au Luxembourg, en Belgique (Bruxelles, Gent), aux Pays-Bas, au Maroc (Rabat et Oujda)[1].



*Figure 3 : Répartition des agences*

- **23 partenaires, leaders des solutions technologiques**

Le positionnement de SQLI à la conjonction du monde du digital et du système d'information de l'entreprise est sa valeur ajoutée. Le groupe a noué un réseau de partenaires pour adresser ses challenges de façon la plus performante possible, et fournir le meilleur conseil, la meilleure solution[1].

Figure 4 : Partenaires de SQLI

## 1.2 SQLI Oujda

Dans le cadre de sa stratégie de développement alliant taille et densité technologique, le Groupe SQLI s'est doté de pôles spécialisés, disposant chacun d'une expertise technologique et/ou métier spécifique.

Pour accompagner le développement de son pôle dédié Open Source de Poitier, en septembre 2006, le Groupe SQLI ouvrait, en partenariat avec l'Université Mohammed Premier, le tout premier centre de R&D / offshore entièrement dédié aux technologies Open Source.

Depuis, SQLI bénéficie, au sein même de l'Université, d'un fort potentiel de ressources de très haut niveau de qualité et de locaux entièrement équipés des technologies les plus avancées. En contrepartie, les experts du Groupe interviennent dans le cadre de formations spécifiques, telles que CMMI, et contribuent à l'enrichissement des cursus scolaires de modules complémentaires.

Concernant l'organisation du groupe SQLI Oujda, il s'agit d'une répartition selon 3 segments qui à leur tour sont répartis en sept équipes :

- Les technologies et usages internet.
- SAP (Net Weaver, CRM...).
- La Business intelligence (BO, open source ...)

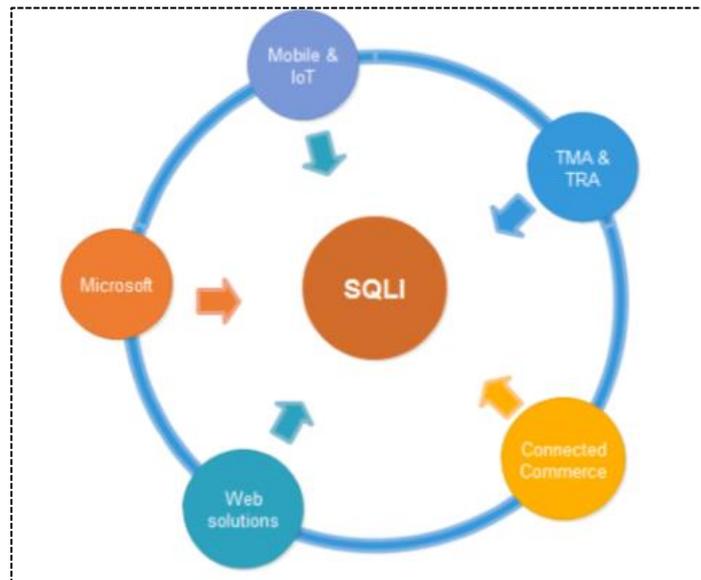


Figure 5 : Skill Centers de SQLI

### 1.3 Le groupe SQLI et Hybris

Le groupe SQLI se fait l'un des intégrateurs Hybris, SAP hybris est son partenaire depuis 2009. En effet, plus que 250 collaborateurs y travaillent sur des projets Hybris permettant aux cyberacheteurs de prolonger l'expérience digitale au-delà du web.

Nouveaux services, customisation produit, cohérence multi-support etc... Ces problématiques sont au cœur des investissements de ses clients. Au Maroc, l'agence SQLI de Rabat, depuis 2010, dirige 80% de son activité vers l'e-commerce notamment Hybris, et ce n'est que jusqu'à cette année que l'agence d'Oujda a décidé d'inaugurer son activité Hybris avec ces deux projets.

## 2 Contexte du projet

ArcelorMittal est un groupe qui se spécialise dans la production sidérurgique et de l'exploitation minière depuis 2006, avec une empreinte industrielle dans 19 pays et une présence dans 60 pays. Le groupe est un fournisseur d'acier de qualité pour plusieurs secteurs d'activité tels que l'automobile, le bâtiment, l'énergie et l'emballage[2].

- **Premier producteur d'acier en France**

En France, ArcelorMittal compte **17 200** salariés répartis sur ses **40** sites de production, ses centres de distribution et de services et ses quatre sites de R&D[2].

ArcelorMittal a produit **10 millions** de tonnes d'acier liquide en 2015 en France (soit environ 10% de la production mondiale du groupe) qu'il transforme ensuite en bobines, feuilles, tôles, ronds, barres[2]...

Les deux projets AMDS et AMC sont développés pour le compte du même client (ArcelorMittal), pour deux filiales différentes.

### 2.1 Présentation du projet AMDS

Afin d'élargir sa base clientèle, ArcelorMittal a décidé de lancer la transformation numérique de ses activités de vente par l'appropriation d'une plateforme e-commerce B2B en se basant sur la solution SAP Hybris.

Les principaux objectifs de ce projet sont :

Objectifs techniques :

- Développer un site e-commerce **stable** et **flexible** basée sur une solution standard du marché.
- Garantir une architecture hautement **performante**.
- Mettre en œuvre une solution **compatible** avec les technologies du groupe.

Objectifs fonctionnels :

- Une stratégie de développement **Cross-canal**.
- Développement **international** favorisé par l'ouverture facile de nouveaux pays grâce à une plateforme nativement multi langues, prix, devise ...
- Visibilité de l'enseigne et de l'offre produit sur le web d'une façon **durable**.

## 2.2 Présentation du projet AMC

**ArcelorMittal Construction** est le premier fournisseur mondial de systèmes de construction en acier et de solutions **composées** de **profils** et de **panneaux** sandwich pour toitures, bardages, sols mixtes et structures légères. L'entreprise propose des systèmes de construction répondant aux exigences thermiques, acoustiques, d'étanchéité à l'air et au feu, ...

Vu la complexité des produits construits par AMC, l'entreprise a décidé d'abandonner l'usage des solutions distinctes et centraliser la gestion de l'information des produits dans un **PIM** en se basant sur le Module **PCM** (Product content Management) fournis par la solution SAP Hybris.

Selon les spécifications fournis par le client, Le PIM doit garantir au moins les fonctionnalités suivantes :

- Consolider les données et le contenu liés aux produits et rendre-les facilement accessibles et gérables.
- Contrôler tout le contenu de produit pour tous les canaux à partir d'un seul endroit.
- Classifier et catégoriser les produits, gérer les versions de catalogue et importer plusieurs catalogues de fournisseurs.

## 2.3 Pourquoi le client a choisi la solution Hybris ?

1. Hybris est parmi les principales solutions européennes gagnant la confiance des références si remarquables.

2. Hybris propose des solutions multicanales pour la vente des produits unifiant toute interaction numérique ou physique.

3. Le client utilise en préalable l'ERP SAP ce qui s'accorde facilement avec Hybris.

## 3 Conduite du projet

### 3.1 Présentation de la méthodologie Scrum

L'approche Agile propose de réduire considérablement l'effet tunnel des approches traditionnelles de type cycle en V ou cycle en cascade en donnant davantage de visibilité, en impliquant le client du début à la fin du projet et en adoptant un processus itératif et incrémental. Elle considère que le besoin ne peut être figé et propose au contraire de s'adapter aux changements de ce dernier. Mais pas sans un minimum de règles.

En effet, Scrum est une méthodologie agile pouvant être appliquée à presque n'importe quel projet et c'est elle la plus utilisée souvent. Son processus est adapté aux projets à **évolution rapide**, selon les exigences clients. Le développement de logiciels en guise de cette méthodologie progresse par une série d'itérations appelé **sprints**, qui durent d'une à quatre semaines que nous l'avons fixé en trois semaines. Le modèle Scrum suggère que chaque sprint commence par **une réunion de planification** et se termine par **un examen**.

Au centre de chaque projet se trouve un **productbacklog** qui représente l'ensemble du travail à faire. Ce dernier est formulé au cours de la phase de planification des itérations.

Le processus de réalisation du projet est formé par une série d'itérations courtes appelées sprints. Chaque sprint vise à mettre en œuvre un nombre fixe d'éléments du **productbacklog** dans ce qu'on appelle un **sprint backlog**, choisi lors de la réunion de planification et durant cette réunion elle-même, l'équipe en entier participe au **découpage** et **chiffrage** des tâches. À la fin du sprint, l'équipe Scrum et les parties-prenantes invitées, se réunissent pour effectuer la **revue** de sprint ; celle-ci dure au maximum 4 heures. L'objectif de cette revue est de valider le logiciel qui a été produit. L'équipe commence par énoncer les items du carnet de produit qu'elle a réalisés. Elle effectue ensuite une **démonstration** du logiciel produit. Tout au long du sprint, l'équipe se réunit à l'occasion d'un rendez-vous quotidien appelé **mêlée**. Chaque membre de l'équipe décrit le travail à faire le jour même, les progrès de la veille, et tous les blocs effacer.

Pour que les réunions soient courtes, la mêlée quotidienne est censée être réalisée avec tout le monde dans la même salle et en position debout pendant toute la durée de la séance.

Lorsque suffisamment d'items du backlog sont réalisés, l'équipe effectue **les tests d'intégration** et génère la documentation nécessaire pour le déploiement du produit.

Dans le développement agile, les équipes Scrum sont prises en charge par deux rôles spécifiques :

**Le Scrum Master** : Il est responsable de la compréhension, de l'adhérence et de la mise en œuvre de la méthode. C'est le leader ; il est au service de l'équipe ; il assiste chaque rôle de l'équipe dans son activité et promet le changement des interactions entre les rôles dans le but de maximiser la valeur du produit de l'équipe. Son autorité s'exerce sur le processus de développement (définition de la durée des itérations, des modalités de tenues et de l'ordre du jour des réunions Scrum, etc.) ; mais il ne dispose d'aucune autorité sur les membres de l'équipe.

**Le Product Owner** : Il représente les entreprises, les clients ou les utilisateurs, et guide l'équipe vers la construction du bon produit ; il porte la vision du produit à réaliser. Il est responsable de maximiser la valeur du produit et du travail de l'équipe de développement.

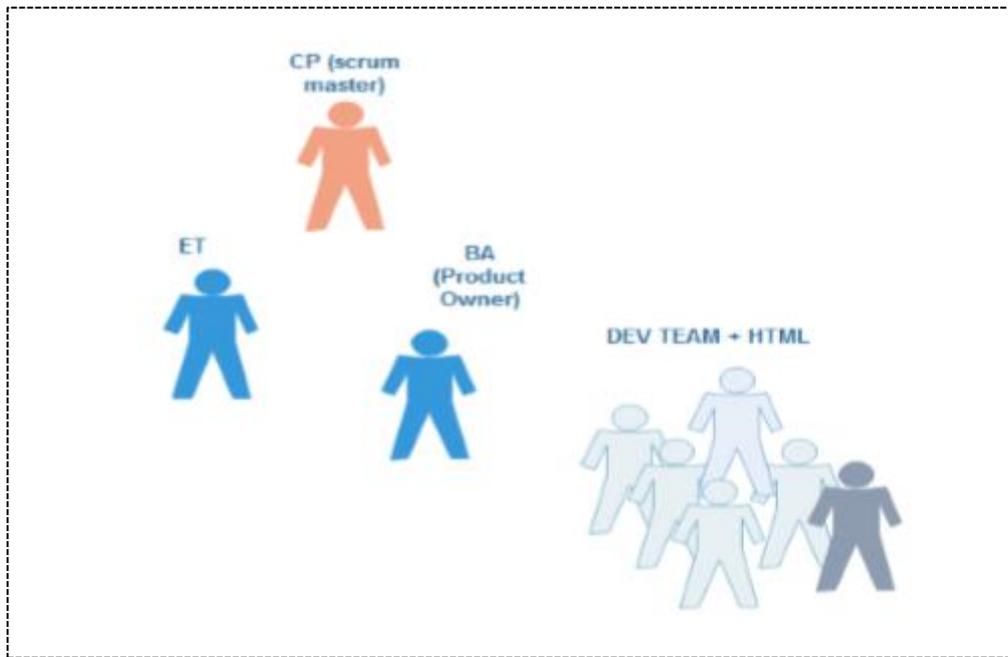


Figure 6 : L'équipe AMDS / AMC de SQLI Oujda

Notre équipe a été renforcée par deux développeurs de SQLI Rabat qui nous ont accompagnés durant les premiers sprints du projet pour nous transmettre leurs savoir-faire d'Hybris.

Durant la réalisation des deux projets, l'équipe de SQLI Oujda était en collaboration continue avec l'équipe de SQLI Rabat et celle de SQLI Belgique.

## 3.2 Implémentation de Scrum sous Jira

Dans cette partie, nous allons expliquer l'implémentation de la méthode Scrum tout en démontrant le cycle de développement adopté. Avant d'entamer cela, une brève présentation de l'outil Atlassian Jira s'impose.

### 3.2.1 Présentation de Jira

**Atlassian** est un éditeur de logiciel australien fondé en 2002. Il développe des produits orientés vers les gestionnaires de projet (manager) et aux développeurs de logiciels. Actuellement, son produit le plus connu est JIRA. C'est un système de suivi de bugs, de gestion des incidents et des projets.

Il combine des méthodes révolutionnaires telles que XP (eXtreme-Programming), Scrum, Lean développement, aidant les équipes à fournir rapidement un niveau supérieur de qualité de code

en meilleure harmonie avec les besoins des clients. Les dernières versions de JIRA sont complètes et plus faciles à utiliser. Voici quelques fonctionnalités importantes :

- JIRA possède une interface très configurable.
- JIRA définit les « issues » devant être suivies (tâches, livrables, etc.).
- JIRA peut piloter les processus de développement sur l'intégralité du cycle de vie (Application Lifecycle Management ou ALM).
- JIRA propose le suivi de toutes les activités définies (tâches, jalons, livrables, anomalies...).
- JIRA intègre particulièrement **Greenhopper**. Cela facilite la gestion et le workflow des projets agiles.
- JIRA gère également la notification de l'intégration continue et la gestion des publications.

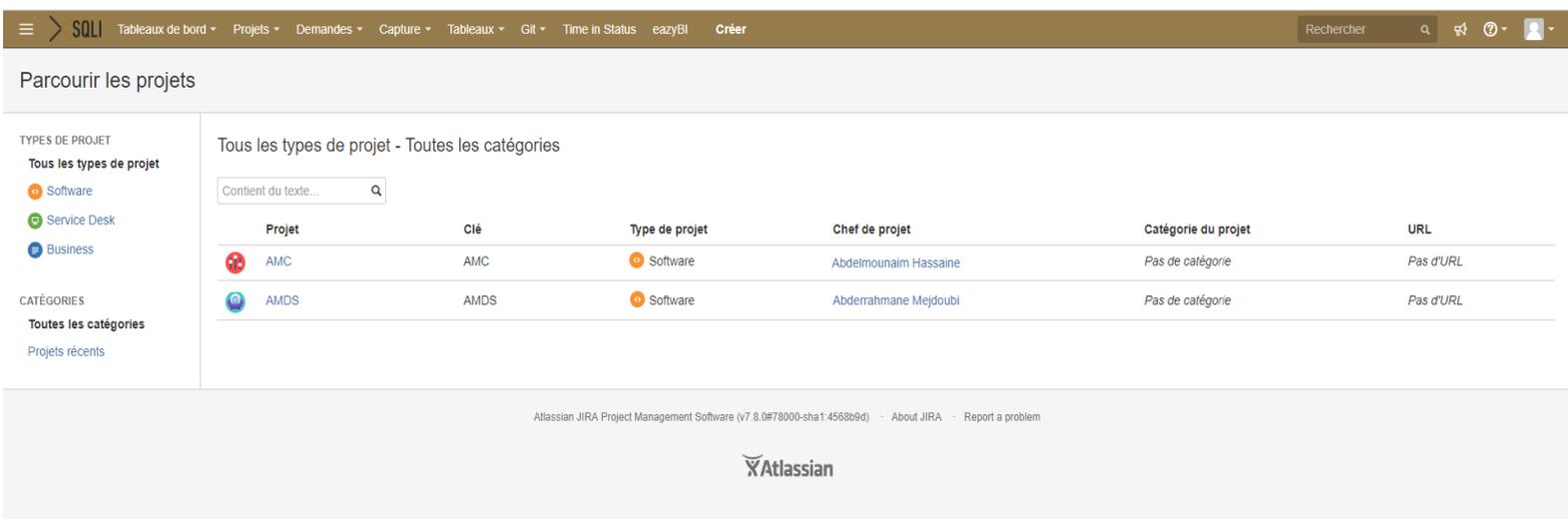


Figure 7 : Interface Jira de SQLI

A titre indicatif le workflow de Jira adopté par SQLI pour le traitement des demandes levées par ses clients est le suivant[3]:

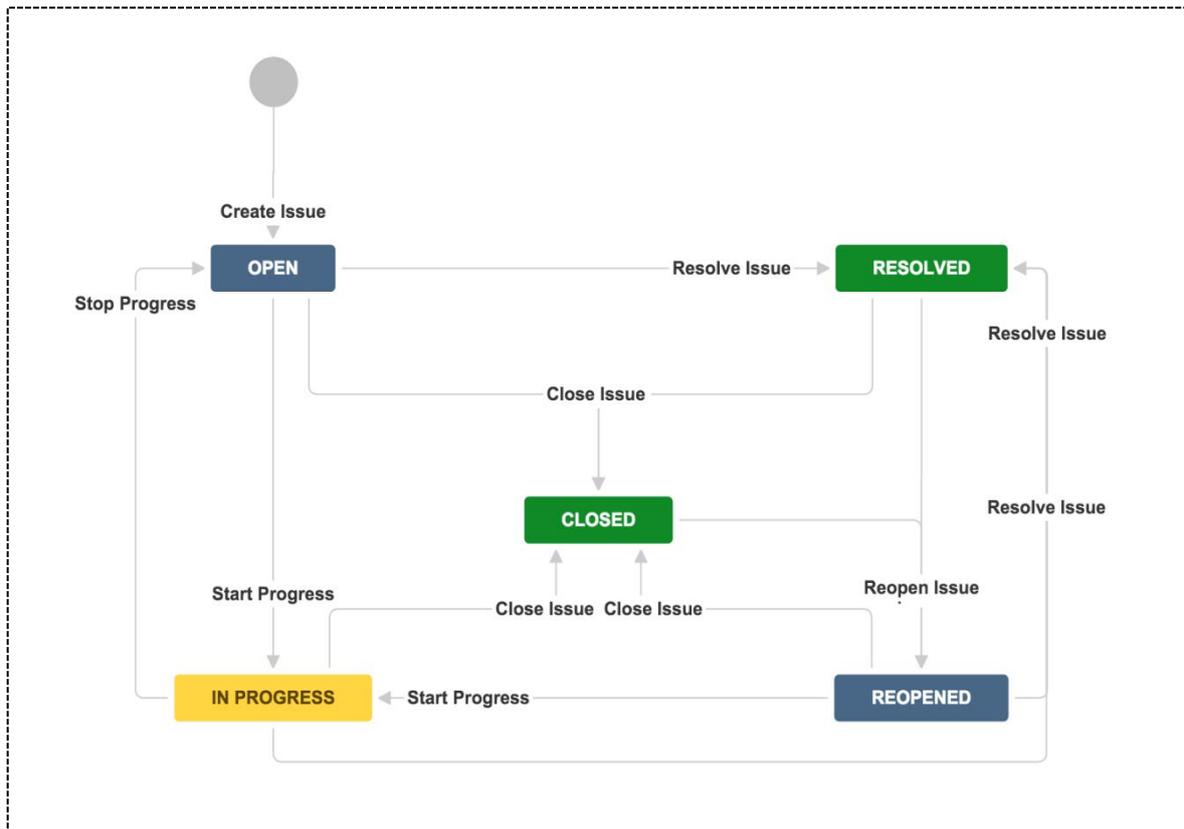


Figure 8 : Graphe de traitement d'une demande client sous Jira

### 3.2.2 Cycle de développement adopté

Chaque projet se déroule sur différentes étapes, généralement cruciales. Sa gestion consiste à l'exécution de ces étapes, aux termes desquelles des points de contrôle doivent être réalisés. Chaque étape fait l'objet d'un livrable et d'une validation à partir d'un document spécifique. Cela permet aux différents intervenants d'avoir une vision claire sur le déroulement du projet, de maîtriser la conformité des livrables à la définition des besoins ainsi que de s'assurer de l'adéquation aux objectifs de coûts et de délai.

Ci-dessous un schéma synthétisant le cycle de développement d'un projet à SQLI :

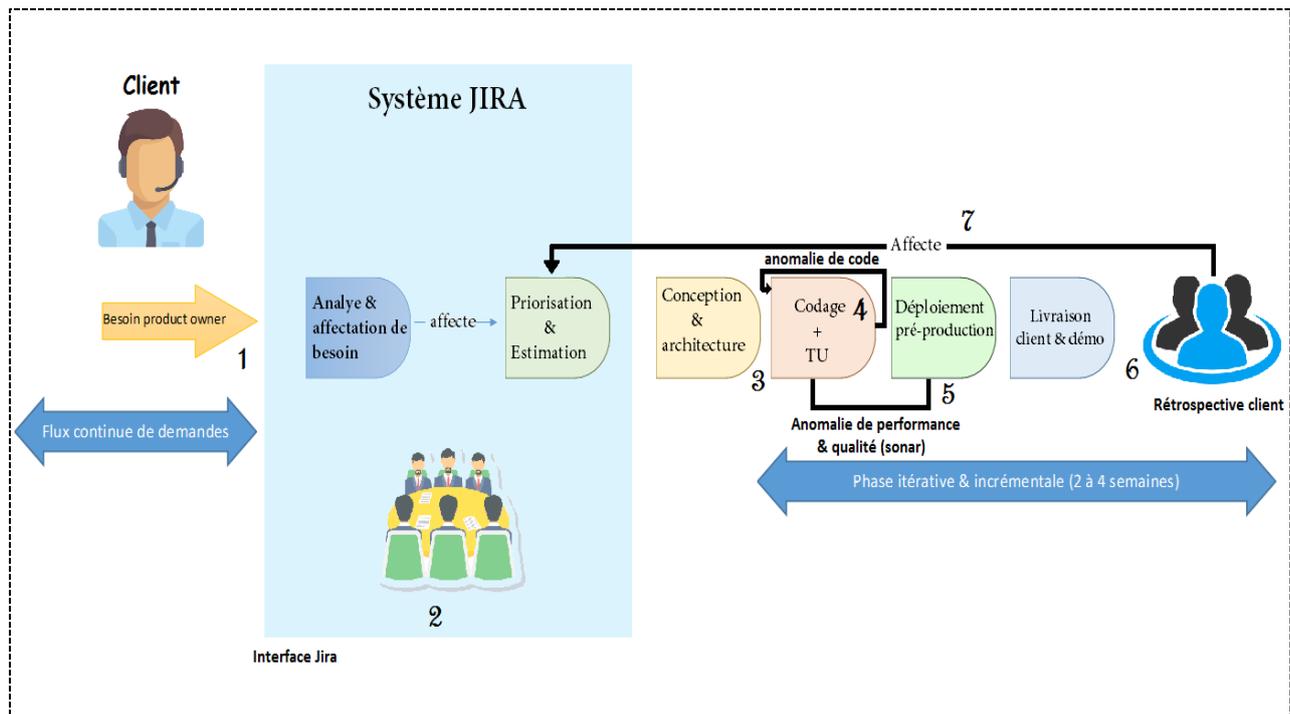


Figure 9 : Cycle de développement d'un projet

Comme le montre le schéma ci-dessus, le développement itératif s'organise en une série de développement de très courte durée (2 à 4 semaines) nommées itérations, 3 semaines dans notre cas. Le résultat de chaque itération est un système partiel exécutable, testé et intégré (mais incomplet). Chaque itération comprend ses propres activités :

**Analyse de besoin** : Au début de chaque itération une réunion de planification aura lieu. Elle a pour but de sélectionner, prioriser et estimer les exigences à implémenter avec la répartition des tâches sur les différents membres de l'équipe (un chef de projet, développeurs et testeurs). En ce qui concerne le chiffrage des tâches, tous les membres d'équipe y participent. Dans notre projet, nous avons estimé les charges en heure. (8h est équivalent à 1 journée).

**Conception & Architecture** : avant d'entamer la phase de programmation, on définit l'ensemble des paramètres (architecture logiciel, modèle logique) qui nous aide à comprendre comment réaliser le système.

**Les tests unitaires** : sont développés en parallèle avec le codage. En cas de détection d'anomalie, le code en question est remanié.

**Déploiement dans le serveur de pré-production** : Si l'exigence à implémenter concerne une partie de code qui présente des défaillances de performance ou des violations de la qualité, le code en question est remanié.

**Déploiement dans le serveur de production** : se fait chez les clients cibles. Une réunion rétrospective avec ce dernier est tenue afin de révéler les points forts et les points à améliorer de cette phase et identifier les différents problèmes rencontrés. Cette réunion permet également d'entreprendre des actions correctives pour la prochaine itération.

### 3.2.3 Personnalisation de Jira au projet

Jira est orienté aux développeurs de logiciels. Il fournit un ensemble d'outils pour la gestion complète des projets informatiques en combinant les principales méthodes agiles (Scrum, XP). Cependant, sa configuration initiale ne répondait pas entièrement aux exigences du système managérial actuel de l'entreprise et du projet. C'est pourquoi, il fallait fournir un grand effort de personnalisation en vue de préserver l'ensemble des bonnes pratiques constituant son référentiel. Ce paramétrage effectué par le chef de projet concerne les parties suivantes :

- **Types de projet** : dans cette première étape, le chef du projet a choisi entre les types de projet valable sur Jira celui du type « **Software** ».
- **Configuration du système de type de demande** : Ces types de demande permettent de catégoriser les demandes au sein d'un projet Jira. Lors du deux projets, on s'est servi des types suivants : *Tâche, sous-tâches, bogue, anomalie, besoin d'information* et *change request*.
- **Les tableaux utilisés** : Lors des deux projets, deux types de tableaux ont adopté *Kanban* et *Scrum*.

Tout au long du cycle de développement, les clients ou Product Owner injectent des demandes dans le système Jira. Ces derniers seront après analysés et reformulés si nécessaire, affectés à une version du projet.

## **Conclusion**

Dans ce chapitre, j'ai défini le contexte général de mon projet de fin d'études en présentant, tout d'abord, l'entreprise d'accueil, qui est l'entreprise SQLI, une entreprise qui accompagne ses clients dans la mise en œuvre et le pilotage de leur transformation digitale. J'ai également introduit le contexte des deux projets AMDS et AMC qui constituent l'objet du sujet de PFE. Finalement, une présentation de la conduite de projet qui est une combinaison de JIRA et du Scrum.

# **Chapitre 2 : Etude et analyse fonctionnelle**

Dans ce chapitre je vais exposer l'ensemble des résultats et la manière avec laquelle nous avons réalisé l'étude et l'analyse fonctionnelle des deux projets : L'analyse des fonctionnalités, les cas d'utilisation et les diagrammes tracés lors de l'étude.

# 1 Analyse et spécification

## 1.1 Projet AMDS

### 1.1.1 Analyse de l'existant

ArcelorMittal vise une transformation rapide et agile sur tous les canaux afin d'offrir une expérience d'exception à ses clients pour effectuer leurs achats à tout moment peu importe la complexité de son business. Actuellement, ArcelorMittal ne possède que des sites web qui englobent l'ensemble d'informations servi pour le visiteur comme les catalogues des produits mais qui ne lui permet pas de faire l'**e-merchandising**.

La vocation de cette plateforme « **Distrishop** » est d'offrir des nouvelles fonctionnalités permettant de rendre le processus d'achat plus flexible et rapide et en respectant les exigences de base suivante :

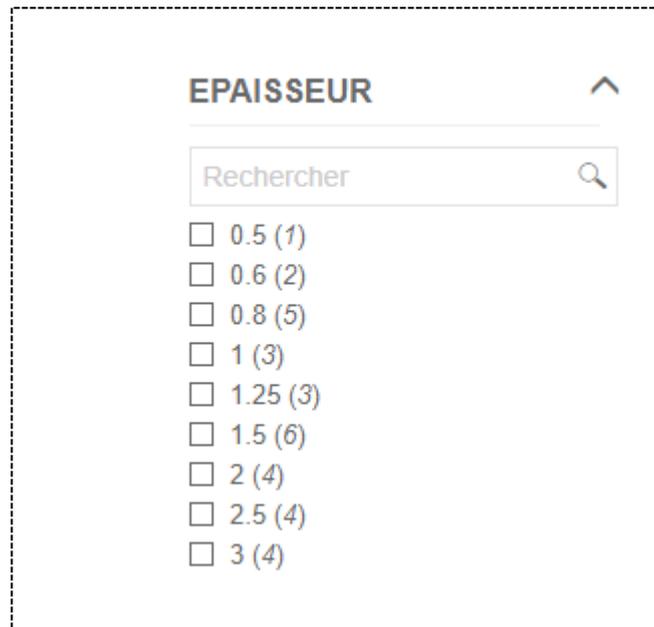
- **Autonomie** : Facilité d'utilisation courante de la solution proposée.
- **Richesse fonctionnelle** : multi sites-prix-devises, mise à jour des catalogues, E-merchandising, mise en place d'opérations commerciales, déploiement à l'international, etc...
- Fiabilité et performance de l'architecture proposée.
- Ouverture cross canal.
- **Richesse de supports** : desktop, mobile, tablette

### 1.1.2 Spécifications fonctionnelles

#### a. Gestion des catalogues et navigation

Ce module regroupe la structure des catalogues produit et la structure des catalogues CMS. Un visiteur de site non connecté peut consulter les différents catalogues, rechercher des produits et consulter les détails d'un produit en accédant au menu et ses sous-catégories. Dans les spécifications de notre site, il s'agit d'un catalogue de produits, un catalogue de contenu et trois catalogues de classifications.

La fonctionnalité de recherche est basée sur une architecture ouverte : **solRfacetsearch**. La recherche à facettes est une technique de recherche d'information correspondant à une méthodologie d'accès à l'information basée sur une classification à facettes. Les facettes sont les critères selon lesquels un utilisateur peut filtrer une collection des données, dans un site e-commerce, la liste des catalogues des produits.



*Figure 10 : La Recherche par facette*

Ce module concerne le menu de telle façon qu'il soit administrable partir du backoffice ainsi que son arborescence ne soit pas figée tout au long le cycle de vie du site.

Le menu de navigation possède trois niveaux des catégories, il représente le point d'entrée vers la page de la liste des produits de chaque catégorie (PLP).

### **b. Gestion des clients**

Cette fonctionnalité et comme l'indique son titre concerne le processus de la création d'un compte client. Hybris dans sa partie B2B distingue entre deux entités : **B2BUnit** (Client) et **B2BCustomer** (contact) de telle sorte que chaque B2BCustomer soit attaché à un B2BUnit. Notre hypothèse fonctionnelle sera qu'il existe deux types de clients :

**Client agence** : Un client agence est un client existant déjà dans le système de données de l'agence ayant un **numéro client ArcelorMittal**.

**Client non agence** : Un client non agence est un client qui n'existe pas dans la base de données de l'agence et n'ayant pas encore un **numéro client ArcelorMittal** , dans ce cas, un visiteur du site peut s'inscrire en tant que : "**Entreprise**", "**Administration**" ou bien "**Particulier**". Une fois que le client s'est inscrit, il ne peut pas se connecter au site pour effectuer une commande qu'après l'activation de son compte par un mail.

La conception technique sera développée davantage dans la partie réalisation.

### **c. Gestion des comptes clients**

Le module concerne l'espace personnel du client, changement des informations personnelles, historique des commandes, les adresses associées, les paniers sauvegardés, modification et réinitialisation du mot de passe etc... Lors de cette partie, nous avons travaillé sur la page « **Mon Compte** » qui est une page non native à Hybris. Cette page représente le point d'entrée vers l'espace personnel d'un client et elle le permet de accéder aux pages suivantes :

- **Mes coordonnées** : les informations personnelles du client
- **Mes commandes** : L'historique des commandes d'un client, il s'agit de deux types de commandes : Commandes web et commandes agence.
- **Mes paniers sauvegardés**
- **Mes planifications**
- **Mon mot de passe** : Espace pour la réinitialisation des mots de passe.
- **Mes adresses** : C'est dans cet espace que le client peut ajouter/mettre à jour ses adresses ou bien consulter son carnet d'adresse. Dans cette partie, on s'est basé sur le natif sauf la partie de la validation du code postal que je vais expliquer son implémentation dans le dernier chapitre.
- **FAQ**

### **d. Stratégie de calcul et d'affichage du prix**

Cette fonctionnalité concerne le calcul et l'affichage des prix d'un produit donné sur le site. Par défaut Hybris gère les prix à l'aide d'une **Price Factory** qui calcule le prix final après avoir appliqué un ensemble des règles liées aux taxes et remises, puis affiche le prix le plus ancien

dans la base de données, mais dans les spécifications de notre site, AMDS souhaite récupérer les prix de ses produits depuis **SAP** via un **service web** des prix et en coordination avec le service web des paniers pour afficher le prix final sur le magasin en ligne en introduisant de nouvelles règles de prix.

Citons-nous les règles importantes :

- La modification du prix depuis le backoffice sera désactivée et aussi le prix du produit, le coût de livraison, la valeur des taxes seront importés depuis **L'ERP SAP**.
- Le prix le plus élevé sera affiché pour les utilisateurs **anonymes** (non connectés) et aussi pour les utilisateurs de type « **Particulier** ».

#### **e. Stratégie d'affichage du détail d'un produit**

Un produit se distingue par ses différentes caractéristiques, dans les spécifications de notre site, le détail d'un produit doit contenir les informations suivantes :



*Figure 11 : Détails d'un produit dans le Distrishop*

Ces informations seront récupérables depuis **SAP** et doivent être administrable depuis le backoffice en termes de contenu et de format. Durant la partie mise en œuvre, je vais mettre en relief les étapes suivies pour répondre à ces exigences.

#### **f. Le processus de check-out**

Par défaut, Hybris offre un tunnel de commande permettant aux internautes de passer les commandes. Quoique le natif ne réponde pas aux exigences de notre client, ce processus sera modifié par l'équipe durant cette réalisation.

Afin qu'un client connecté puisse effectuer une commande, il doit passer par un processus de check-out contenant cinq étapes :

- ✓ **Etape 1 : Type de paiement** : il existe deux types paiements : soit par **carte** ou bien par **compte** tous les clients peuvent choisir le type "paiement par carte" par contre, le "paiement par compte" n'est valable que pour les clients AMDS ayant un numéro de client valide ainsi qu'un crédit valable sur SAP.
- ✓ **Etape 2 : Adresse de livraison** : Dans cette étape, le client doit fournir une adresse de livraison et éventuellement une adresse de facturation.
- ✓ **Etape 3 : Mode de livraison** : quatre modes de livraison seront disponibles dans le magasin en ligne : **Tournée**, **Livraison**, **Enlèvement en point relais**, **Enlèvement en agence**.
- ✓ **Etape 4 : Récapitulatif de la commande.**
- ✓ **Etape 5 : Paiement** : lors de cette étape, l'utilisateur doit accepter les termes de conditions pour passer la commande ; Lorsque l'utilisateur clique sur le bouton, il sera redirigé vers la page du fournisseur de services de paiement qui est une page externe où il peut entrer ses informations de paiement. Si le paiement est réussi, l'utilisateur sera renvoyé à la dernière étape de check- out « **Confirmation de commande** », et après avoir réussi le paiement, Hybris envoie la commande à SAP et un mail de confirmation au client.
- ✓ **Etape 6 : Confirmation de la commande.**

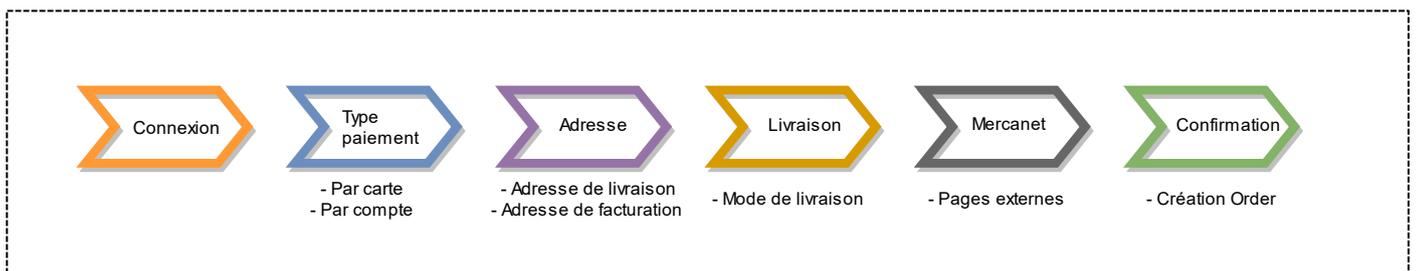


Figure 12 : Processus de check-out dans le site

Le processus de check-out a été enrichi par les fonctionnalités suivantes :

1. Une vérification de crédit dans la partie de type de paiement.
2. Changement de l'ordre des étapes de check-out.
3. Personnalisation des informations affichées dans le récapitulatif de commande.
4. Les coûts de livraison pour quelques modes de livraison sont récupérés depuis SAP via un service web.
5. Changement de traitement de l'adresse de livraison et de facturation.
6. Envoi de la commande à SAP et la personnalisation de l'email envoyé aux clients.

### **g. Gestion du contenu CMS**

Cette partie concerne la partie CMS de la plateforme, Le gestionnaire de contenu se charge de la partie commune aux pages / composants du site et génère les pages / composants à partir du texte ou des données fournies ce qui nous permet de gérer la structure du site, ajouter et classer les pages / composants et administrer leur contenu depuis le backoffice.

Hybris nous impose un certain processus à suivre pour la création d'une page ou un composant CMS administrable depuis le backoffice, je vais l'élucider au niveau de la partie mise en œuvre.

### **h. Import et conversion des medias via Cronjobs**

Hybris définit la notion de **Cronjob** pour pouvoir planifier un traitement (Job) à effectuer à une heure spécifique ou pendant une période de temps (chaque jour par exemple).

Pour notre projet AMDS j'ai implémenté deux Cronjobs pour :

**Import des images et association avec les produits** : en se basant sur un fichier CSV fournit par le client qui décrit la correspondance entre les images et les produits.

**Conversion des images** : à l'aide de l'outil **ImageMagick** qui permet de convertir une image donnée en plusieurs formats.

Les deux Cronjobs peuvent être lancé à partir du backoffice.

### **1.1.3 Analyse des besoins**

#### **a. Identification des acteurs**

Dans ce paragraphe, j'identifie les principaux acteurs de l'application :

- **Le client** : Cet acteur peut accéder au site ou à son espace personnel où il peut bénéficier de toutes les fonctionnalités du site, faire des commandes en ligne, gérer son panier, enregistrer ses commandes, etc.
- **L'administrateur** : le webmaster du site final peut accéder à l'ensemble de la plateforme notamment le backoffice du site et l'administrer.

#### **b. Diagrammes de cas d'utilisation**

Les cas d'utilisation permettent de représenter le fonctionnement du système vis-à-vis des acteurs.

Voici les principaux cas d'utilisation que j'ai pu extraire :

Use case n°1 : gestion des commandes

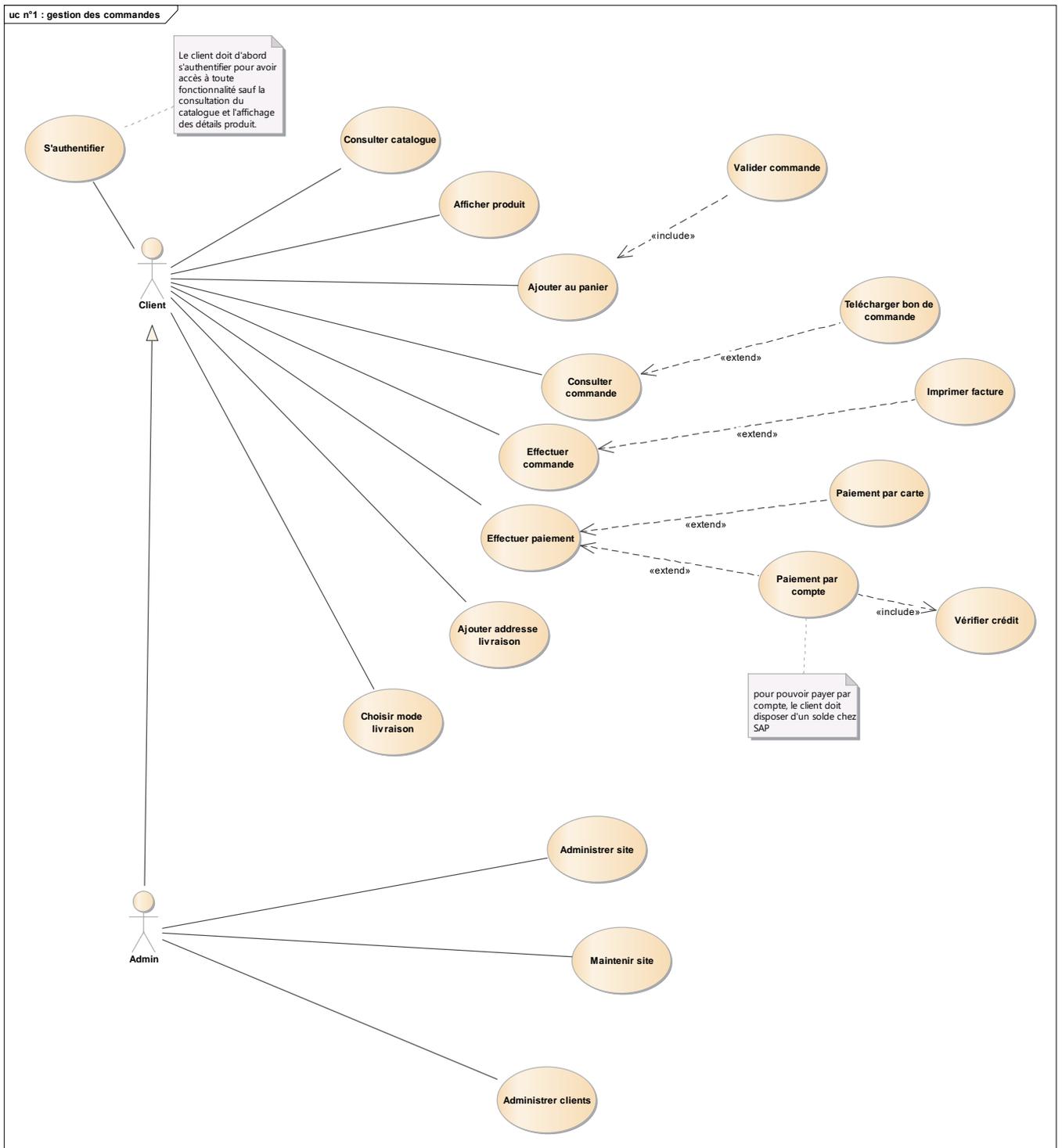


Figure 13 : Diagramme de cas d'utilisation Gestion des commandes

### c. Diagrammes d'activité

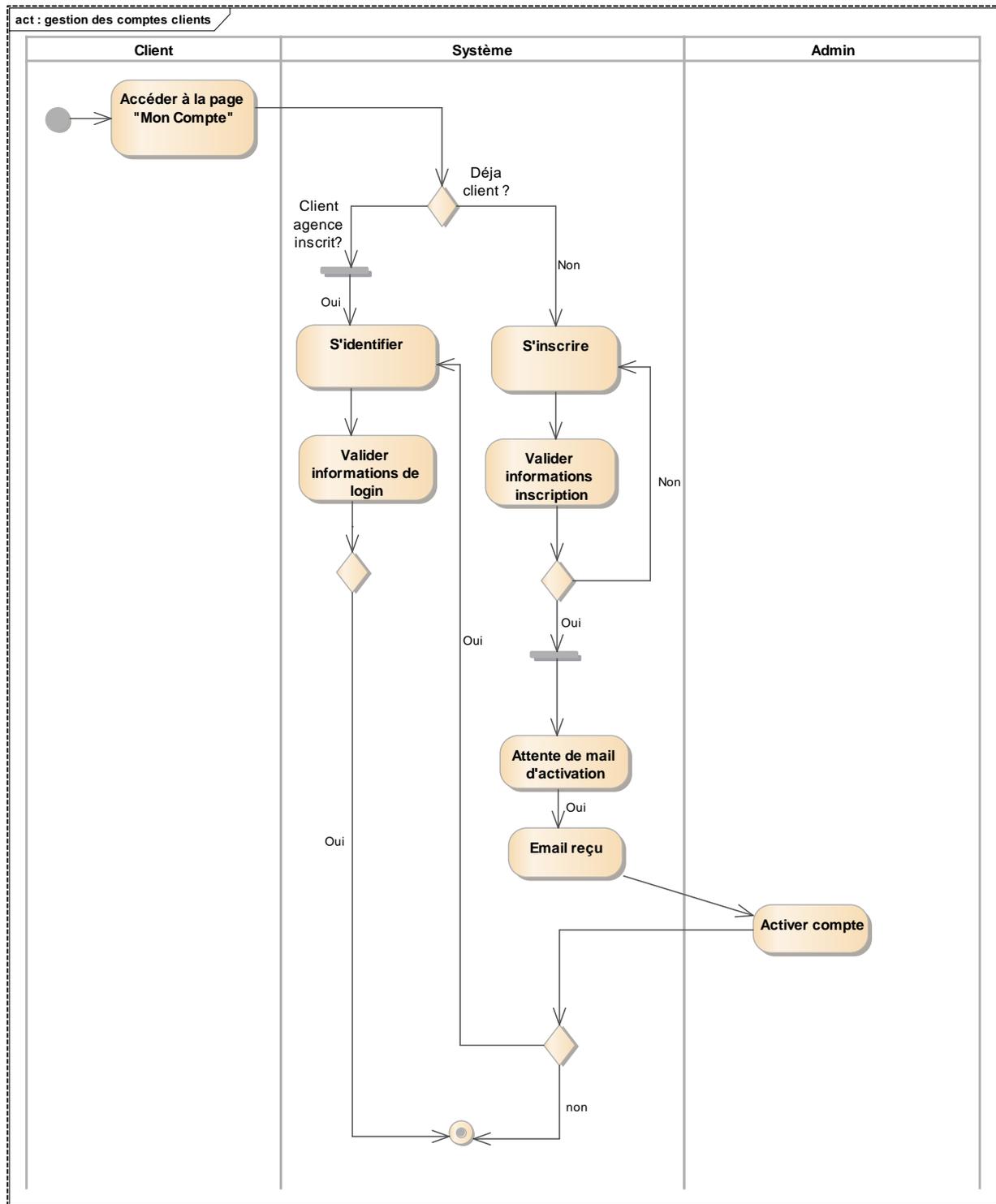


Figure 14 : Diagramme d'activité de la gestion des comptes clients

### d. Diagramme de séquence

Afin de décrire comment les éléments de système interagissent entre eux et avec les acteurs durant le processus de check-out, on présente le diagramme de séquence suivant :

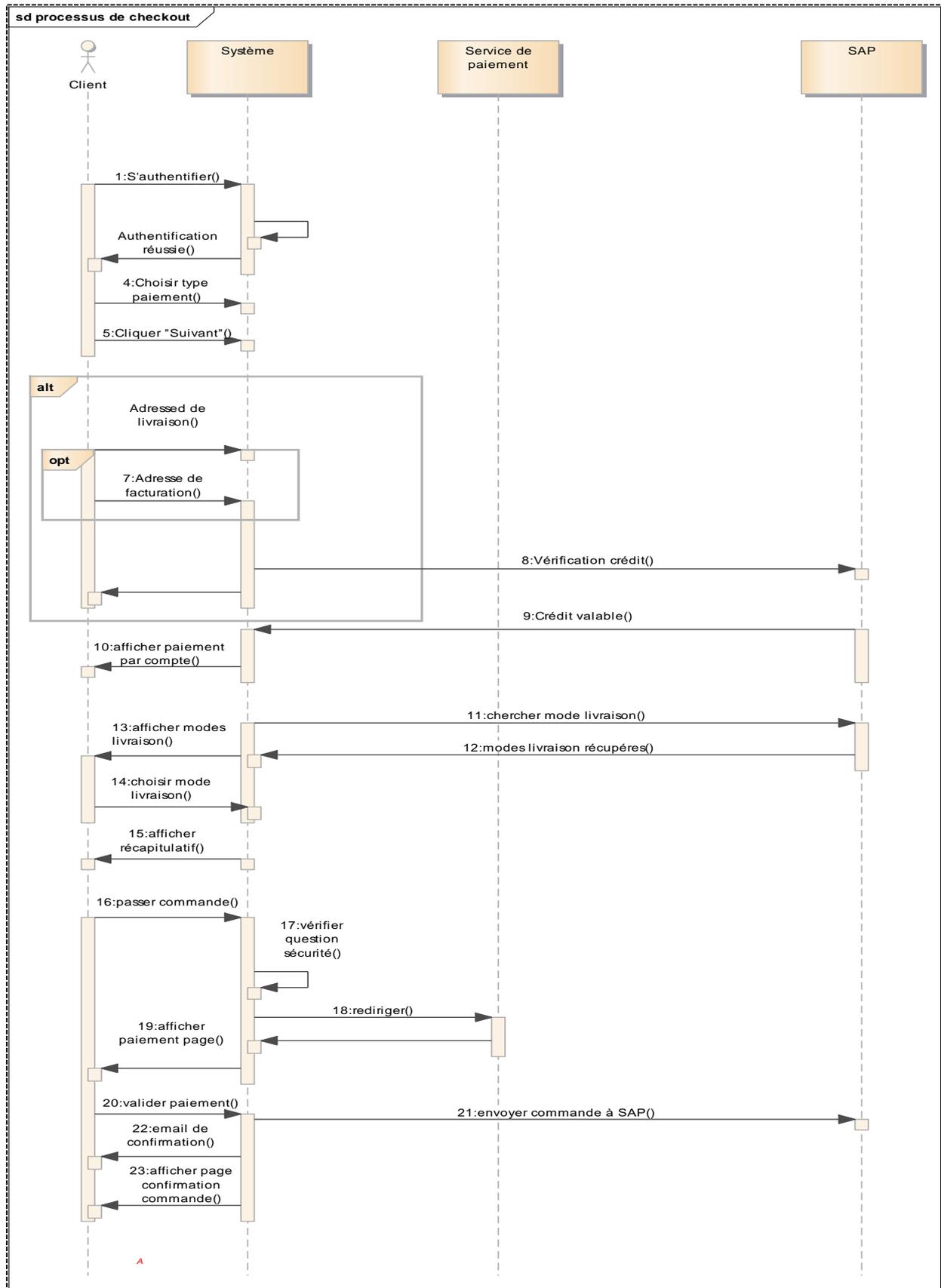


Figure 15 : Diagramme de séquence du processus checkout

### 1.1.4 Flux AMDS

Voici les principaux modules logiciels internes impliqués dans l'activité ArcelorMittal :

<b>HYBRIS</b>	<b>Plateforme qui effectue des échanges de données avec l'ensemble des systèmes ci-dessous.</b>
<b>SOLR</b>	<b>Moteur de recherche</b>
<b>MYSQL</b>	<b>Base de données</b>
<b>DATAHUB</b>	<b>Import / export des données entre Hybris et SAP</b>

*Tableau 1 : Modules logiciels internes*

<b>MERCANET</b>	<b>Plateforme de paiement en ligne</b>
<b>SAP</b>	<b>ERP qui contient l'ensemble des données du client</b>

*Tableau 2 : Services externes*

## 1.2 Projet AMC

### 1.2.1 Analyse de l'existant

La caractérisation des produits fabriqués et vendus par AMC se situe actuellement dans un système existant appelé SOFIE (Système Organisationnel Fédérateur de l'Information en Europe) développée en COBOL.

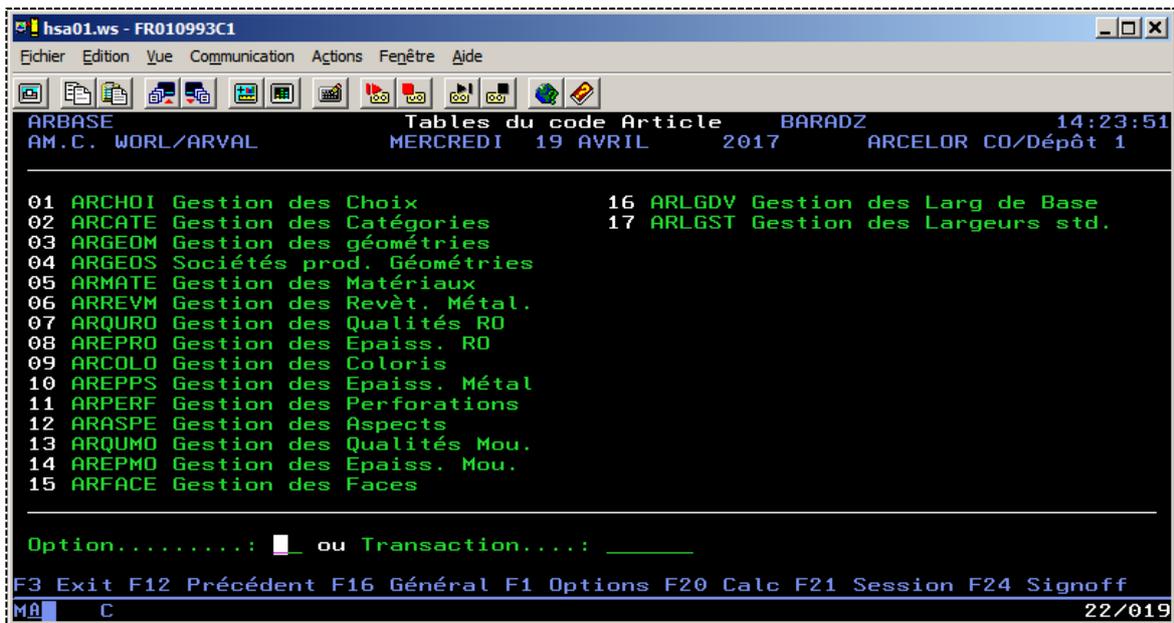


Figure 16 : Interface principale de l'application SOFIE

SOFIE contient trois types de données liées aux produits:

**Modèle de produit :** Un produit est défini par un ensemble de caractéristiques appelées aussi attributs du produit **SOFIE** : géométrie, matériau, revêtement métallique, revêtement organique, couleur, perforation, aspect, épaisseur du métal, isolation. Les produits sont configurables sur la base de ces caractéristiques.

**Tableaux de référence:** les tables qui représentent les attributs SOFIE : géométrie, matériau, revêtement métallique, revêtement organique, couleur...

**Règles de faisabilité :** La création d'un nouveau produit est soumise à des règles de vérification sur les combinaisons possibles à partir des tables de faisabilité. Une combinaison est une association de caractéristiques définissant une partie d'un produit en fonction de contraintes industrielles.

## 1.2.2 Spécifications fonctionnelles

### a. Création des tables contenant les informations SOFIE

Avoir La même architecture de la base donnée utilisée dans l'application SOFIE est parmi les fonctionnalités les plus cruciales demandées par le client vu son influence directe sur l'import des donnés.

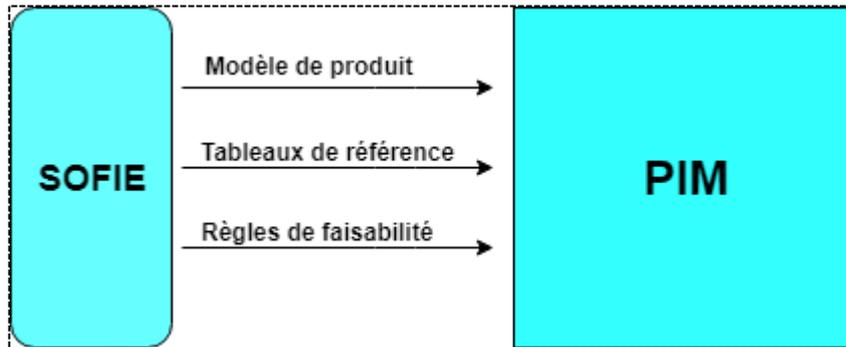


Figure 17 : Données SOFIE / PIM

La construction de cette architecture a engendré un ensemble des modifications qui seront décrit dans la partie réalisation.

### b. Configuration des rôles et droits d'accès

Pour le PIM d'AMC, le client cherche à classier leurs employés selon les profils suivants :

- Administrateurs
- Chef de produit
- Approbateur de produit

Chaque employé peut appartenir à un ou plusieurs groupes d'utilisateurs, cela dépend du profil de l'employé.

### c. Import des données

AMC cherche une solution efficace qui va permettre de copier l'ensemble des données disponibles sur SOFIE dans le PIM, pour atteindre cet objectif notre ET a proposé l'utilisation du HotFolder hybride.

Le choix de cette solution sera justifié par la suite dans la partie réalisation.

### 1.2.3 Analyse des besoins

#### a. Identification des acteurs

Comme nous avons cité précédemment trois acteurs vont avoir accès au PIM :

- a. Chef de produit
- b. Approbateur de produit
- c. L'administrateur

Ci-après la liste des acteurs et leurs permissions vis-à-vis le PIM :

Profile (Role)	Groupe d'utilisateurs	Permissions
<b>Administrateur</b>	amc- admingroup	<ul style="list-style-type: none"> <li>• Accès à toutes les données AMC</li> </ul>
<b>Chef de produit</b>	Amc - productmanagergroup	<ul style="list-style-type: none"> <li>• Gérer les attributs modifiables sur le produit</li> <li>• Gérer les médias en relation avec les produits</li> </ul>
<b>Approbateur de produit</b>	amc- productapprovergroup	<p>Possède les mêmes droits que le chef de produit</p> <p>En outre, il peut :</p> <ul style="list-style-type: none"> <li>• Gérer les catégories et l'assignation de produit</li> <li>• Approuver l'état du produit</li> </ul>

*Tableau 3 : Permissions des employés AMC*

**b. Diagrammes de cas d'utilisation**

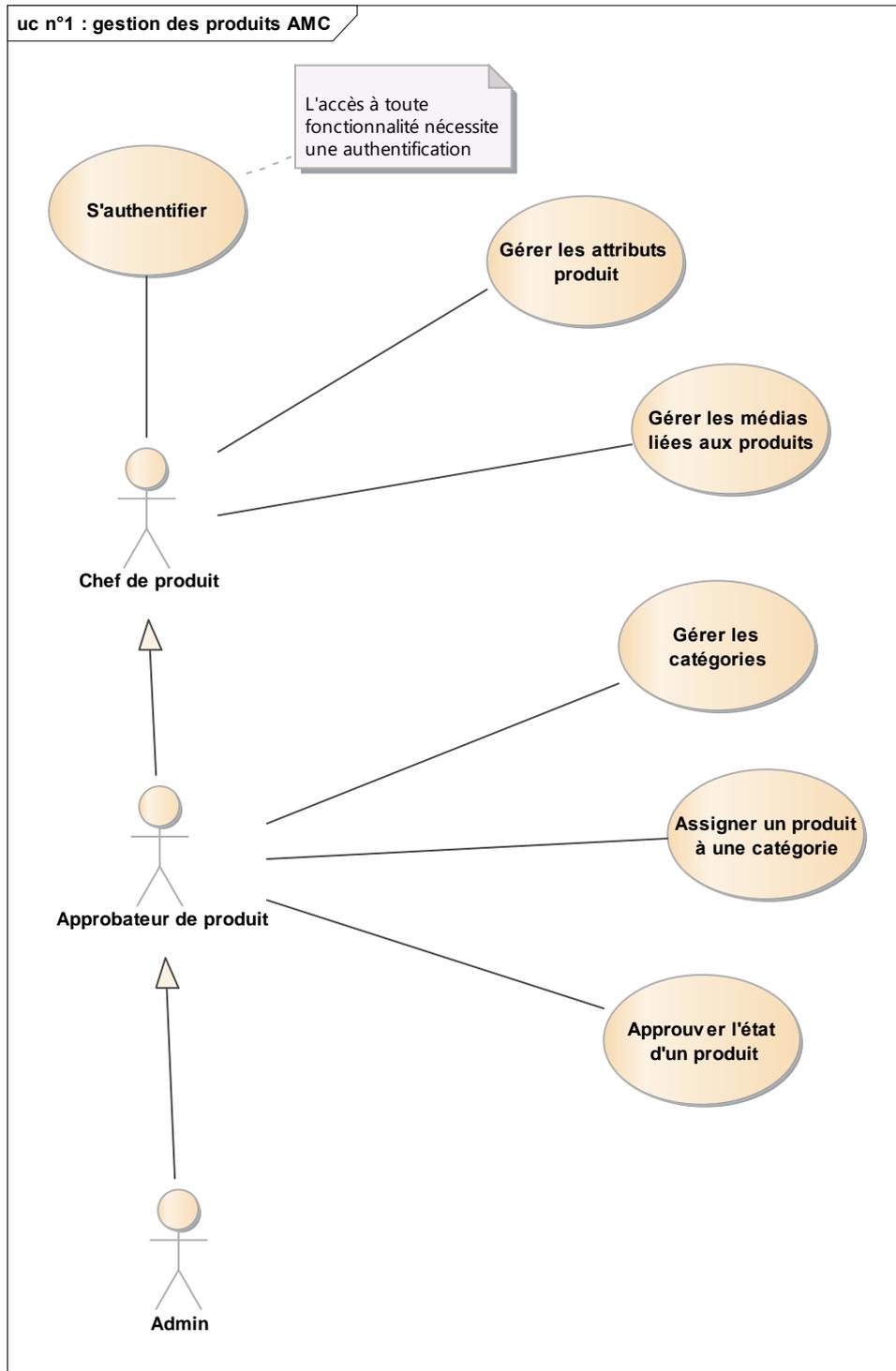


Figure 18 : Diagramme de cas d'utilisation de gestion des produits AMC

c. Diagrammes d'activité

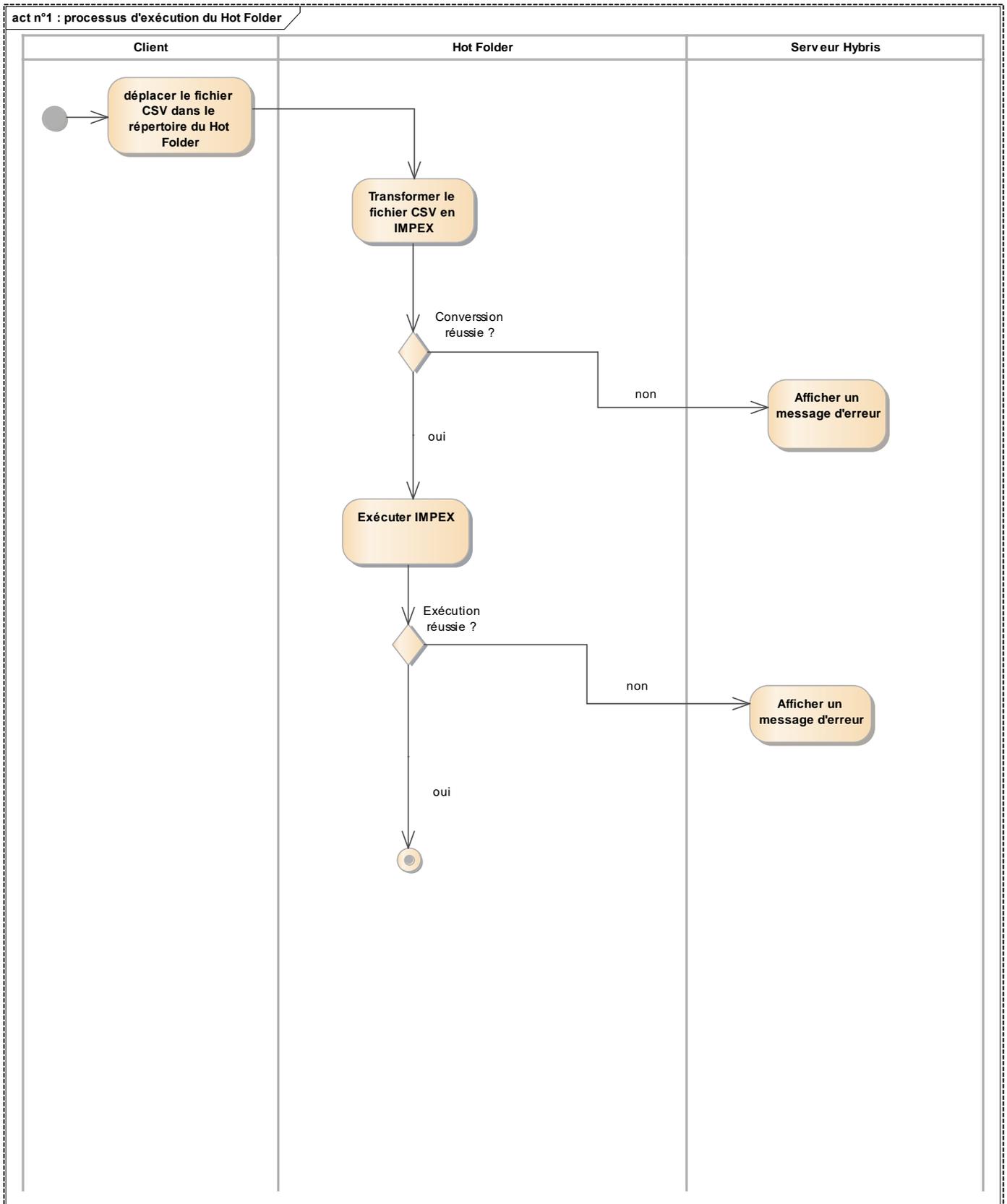


Figure 19 : Processus d'exécution du Hot Folder

## **Conclusion**

Dans ce chapitre j'ai entamé la partie importante du projet, qui est l'étude et l'analyse du projet. D'abord j'ai commencé par citer les différentes spécifications fonctionnelles, sous leurs formes User Stories. Puis, j'ai dégagé les cas d'utilisations possibles pour finir par une analyse du site en utilisant des diagrammes de séquence et d'activité basés sur les normes UML.

# Chapitre 3 : Présentation technique

Lors de ce chapitre, je vais exposer l'architecture technique et applicative adoptée durant les deux projets, puis je vais présenter la plateforme Hybris, ses fonctionnalités de base ainsi que les services qu'elle offre. Finalement, une description des outils que nous avons utilisés pour la réalisation et pour l'établissement de notre usine logicielle.

# 1 Architecture

## 1.1 Présentation de la plateforme Hybris

Dans cette partie je vais présenter de la plateforme Hybris et plus précisément la version 6.3 de la plateforme.

Classée parmi les leaders dans les rapports Gartner Magic Quadrant for Marketing Resource Management et celui de Forrester en 2016, Hybris est une solution allemande Créé en 1997 et choisie par Adidas, Reebok, Levi's, Nespresso, ainsi que plusieurs autres grandes enseignes commerciales.

Racheté début juin 2013 par SAP, Hybris s'est vu bénéficier d'un nouvel élan de développement suite au versement d'importantes sommes de fonds par les investisseurs.

Cette solution se prouve experte dans la gestion de grands catalogues et le multicanal. Offrant plusieurs « **Accelerators** » Prenant la forme de boutiques de démonstration (demonstration store), qui sont des sites clés en mains qui peuvent servir soit de démonstration ou être modifiés et adaptés aux besoins du client pour être mis en ligne en production.

## 1.2 Architecture d'Hybris Commerce

L'architecture de Hybris Commerce est flexible et modulaire, il est divisé en packages individuels, tels que le **Commerce**, le **Content**, le **Channel** et **Order**[4]. Hybris Commerce contient plusieurs couches, chacune ayant une fonction différente et un niveau d'abstraction de données.

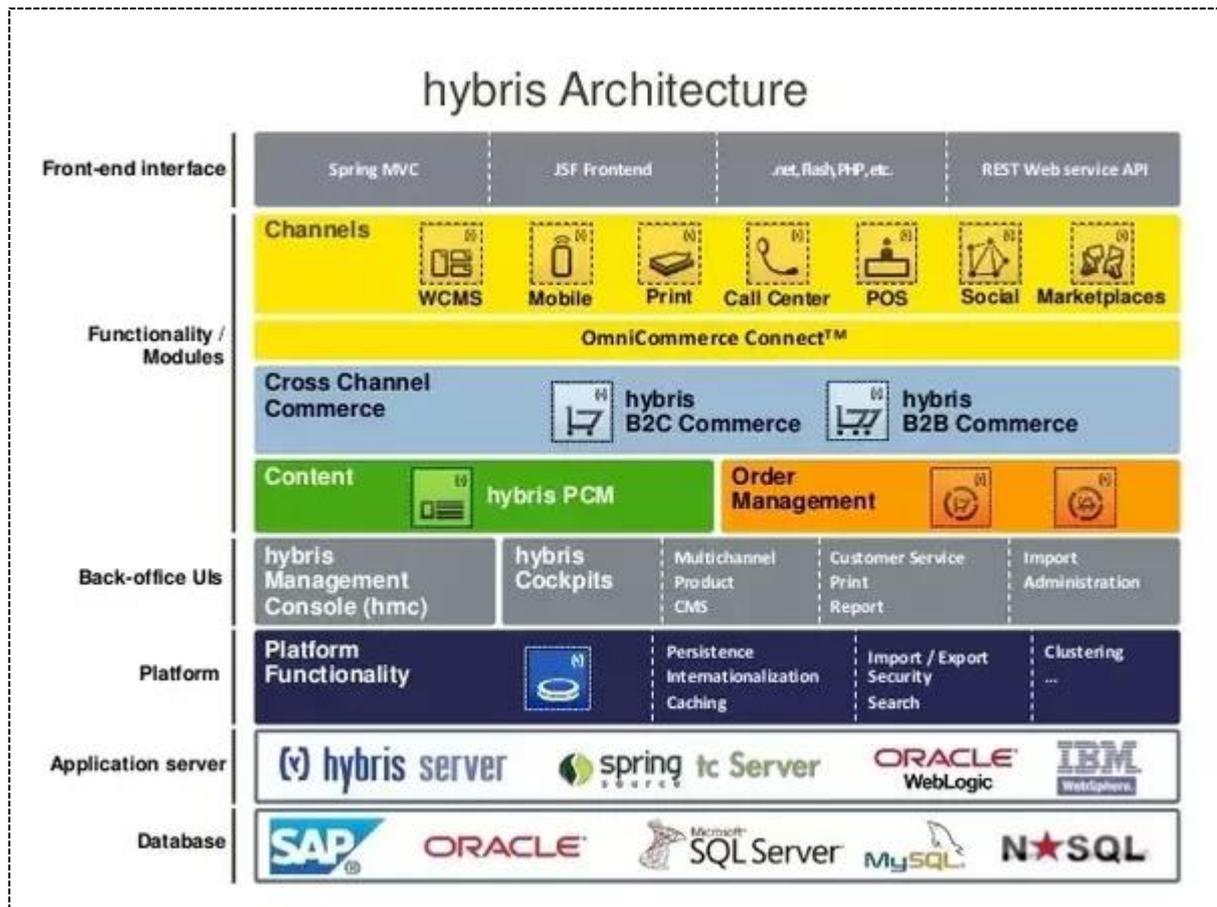


Figure 20 : Architecture d' Hybris

Pour simplifier, Hybris est structuré comme suit :

### 1.2.1 Content

Le contenu est un composant fondamental pour un site e-commerce, plus spécialement les informations liées aux produits vendus[4], notamment :

- Catalogues produits
- Systèmes de classification et Catégories de produits
- Médias (comme les images des produits etc...)

### 1.2.2 Orders

Ce sont les Commandes. C'est la couche qui contient tous les détails liés aux commandes des produits à savoir[4] :

- Promotions
- Paiement
- Gestion des inventaires
- Gestion et historique des commandes
- Détails des clients

### 1.2.3 Channel

Channel réfère à comment on veut atteindre les clients. Ceci peut être fait à la fois par des méthodes physiques ou virtuelles comme[4] :

- Web store (interfaces Desktop et Mobile)
- Magasins
- Les médias imprimés
- Emails

### 1.2.4 Commerce

Le composant Commerce apporte une collection de modules (fonctionnalités et options) dépendant du type de site commerce utilisé. Par défaut Hybris fournis deux types selon l'objectif du site :

**B2C (Business to Consumer)** : B2C est pour les entreprises désirant vendre pour des individus, un site B2C vise à donner au client une méthode facile pour trouver, connaître et acheter les produits[5].

**B2B (Business to Business)** : B2B est adopté quand une entreprise vise vendre ses produits pour d'autres entreprises. Ceci essentiellement concerne des commandes avec de très grandes quantités du même produit[5].

### 1.2.5 Platform

C'est le dernier composant et le Core de la suite, c'est là où se trouve l'interface de l'application.

Hybris représente typiquement son architecture sous le format gâteau visible sur la figure ci-dessous [4]

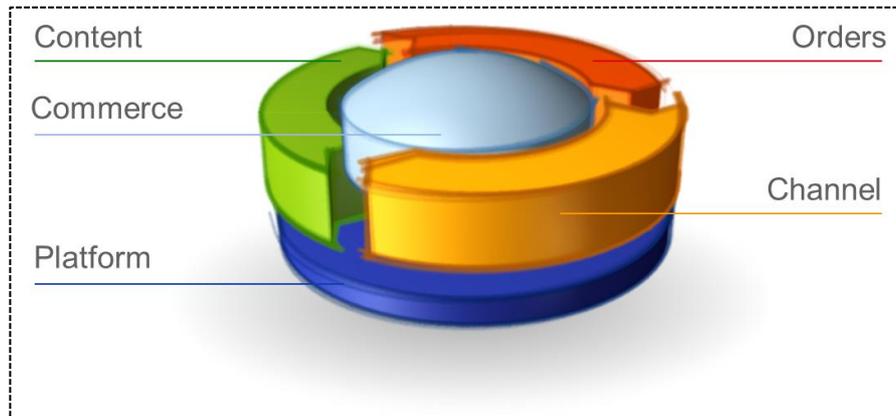


Figure 21 : Architecture générale de Hybris Commerce Suite

## 1.3 Hybris Accelerators

Comme cité dans la présentation, Hybris offre de multiples Accelerators, sous formes de sites clés en main facilement personnalisables et prêts à être déployés, que ce soit pour le B2C ou le B2B. Ces sites sont des solutions de commerce omni-canal, avec des templates de storefronts complets et les outils métiers nécessaires pour la mise en place de n'importe quel type d'utilisation e-commerce.

Plusieurs types d'Accelerators existent[5] :

- **L'Accelerator B2C Electronics** : site pour vente des produits électroniques (appareils photos etc...).
- **L'Accelerator B2C Apparel** : site pour vente des produits d'habillements et textile.
- **L'Accelerator B2C Telco** : site pour vente des produits télécommunications (abonnements téléphoniques, téléphones etc...).
- **L'Accelerator B2B Power Tools** : site pour vente des outils de travaux pour professionnels et c'est celui que nous utilisons dans ce projet. Tous ses sites sont fournis avec des données de bases pour les démonstrations et développement.

## 1.4 ImpEx

Hybris Commerce Suite est livré avec une extension d'importation / exportation appelé **ImpEx**, c'est une interface entre les fichiers CSV et la plateforme Hybris.

Cette extension permet de créer, mettre à jour, supprimer et exporter des éléments de la plateforme tel qu'un client, un produit, les données d'une commande à partir des fichiers CSV. Ces opérations peuvent être effectuées pendant l'exécution, l'initialisation ou la mise à jour du système[4].

Durant l'élaboration de ces deux projets, nous avons utilisé les fichiers impex assez fréquemment, les champs de son utilisation sont nombreux, citons-nous quelques-uns :

- Importer des données clients dans un système de production.
- Synchroniser les données avec d'autres systèmes, comme un ERP.
- Migrer des données d'une installation hybris vers une autre.

En gros, c'est une sorte d'abstraction de la base des données.

Ci-dessous on trouve un exemple de syntaxe d'un fichier impex :

```
marco
$catalogVersion=catalogVersion(Catalog(id),version)[unique=true]
Entête (Header)
INSERT_UPDATEProduct; code[unique=true]; name[lang=en]; unit(code); $catalogVersion
valeur
;W580i;Sony Ericsson W580i; pieces; Default:Staged ; iphone5;Apple iphone 5; pieces; Default:Online
Types of Modes:
INSERT Product; code[unique=true]; name[lang=en]; unit(code); $catalogVersion
UPDATE Product; code[unique=true]; name[lang=en]; unit(code); $catalogVersion
REMOVE Product; code[unique=true]; name[lang=en]; unit(code); $catalogVersion
```

Tout au long la réalisation de ce projet, nous utilisons souvent les fichiers impex vu leur importance dans la plateforme Hybris.

## 1.5 Architecture technique du Distrishop

Hybris interagit avec différents systèmes et composants selon l'architecture suivante :

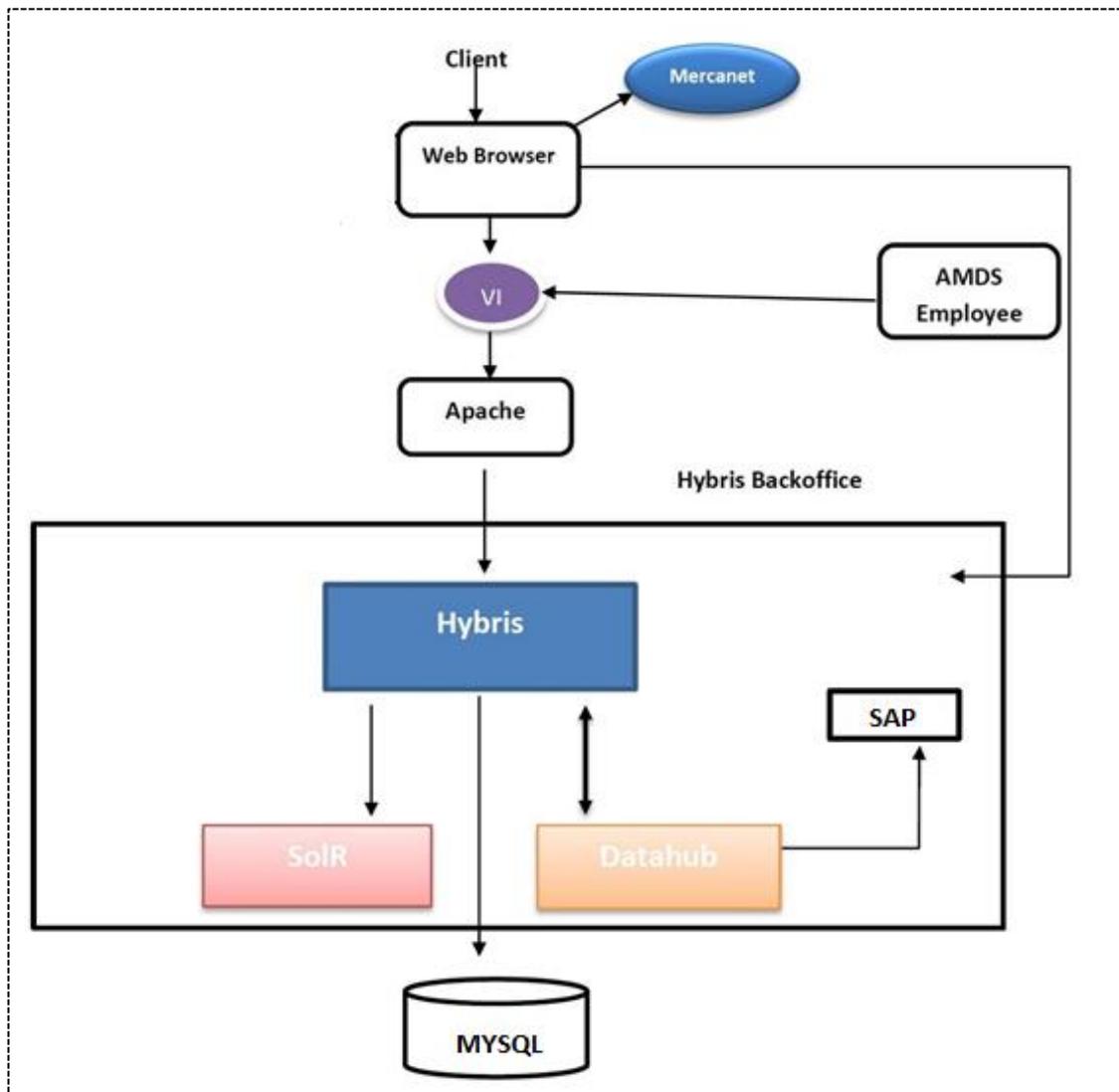


Figure 22 : Architecture technique du Distrishop

## 1.6 Architecture de développement

Le **contrôleur** récupère la **requête** HTTP envoyé par l'utilisateur, ensuite il fait appel à la **façade** pour invoquer le bon **service** qui à son tour fait appel au **DAO** pour récupérer les données d'Hybris. La façade envoie le modèle retourné par le service au **convertisseur** qui fait la

conversion du modèle au **DTO**, ce dernier va être envoyé à nouveau à la façade pour l'envoyer au contrôleur. Ce DTO est l'objet de l'affichage sur la **JSP**.

L'architecture ci-dessus permet d'avoir une indépendance entre les différentes couches, grâce à la façade qui nous permet de changer de la technologie tout en conservant le métier intact.

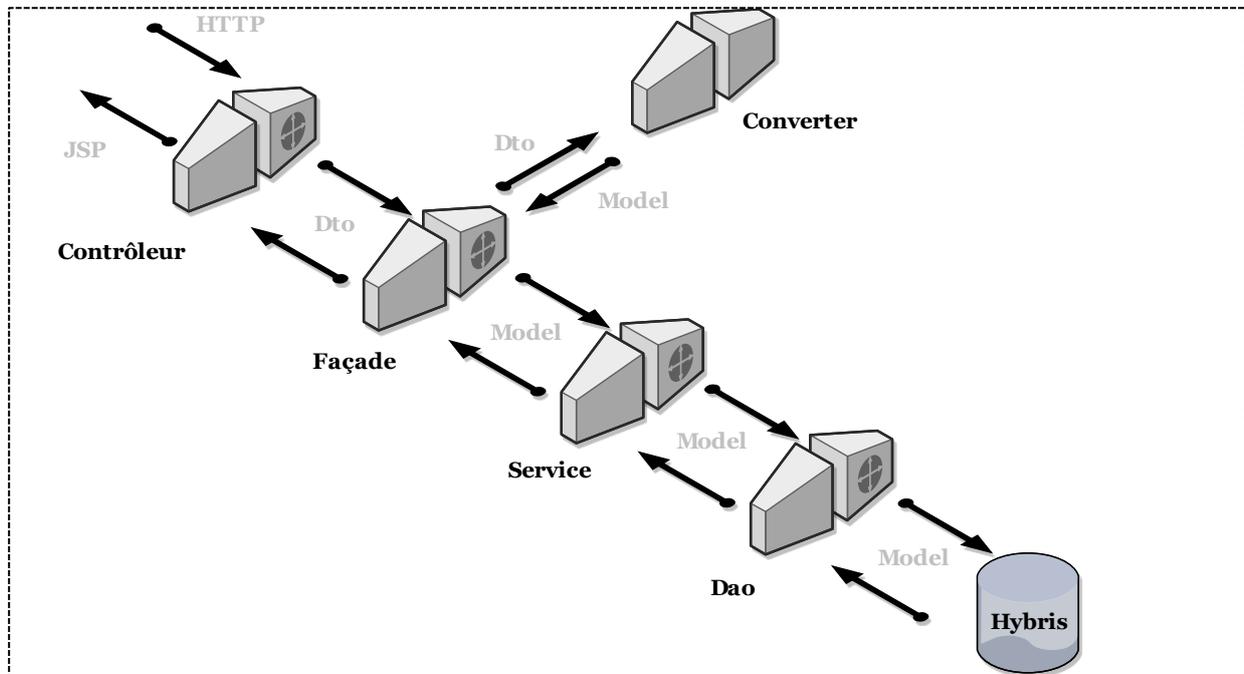


Figure 23 : Architecture de développement

## 2 Plateforme Hybris

### 2.1 Les évolutions de la version 6.3 de Hybris

#### 2.1.1 Hmc vs Backoffice

Parmi les points les plus remarquables dans la récente version de Hybris c'est l'intégration du module hMC (hybris Management Console) dans le module Backoffice ce qui engendre l'élimination de l'interface hMC.

D'abord, Le module hMC est l'interface graphique de l'utilisateur pour la plateforme Hybris.

hMC est l'outil d'administration de la suite Hybris, exécuté dans un navigateur web. En l'utilisant, on peut gérer les différents objets d'Hybris, à savoir, les produits, les données des consommateurs, les commandes, etc...

L'interface hMC contient trois domaines majeurs, comme illustré dans la figure suivante :



Figure 24 : Vue générale du hMC

Le **Toolbar hMC (1)** en haut, l'arborescence à gauche (La navigation en hMC), **Content Area (2)** au milieu et l'**Explorer Tree (3)** à gauche.

Le **Content Area** affiche l'arborescence avec des icônes larges et des descriptions de chacune des entrées, comme le montre la figure Il peut être utilisé comme une interface avec trois éléments majeurs (Search Area, Result Area, et Editor Area).

Le dernier backoffice nommé simplement « **Backoffice** » représente la troisième et dernière génération d'interface d'administration. Il vise à remplacer le hMC ainsi que les cockpits, néanmoins toutes les fonctionnalités des anciens backoffices ne sont pas encore implémentées, il garde presque la même structure d'arborescence que le hMC.

Actuellement, les objets Hybris sont administrables depuis le Backoffice comme illustré sur la figure suivante :

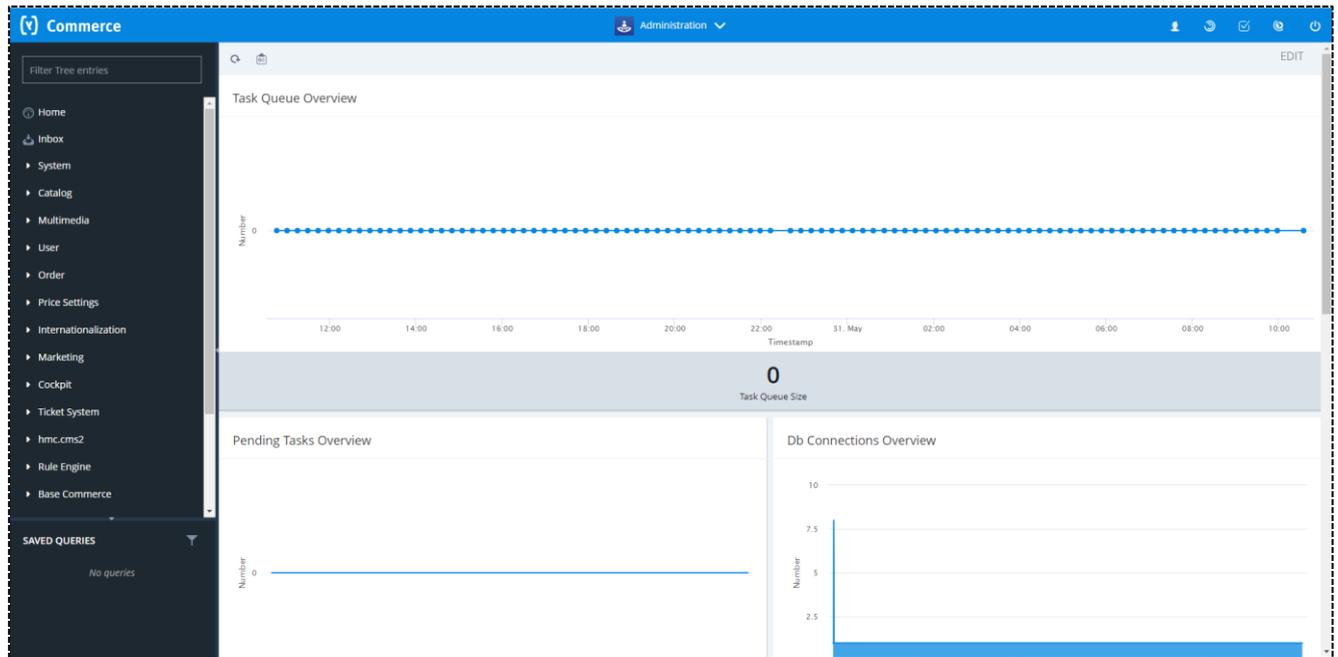


Figure 25 : Vue générale du Backoffice

### 2.1.2 Accelerator B2C versus Accelerator B2B

Dans les versions antérieures, on faisait forcément une séparation entre les deux accélérateurs B2C et B2B, mais à partir de la version 6 de Hybris, on se basait sur le contenu de la version initiale générée à base d'accélérateur B2C plus les Add-ons B2B pour avoir une solution B2B.

## 3 Outils et technologies utilisés

À part l'outil principal Hybris décrit auparavant, nous avons utilisé les outils et technologies suivantes dont quelques-unes sont intégrées dans le «out-of-the box» de la plateforme Hybris (Spring, Web services ...).

### 3.1 Environnements de développement et outils

#### 3.1.1 Eclipse

Eclipse est un environnement de développement intégré (IDE) hautement extensible. Son objectif est de produire et fournir des outils pour la réalisation de logiciels ; il englobe les activités de programmation, de modélisation, de conception, de testing, de gestion, de configuration, reporting, etc...



### 3.1.2 Jira

Comme expliqué lors du deuxième chapitre, les exigences de client seront intégrées sur JIRA afin que les développements soient surveillés en temps réel par tous les acteurs du projet.



### 3.1.3 Confluence

Confluence est un logiciel de rédaction des contenus et de partage des informations. Organisé dans des zones appelées « **spaces** » qui contiennent des pages de contact, des pièces jointes et d'autres types de contenu pouvant être visualisés et édités par les utilisateurs. Les spécifications de client sont rédigées sur Confluence, le Product Owner a accès en mode lecture et peut ajouter des commentaires sur certaines parties de la spécification pendant le processus de validation.



### 3.1.4 Git

Git est un logiciel de gestion de versions décentralisé, il ne repose pas sur un serveur centralisé. C'est un outil de bas niveau, qui se veut simple et performant, dont la principale tâche est de gérer l'évolution du contenu d'une arborescence. C'est le remplaçant de **SVN** dans la première version de la forge logicielle. Il a été adopté pour son efficacité dans le travail collaboratif, notamment par l'apport des plugins tels que git-flow qui rationalise le cycle des livraisons (releases) en utilisant des branches séparées pour le développement de fonctionnalités, la préparation des livraisons, et la maintenance. Dans notre cas, GIT est géré par le logiciel **gitLab** côté serveur et **sourceTree** côté client.



## 3.2 Frameworks utilisées

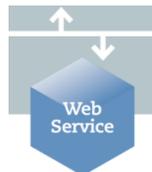
### 3.2.1 Spring

Spring est un Framework libre pour construire et définir l'infrastructure d'une application java, il facilite le développement et les tests. Il est nommé un conteneur dit " léger ", c'est à-dire une infrastructure similaire à un serveur d'applications JEE. Il prend donc en charge la création d'objets et la mise en relation d'objets par l'intermédiaire d'un fichier de configuration qui décrit les objets à fabriquer et les relations de dépendances entre eux. Le grand avantage, par rapport aux serveurs d'application, c'est qu'avec Spring, les classes n'ont pas besoin d'implémenter une quelconque interface pour être prises en charge par le Framework (au contraire des serveurs d'applications JEE).



## 3.3 Web Service

La technologie des services Web est un moyen rapide de distribution de l'information entre clients, fournisseurs, partenaires commerciaux et leurs différentes plates-formes. Les services Web sont basés sur le modèle SOA, ils communiquent via un ensemble de technologies fondamentales qui partagent une architecture commune comme HTTP, WSDL, REST, SOAP ainsi de suite ... Lors de ce projet, nous avons utilisé l'architecture REST pour la récupération des données depuis SAP, cette architecture repose sur le protocole HTTP en accédant à une ressource via différentes opérations (GET, POST, PUT, DELETE) et en se basant sur le format JSON comme format d'échange.



## **Conclusion**

Dans ce chapitre j'ai exposé l'architecture de notre site, comment nous allons séparer les différentes parties et comment elles interagissent. J'ai aussi présenté les fonctionnalités les plus importantes de la plateforme Hybris. Puis j'ai clôturé le chapitre par une présentation des technologies utilisées et l'usine logicielle que nous avons mise en place pour la réalisation des deux projets.

# Chapitre 4 : Réalisation et mise en œuvre

Dans ce chapitre, je présente la partie mise en œuvre concernant les deux projets. Dans un premier temps j'exposerai la configuration initiale en terme planification agile. On procédera ensuite à l'analyse technique de la mise en place de la solution.

# 1 Configuration initiale du projet

Un projet est entamé généralement par ce qu'on appelle dans le monde agile **sprint 0**.

Cette itération est initiée par une réunion de lancement dédiée aux travaux préparatoires du projet à savoir : la construction du **productbacklog** et la définition de l'architecture générale du projet et ainsi de suite.

## 1.1 Réunion de lancement

Lors de cette réunion, les différents rôles (**Scrum Master**, **Product Owner** et **l'équipe projet**) se réunissent pendant au moins une demi-journée pour identifier et prioriser les scénarios d'usage à inclure dans les prochains mois de travail. Ils doivent s'assurer qu'ils possèdent tous les droits nécessaires pour accéder aux différents outils de gestion de projet et de configuration et qu'ils sont suffisamment compétents pour prendre en main le projet. L'un des enjeux est de valider toutes les décisions architecturales affectant le projet. Un planning initial, qui sera ajusté au fur et à mesure de l'avancement de l'équipe, est mis en place grâce à l'outil **Jira** en s'appuyant sur la charge de chaque collaborateur et de sa disponibilité.

Voici le tableau récapitulatif des dates prévisionnelles des itérations dont lesquelles j'ai participé durant les deux projets :

	<b>Itération</b>	<b>Date de début</b>	<b>Date de fin</b>
<b>AMDS</b>	Itération 0	01 Mars	18 Mars
	Itération 1	20 Mars	05 Avril
	Itération 2	08 Avril	25 Avril
<b>AMC</b>	Itération 0	01 Mai	15 Mai
	Itération 1	17 Mai	30 Mai

*Tableau 4 : Récapitulatif des dates prévisionnelles des itérations du projet AMDS et AMC*

## 1.2 Vérification de l'environnement de travail

Avant de se lancer dans le processus de développement, il faut s'assurer que l'environnement de développement est bien stable chez tous les membres d'équipe. Pour notre projet, l'environnement portable était préparé par l'expert technique.

### 1.3 Découpage des fonctionnalités par itération

Le sprint backlog construit au début de la réunion du planning contient l'ensemble des tâches sélectionnées du product backlog. Nous avons réparti les fonctionnalités par sprint comme montre le tableau suivant :

		Fonctionnalités développées	User Story includes	Version Hybris
AMDS	Itération 1	Page d'accueil, Catalog & Navigation, Inscription	41	6.3.0.1
	Itération 2	Login, Mon compte, Pricing, Processus du check-out, CMS content	60	6.3.0.4
AMC	Itération 1	Création des tables contenant les informations SOFIE, Configuration HotFolder, configuration des droits d'accès	10	6.5.0.1

Tableau 5 : Découpage des fonctionnalités par sprint

## 2 Mise en œuvre

### 2.1 Projet AMDS

#### 2.1.1 Gestion des catalogues et navigation

Cette partie est découpée en deux, la première concerne la création des catalogues selon le besoin de notre client et la deuxième concerne tout ce qui est navigation dans le site depuis la page d'accueil (menu et bas de page)[6].

Pour les catalogues, on a créé trois types de catalogues : catalogue des produits (**ProductCatalog**), catalogue de contenu (**ContentCatalog**) et quatre catalogues de classification, ce dernier contient les attributs de classifications utilisés pour configurer les facettes. L'administrateur du site peut les trouver dans le backoffice comme suit :

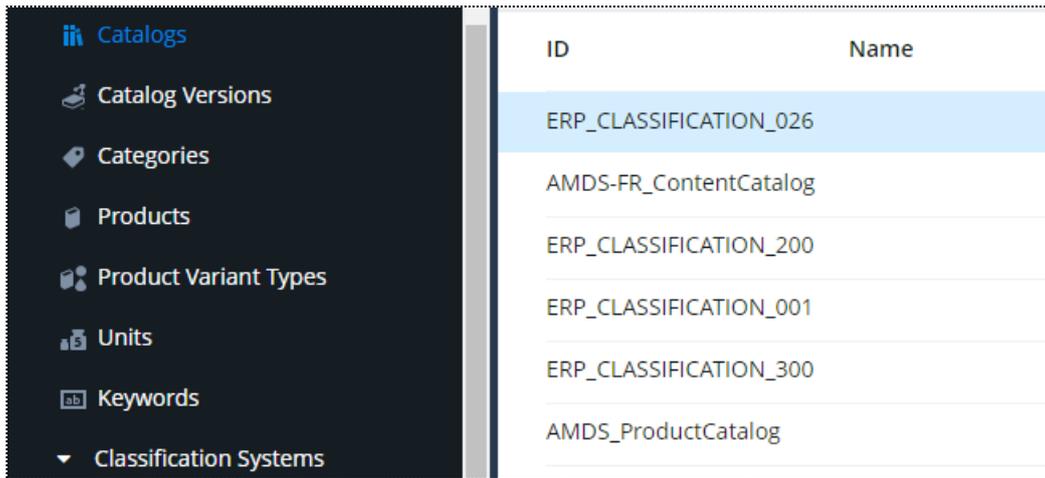


Figure 26 : Les catalogues du site AMDS via Backoffice

Pour la partie navigation, j'ai construit un menu de navigation hiérarchique et administrable contenant les catégories des produits comme dans la figure ci-dessous :

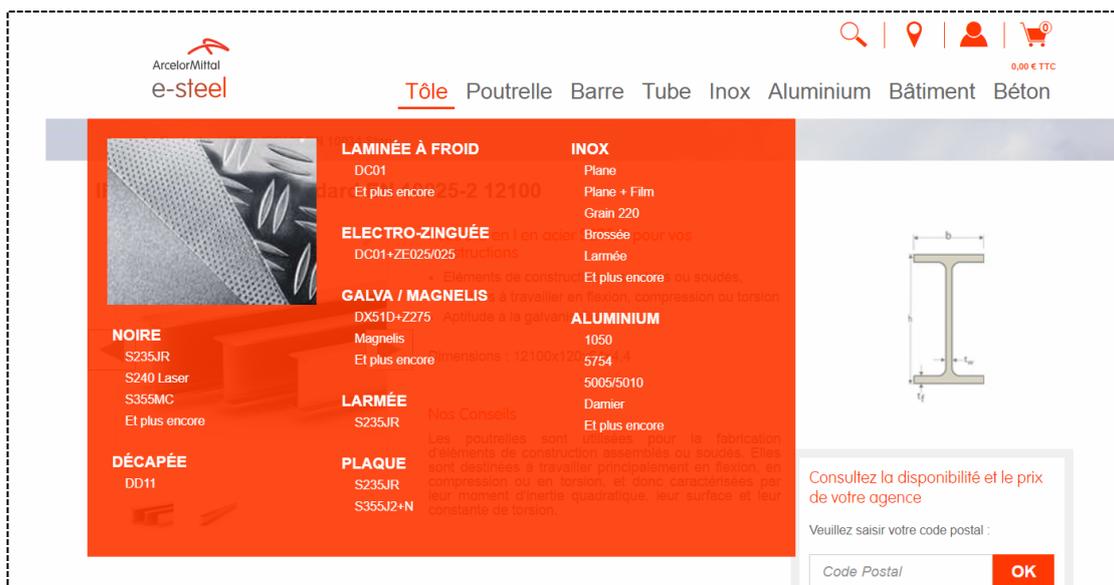


Figure 27 : Menu de navigation du site

Quant au bas de page, on l'avait décomposé en quatre parties selon les spécifications du client. Nous allons définir les défis que nous avons rencontrés lors de la construction du bas de page et par la suite mettre en œuvre la solution adoptée.



Figure 28 : Bas de page du site

### a. Problématique

L'accélérateur **PoweTools** sur lequel nous nous sommes basés ne possède que **la partie (2)** de la figure comme partie administrable du composant de bas de page, ce qui contredit avec ce qui était demandé : **les parties (1), (3) et (4)** doivent aussi être des composants administrables CMS du bas de page.

### b. Solution

Pour remédier à ce problème, j'ai modifié le modèle du composant **footerNavigationComponent** en ajoutant la partie (1) en tant que **liste des bannières**, la partie (3) comme un **paragraphe CMS** et la partie (4) en tant qu'un composant **CMS des liens**.

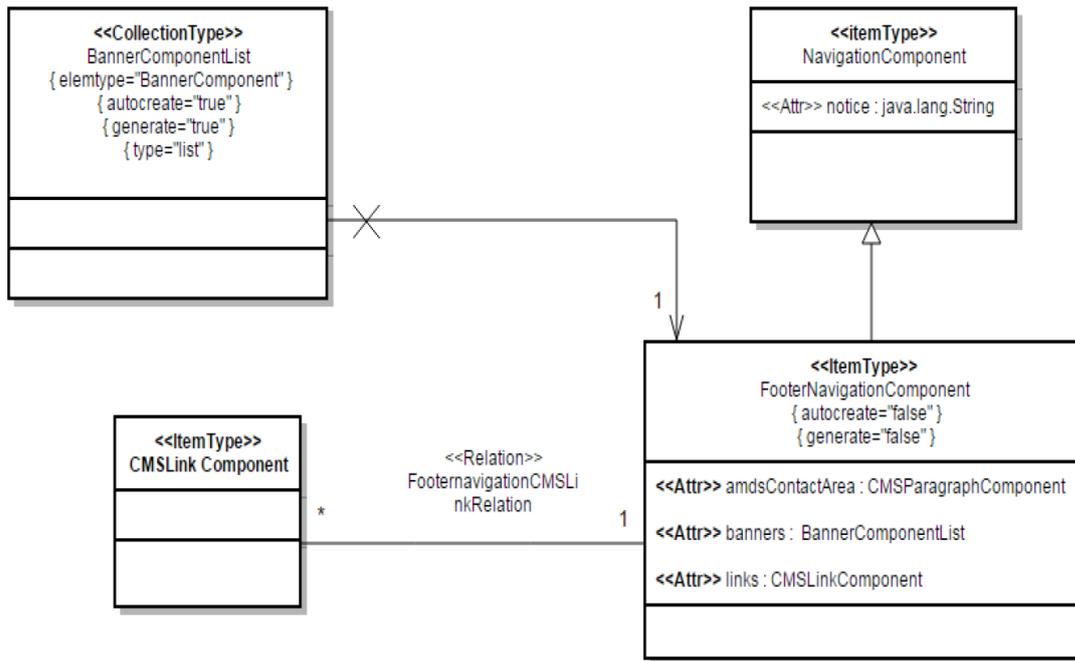


Figure 29 : Diagramme de classe du bas de page

### 2.1.2 Gestion des utilisateurs

Lors de l'inscription d'un client, il choisit entre deux modes d'inscription selon son type client **agence** ou client **non agence**. Dans cette partie, nous avons introduit à Hybris ce nouveau traitement de mode d'inscription qui fait partie du module B2BRegistration, nous allons nous focaliser sur la partie client non agence que j'ai développée moi-même [6].

#### a. Problématique

Techniquement, la différence entre les deux modes d'inscription sera dans le fait que lors de l'inscription client agence on se contente par la création/mise à jour d'un contact (**B2BCustomer**) et l'attacher à un **client (B2BUnit)** qui existe déjà. Par contre, lors de l'inscription d'un client non agence, on procède carrément par la création d'un nouveau client (**B2BUnit**) et l'attacher au contact crée, chose qui demande un certain privilège pour raison de sécurité.

#### b. Solution

La création d'un nouveau **B2BUnit** nous a imposé la création d'une session **Administrateur** temporaire en se servant d'une classe interne anonyme et implémenter dedans notre code de la création du client. J'ai aussi modifié le modèle de B2Bunit et B2BCustomer pour ajouter les attributs qui les caractérisent. Le code implémenté suit l'algorithme suivant :

D’abord on crée un nouveau client avec **un numéro client ArcelorMittal** (ce champ peut être vide lors de la création car il est supposé que le client non agence ne possède pas encore un numéro client ArcelorMittal) et un **numéro de TVA intracommunautaire**. Ensuite, on crée un nouveau contact puis on attache le nouveau contact au nouveau client et on empêche le nouveau contact de s’identifier dans le site web qu’après l’activation de son compte via un email de confirmation. (Voir tableau 6).

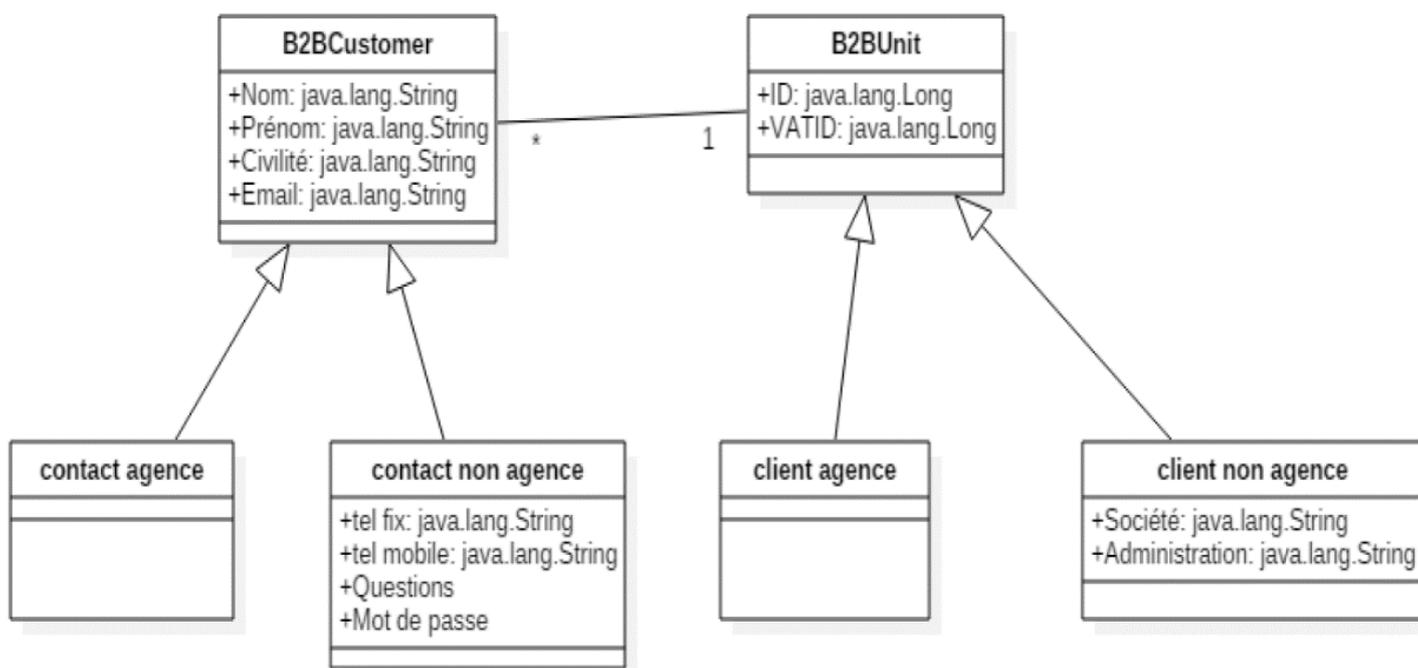


Figure 30 : Diagramme de classe de l’inscription - client non agence

Hybris	Spécifications
B2BCustomer	Contact
B2BUnit	Client
ID	N° Client ArcelorMittal
VATID	N° TVA intracommunautaire

Tableau 6 : Mapping des termes entre Hybris et les specs

### 2.1.3 Gestion des comptes clients

Cette partie comme évoqué lors du deuxième chapitre contient l’espace personnel du client, nous avons travaillé sur trois parties importantes : **la validation du poste code de l’adresse**,

**administrer les blocs « nos conseils »** qui est inclus dans toutes les pages de mon compte et finalement le traitement du **téléchargement du bon de commande** dans la page historique des commandes d'un client pendant laquelle nous avons procédé par la construction d'un document PDF qui contient les détails de la commande en question[6].

### a. Problématique

Concernant la validation du code postal, le client voulait le valider avec une liste des codes postaux importés depuis le backoffice.

### b. Solution

Pour réaliser cette tâche j'ai créé un nouveau modèle « **AMDSZipCodeModel** ».

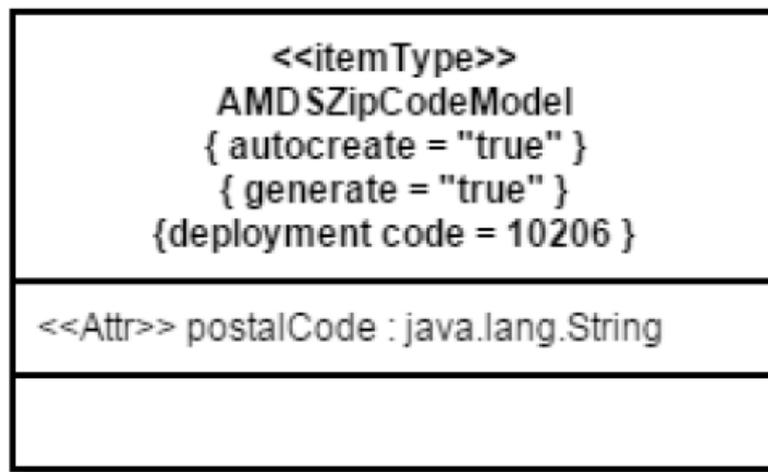


Figure 31 : La classe Zip Code Model

Ce modèle sera alimenté par une liste des codes postaux valides depuis un fichier **impex** ce qui fait le client lors de l'ajout d'une adresse, il doit entrer l'une des valeurs existantes dans la liste pour qu'elle soit valide.

#### 2.1.4 Stratégie d'affichage du détail d'un produit

Un produit est toujours caractérisé par des spécifications informatives, notamment, **les conseils**, **les spécifications techniques** et **la description détaillée**. Ceci fera l'objet de cette partie.

Accueil / Poutrelle / IPE / IPE120 EN 10034 Stan...

### IPE120 EN 10034 Standard EN 10025-2 12100

En stock

**Poutrelle en I en acier S275JR pour vos constructions**

- Éléments de construction assemblés ou soudés, destinés à travailler en flexion, compression ou torsion
- Aptitude à la galvanisation

Dimensions : 12100x120x64x4,4

**Nos Conseils**

Les poutrelles sont utilisées pour la fabrication d'éléments de construction assemblés ou soudés. Elles sont destinées à travailler principalement en flexion, en compression ou en torsion, et donc caractérisées par leur moment d'inertie quadratique, leur surface et leur constante de torsion.

Quantité  Prix unitaire\*

**AJOUTER AU PANIER**

\* Prix dégressif en fonction du poids total de votre panier.

\* Connectez-vous pour voir vos tarifs personnalisés

Détails produit    Spéc. techniques    Avis

Les poutrelles sont des produits sidérurgiques laminés à chaud en forme de I, H ou U. Elaborées avec un taux de silicium compris entre 0,15% et 0,25%, elles garantissent une aptitude à la galvanisation. On appelle la partie centrale de la poutrelle « l'âme » tandis que ses côtés sont appelés « ailes » ou « semelles ». Les poutrelles, quelque soit leur forme (I, H ou U), présentent les caractéristiques communes suivantes :

- une hauteur supérieure ou égale à 80 mm
- des ailes de largeur égale et normalement symétriques
- les faces extérieures des ailes sont parallèles
- les ailes sont soit d'épaisseur décroissante de l'âme vers le bord (profilés dits « à ailes inclinées ») soit d'épaisseur constante (profilés dits « à ailes parallèles »).
- la surface des âmes est raccordée par des congés aux faces intérieures des ailes. Des zones grises, marbrées ou rugueuses peuvent apparaître mais n'ont aucune conséquence sur la tenue à la corrosion. La poutrelle est caractérisée par les éléments dimensionnels suivants :
- Les largeurs et épaisseurs de l'âme et des ailes
- Le rayon entre l'aile et l'âme

Dimensions : 12100x120x64x4,4

Figure 32 : Page détail d'un produit

### a. Problématique

Ces informations affichées doivent être **dynamiques, administrables** en termes de **contenu** et de **forme** au niveau du backoffice.

### b. Solution

Pour répondre à ce besoin, j'ai utilisé la notion des **populateurs** (populators) et des **convertisseurs** (converters).

Les populateurs servent à transmettre les données depuis le **modèle** vers la **façade**, ils sont utilisés dans le cas où on a besoin d'afficher une information existante dans la base des données (couche présentation), Quant aux convertisseurs, ils ont le rôle de parcourir une liste des populateurs.

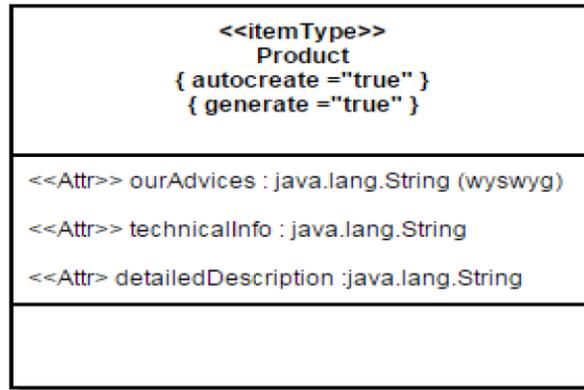


Figure 33 : Nouveaux attributs du ProductModel

Le schéma suivant explique comment les informations contenues dans les attributs créés ci-dessus vont être affichés depuis la base des données (Model) :

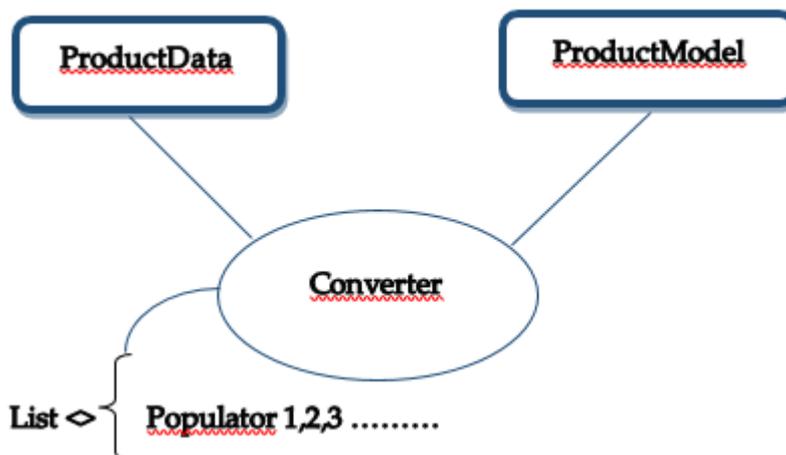


Figure 34 : Schéma expliquant l'opération de data populating

### 2.1.5 Le processus de check-out

Cette fonctionnalité est l'une des fonctionnalités les plus cruciales dans le Distrishop car elle contient le processus de la passation d'une commande.

#### a. Problématique

Le défi que nous avons rencontré pendant la mise en œuvre de cette partie est d'une part, celui des régressions suite à la modification des données de « **PowerTools** » lors des

itérations précédentes ce qui nous a empêchés de passer une commande avec succès. D'une autre part, l'éloignement des spécifications du client de ce qu'offre l'accélérateur d'Hybris.

## b. Solution

Pour remédier à ce problème, nous avons récupéré les données de l'accélérateur **PowerTools** d'une façon temporaire.

Accueil / Finaliser ma commande / Types de paiement

◀ Modifier mon panier

**Finaliser ma commande** 9 articles - 1 892,92 €\*

**1. Types de paiement**

Définissez une référence de commande personnelle (facultatif)

Paiement par carte bancaire

Vous serez redirigé vers l'interface de paiement lors de la confirmation de votre commande.

**SUIVANT**

2. Adresse

3. Livraison

4. Récapitulatif

5. Payer la commande

Produit	Quantité	Total
1. Tube carré soudé 50x3m m acier inoxydable 1.4301 EN 10088-1 6020	9	1 577,43 €
<b>Sous-total</b>	<b>9</b>	<b>1 577,43 €</b>
<small>Sous-total HT</small>		<small>1 577,43 €</small>
<small>T.V.A (20%)</small>		<small>315,49 €</small>
<b>Montant total TTC</b>		<b>1 892,92 €* </b>
<small>sans frais de livraison</small>		

Figure 35 : Les étapes de la création d'une commande dans le Distrishop

Un client bloqué n'a pas le droit de passer une commande, s'il se trouve sur la page « **panier** », le bouton « **passer commande** » sera bloqué et s'il se trouve sur la page « **check-out** » il sera redirigé vers la page « **panier** » avec un message d'erreur affiché.

Pour la réalisation de cette tâche, j'ai modifié le modèle du **B2Bunit** en ajoutant un attribut contenant l'information de **l'état du client**. Cette information est récupérée depuis SAP et elle peut être administrable depuis le backoffice.

```

<<itemType>>
  B2BUnit
  { autocreate = "true" }
  { generate = "true " }

<<Attr>> SAPBlockedContact : java.lang.Boolean
    
```

Figure 36 : Classe B2BUnit Model modifiée

Je recherche...

ArcelorMittal e-steel

Tôle Poutrelle Barre Tube Inox Aluminium Bâtiment Béton

Accueil / Mon panier

Votre commande ne peut pas être finalisée, veuillez vous rapprocher de votre agence.

Continuer mes achats

Mon panier | 9 articles - 1 892,92 € TTC

Article	Prix unitaire	Quantité	Total HT
1. Tube carré soudé 50x3mm acier inoxydable 1.4301 EN 10088-1 6020 00000000002121979 6020x50x3	175,27 €	9	1 577,43 €
<b>Sous-total</b>		<b>9</b>	<b>1 577,43 €</b>

**A noter**

Nous livrons\* sur la région de Toulouse (départements 31 / 32 / 81 / 82 / 09 / 11 / 66 / 12 / 46).

\*selon conditions de livraison

**Total de mon panier**

Sous-total 1 577,43 €

Un code promo ?  OK

Sous-total HT 1 577,43 €  
T.V.A (20%) 315,49 €

**Montant total TTC 1 892,92 €**  
sans frais de livraison

SAUVEGARDER MON PANIER

IMPRIMER MON PANIER

PASSER COMMANDE

Figure 37 : Client AMDS bloqué

Après avoir effectué le paiement, le client est redirigé vers la page de confirmation de la commande où il va trouver affichées les informations de la commande qu'il vient d'effectuer selon le type de paiement choisi dans la première étape du check-out.

### 2.1.6 Gestion du contenu CMS

La partie CMS prend une partie cruciale dans ce projet, nous avons procédé par la création d'un ensemble des pages CMS qui sont administrable ainsi que leur contenu depuis le backoffice[6], nous nous sommes basés sur une **template** créée par des fichiers **impex**.

En général, on suit le processus suivant pour qu'on obtienne le résultat souhaité :

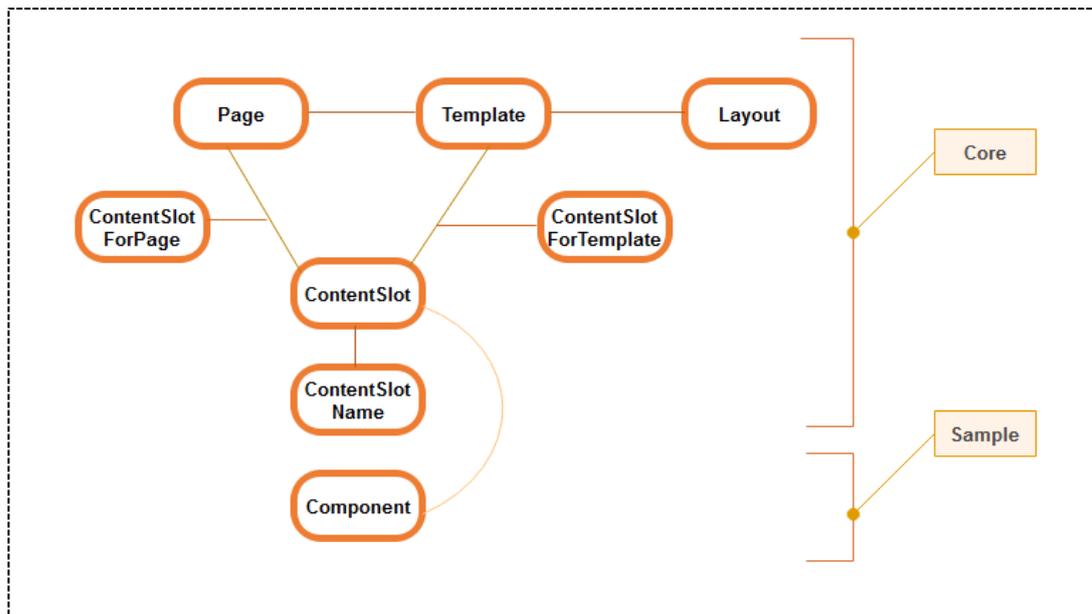


Figure 38 : Processus de la création d'une page CMS

Une page CMS dans hybris est composée d'un modèle de pages (**template**) qui définit un certain nombre de fentes (**slots**) permettant de positionner les composants placés sur la page.

Le core représente le noyau d'Hybris, c'est dans ce niveau que nous créons les « **templates** » et les attacher à des « **layout** », ces « **layout** » sont des pages JSP spéciales qui contiennent les différentes positions qu'une « **template** » doit posséder, ces positions sont définies aussi dans le fichier **.vm** de cette « **template** ».

Les fichiers **\*.vm** sont des fichiers de la **Velocity Template Engine**, et comme l'indique son nom, c'est un moteur qui génère des templates et qui assure la séparation entre la couche présentation et la couche métier.

Après avoir introduit notre template, on crée les différentes parties de la template qui vont contenir nos composants, on les appelle des « **content slot** ». Ces « **content slot** » vont être attaché aux positions définies tout à l'heure.

### 2.1.7 Gestion des prix

Bien qu'Hybris offre un module de Pricing qui se charge de la gestion des prix, nous nous sommes basés lors de ce projet sur les prix importés depuis SAP en utilisant les services web. Pour calculer le prix final, deux services web nous ont servi à savoir : service web du prix (Price WS) et service web du panier (Cart WS).

Accueil / Mon panier

◀ Continuer mes achats

**Mon panier** | 9 articles - 1 892,92 € TTC

Article	Prix unitaire	Quantité	Total HT
1. Tube carré soudé 50x3mm acier inoxydable 1.4301 EN 10088-1 6020 00000000002121979 6020x50x3	175,27 €	9	1 577,43 €
<b>Sous-total</b>		<b>9</b>	<b>1 577,43 €</b>

**A noter**

Nous livrons\* sur la région de Toulouse (départements 31 / 32 / 81 / 82 / 09 / 11 / 66 / 12 / 46).

\*selon conditions de livraison

**Total de mon panier**

Sous-total 1 577,43 €

Un code promo ?  OK

Sous-total HT 1 577,43 €  
T.V.A (20%) 315,49 €

**Montant total TTC** 1 892,92 €  
*sans frais de livraison*

SAUVEGARDER MON PANIER

IMPRIMER MON PANIER

PASSER COMMANDE

Figure 39 : Page panier du Distrishop

### 2.1.8 Import et conversion des medias via Cronjobs

Cette fonctionnalité doit permettre au client d'ajouter ou bien modifier les images des produits, ces images doivent être disponibles dans plusieurs formats pour pouvoir les utiliser dans différents pages.

Pour atteindre cet objectif j'ai développé deux **jobs** (services), le premier est celui qui permet de faire l'import des médias ainsi que l'association avec leurs produits, le deuxième convertit un ensemble des images données en plusieurs formats déjà prédéfinis.

Les deux jobs sont rattachés à deux Cronjobs différents qui peuvent être lancés depuis le backoffice.

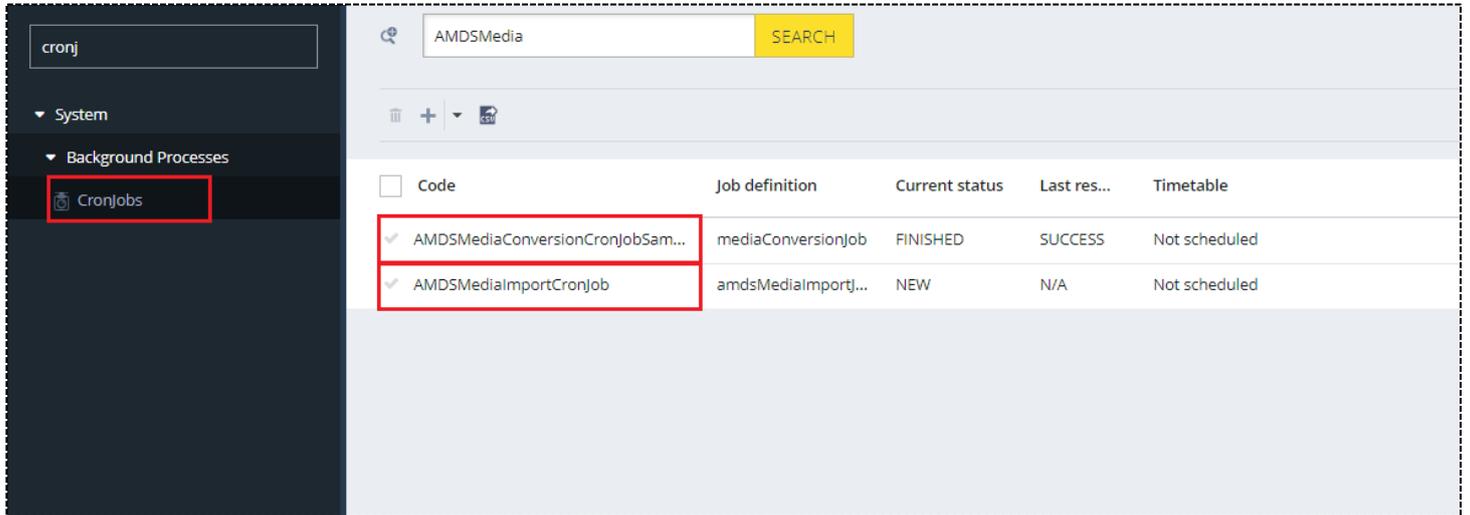


Figure 40 : Les deux cronjobs pour l'import et la conversion des médias

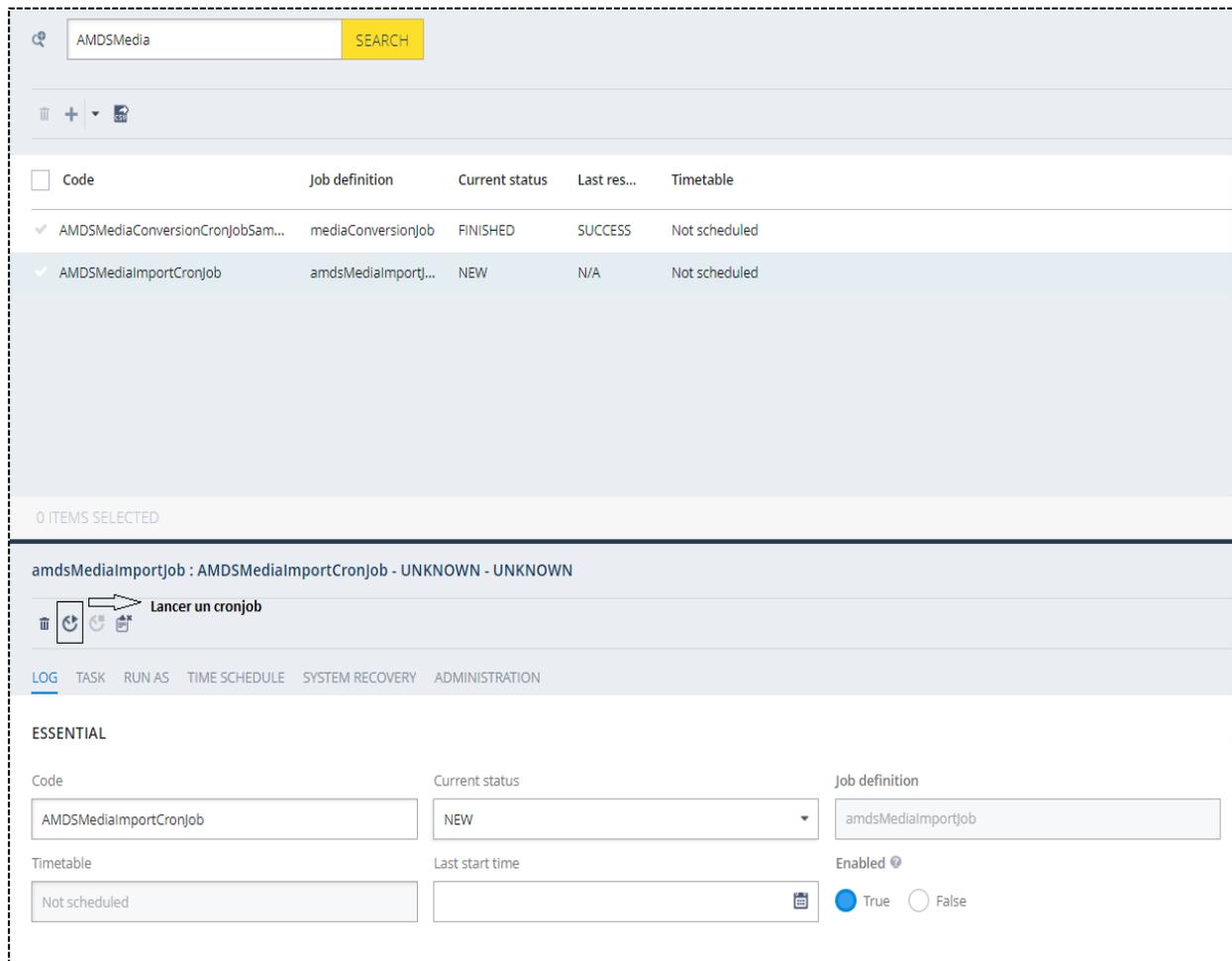


Figure 41 : Lancer un Cronjob depuis le backoffice

## 2.2 Projet AMC

### 2.2.1 Création des tables contenant les informations SOFIE

La migration vers le nouveau PIM nécessite que ce dernier ait la même structure que SOFIE, c'est la raison pour lequel nous avons commencé notre projet par une personnalisation du backoffice.

Parmi les changements majeurs que j'ai apportés :

- Ajout d'un nouveau nœud qui regroupe l'ensemble des tables SOFIE créé.

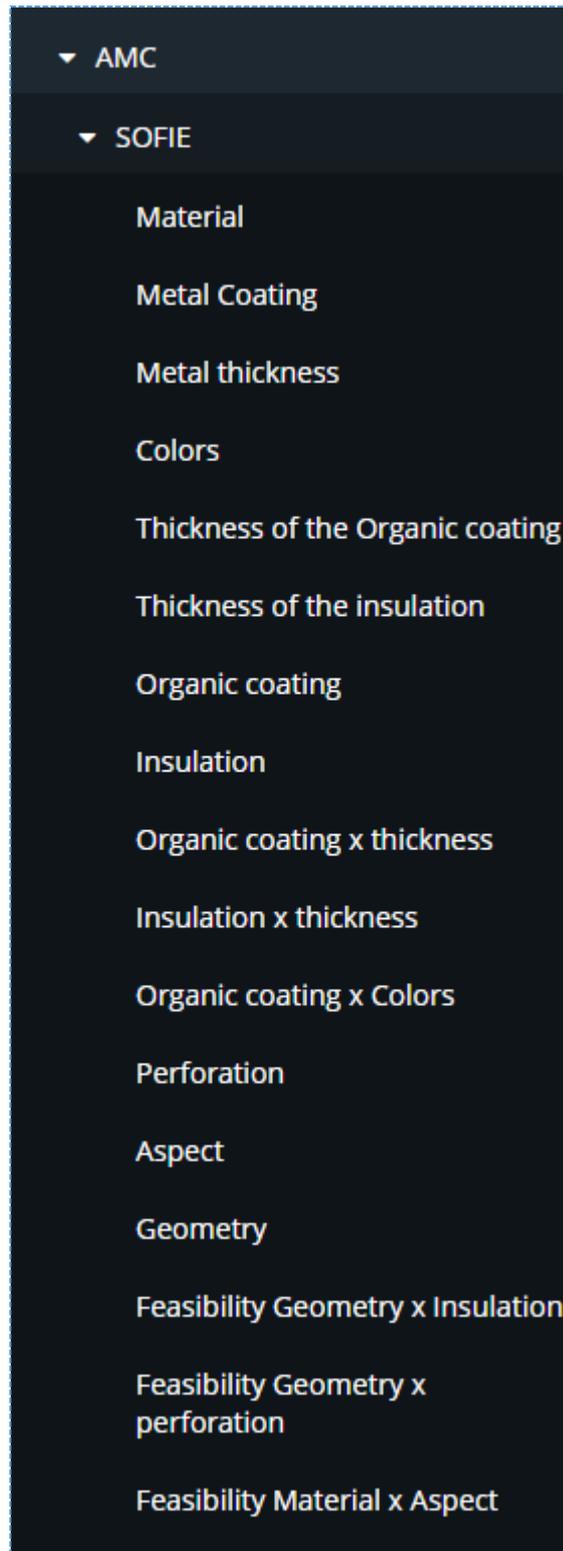


Figure 42 : Nouveaux nœuds

## 2.2.2 Import des données

### a. Fonctionnement

Comme j'ai cité auparavant, l'ET du projet a choisi d'implémenter la notion du Hot Folder hybride pour effectuer l'import des données.

Le Hot Folder est l'un des meilleurs moyens d'importer les données dans Hybris, il est basé sur le framework **Spring intégration**[7].

Parmi les raisons qui expliquent le choix du HotFolder, on peut citer les suivantes :

- Facile à configurer.
- Exploiter la puissance du Framework Spring intégration.
- Très rapide que l'importation par impex (utilise un import multi-threading).

L'idée principale derrière le Hot folder c'est transformer les fichiers csv en entrée à des impex en utilisant des **ImpexConverters** préconfiguré et de les importer en utilisant **ImportService**.

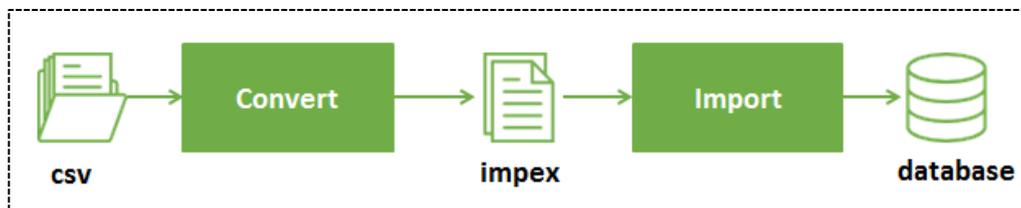


Figure 43 : Processus d'exécution du Hot Folder

### b. Implémentation

Hybris est doté d'une configuration initiale et complète du Hot Folder.

Les **ImpexConverters** pour les prix, les produits, les médias, les stocks, les clients ..., sont déjà implémentés et sont parfaits, cependant, nous allons étendre les fonctionnalités déjà prédéfinies pour pouvoir importer les attributs SOFIE.

Pour Configurer le Hot Folder, il faut passer par deux étapes :

- Définir un répertoire de base où les fichiers CSV seront mis.
- Créer un **ImpexConverter** et l'associer avec un **MappingConverter**.

## 3 Etat d'avancement et perspective

### 3.1 Etat actuel

Nous sommes maintenant arrivés à 80% du projet **AMDS**, la majorité des fonctionnalités majeures requises se sont réalisées dans les trois sprints achevés. Juste après la livraison de la troisième itération et avant de commencer le développement des fonctionnalités de la quatrième itération, nous avons dédié toute une période pour la stabilisation du Distrishop et aussi pour traiter les nouvelles demandes de modification prioritaires à notre client. La mise en ligne (**GoLive**) de la version v1.0 de notre Distrishop est déjà fait pour ensuite entamer la discussion des fonctionnalités à ajouter et à modifier dans la prochaine version.

Pour le projet **AMC**, il nous reste encore un sprint pendant lequel on va développer une fonctionnalité qui permettra au client de générer des documents PDF contenant les détails de ses produits.

### 3.2 Les tests

Les tests unitaires, les tests croisés et les tests de non régression sont effectués par l'équipe de test. Ces tests sont effectués en parallèle avec le développement et pendant les derniers jours avant la livraison du sprint et après avoir fait le « **Code Freeze** » de développement du sprint.

### 3.3 Déploiement

Les deux projets ont été déployés dans trois environnements à savoir :

- **DEV01** : C'est l'environnement de test de SQLI.
- **UAT01** : C'est l'environnement de test de notre client ArcelorMittal (pré-production)
- **PROD** : C'est l'environnement de production

Afin que notre équipe de test puissent faire ses tests d'une façon quotidienne et en temps réel le site est mis dans la PIC\_SQLI, une usine de développement de SQLI tel que :

- **APP1** et **APP2** ainsi que leurs bases de données sont à Rabat.
- **APP3** et **APP4** ainsi que leurs bases de données sont à Oujda.

### **3.4 Perspectives**

**AMDS**, étant le premier projet Hybris à SQLI Oujda représente la première plateforme e-commerce B2B d'ArcelorMittal qui le permettra de faire le e-merchandising en commençant par leur agence de Toulouse en France. Ceci dans l'objectif de mesurer la satisfaction du client vis-à-vis ce Distrishop et qui va par la suite permettre SQLI Oujda de réaliser des versions ultérieures du Distrishop et aussi inclure toutes les autres agences du client en Europe

De même, **AMC** est le premier projet PIM développé au sein de SQLI Oujda, dans une perspective proche d'avoir un site e-commerce similaire à celui du projet **AMDS**, cela va permettre à SQLI Oujda de poursuivre son activité d'Hybris.

### **Conclusion**

Lors de ce chapitre, j'ai expliqué le processus de développement des exigences du client en adaptant la méthodologie Scrum et en respectant les principes de développement d'Hybris ainsi que les normes de codage du Java afin de livrer un produit performant et de qualité. Vers la fin de ce chapitre, j'ai décrit l'état actuel, et ce qui reste à faire en termes de fonctionnalités et les perspectives et objectifs à atteindre.

## Conclusion générale

Le présent rapport décrit les deux projets AMDS et AMC auxquels j'ai participé durant mon stage PFE.

Pour AMDS il s'agit de souligner les différentes phases nécessaires à la mise en œuvre de la première version du Distrishop. Ce dernier qui est une plateforme e-commerce B2B a eu comme objectif la transformation numérique des activités de vente du groupe ArcelorMittal. Pour ce faire, le Distrishop devait être réalisé à base d'Hybris d'où la formation technique et fonctionnelle de la plateforme Hybris que j'ai suivi pendant le premier mois du stage.

Le Distrishop permet actuellement à un client de s'inscrire, se connecter après l'activation de son compte, ajouter des produits à son panier, les consulter, passer une commande après le suivi d'un processus de check-out assez compliqué au niveau des exigences, de plus le client peut lancer des jobs déjà prédéfinis à partir du backoffice.

Concernant le projet AMC, ce rapport ne présente que certaines fonctionnalités demandées par le client et réalisées par moi-même à savoir, la création des tables qui vont contenir les informations exportées depuis l'application SOFIE, l'import de ces informations vers le PIM à l'aide du Hot Folder. Les autres fonctionnalités comme la génération des PDF seront développées après la période de stabilisation.

Pour cette réalisation, nous avons opté pour une réalisation à base Scrum, avec des itérations de trois semaines, en appliquant plusieurs règles de gestion, qui consistent à faire une réunion de planification au début de chaque sprint, faire une démonstration par sprint ainsi que livrer un release notes à la fin de chaque sprint.

Le développement est axé sur les hypothèses fonctionnelles issues du découpage effectué lors des workshops avec le product owner.

Tout ce processus de découpage fonctionnel et de gestion a aidé à aboutir aux résultats attendus de chaque itération, tout en évitant les dépassements dans les livraisons et réalisations en répondant aux exigences du client.

Sur le plan personnel, ce stage m'a permis d'acquérir une nouvelle connaissance en termes de la familiarisation avec une plateforme e-commerce à base JEE qui est celle d'Hybris, également j'ai approfondi mes connaissances dans la conception d'architecture, la qualité logicielle, les bonnes pratiques de codage et de management du projet. En effet, et grâce à l'encadrement continu de mes responsables, j'ai pu développer l'esprit d'analyse et de communication et la capacité d'écoute, l'aptitude à convaincre et défendre mon point de vue au sein d'une équipe multinationale et grande en nombre.

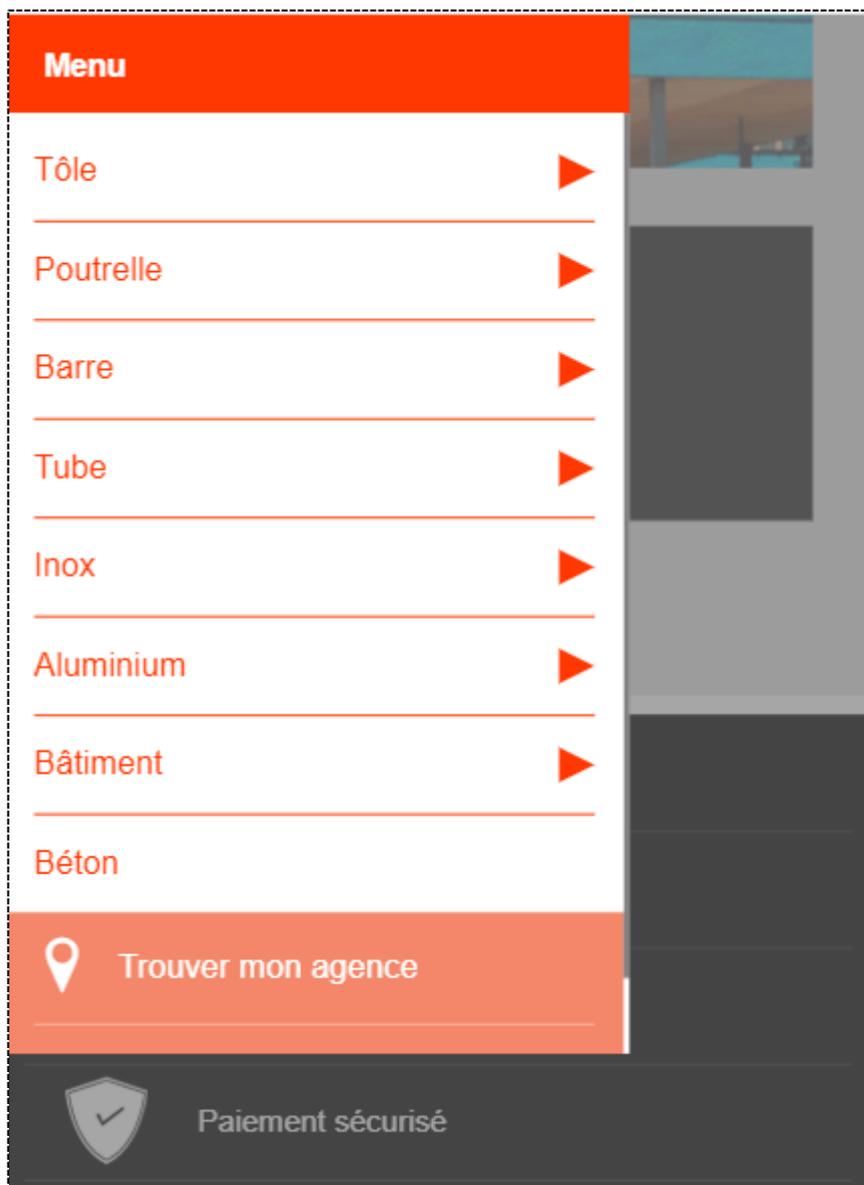
## Webographie

- [1] SQLI, «Site SQLI,» [En ligne]. Available: <http://www.sqli.com/> .
- [2] Arcelormittal, «Arcelormittal France,» [En ligne]. Available: <http://www.arcelormittalinfrance.com/our-business.aspx>.
- [3] Atlassian, «Jira,» [En ligne]. Available: <https://jira2.sqli.com/jira> .
- [4] S. Hybris, «Wiki Hybris,» [En ligne]. Available: <https://wiki.hybris.com>.
- [5] S. hybris, «Help Hybris,» [En ligne]. Available: <https://help.hybris.com> .
- [6] Atlassian, «Confluence,» [En ligne]. Available: <https://jira2.sqli.com/confluence> .
- [7] M. E. fakir, «Stackextend,» [En ligne]. Available: <https://www.stackextend.com/hybris/everything-about-hot-folder-in-hybris/>.

## Annexes

### Annexe A : Le responsive dans Distrishop v1.0

- Mobile



*Figure 44 : Menu de navigation sur Mobile*

Le menu s'adapte en fonction de la taille de l'écran des visiteurs, afin d'offrir une meilleure expérience de navigation.



*Figure 45 : Page Login et inscription sur Mobile*

De même, la page qui permet à un visiteur de s'authentifier/s'inscrire est responsive et adaptable avec n'importe quel appareil.

Par exemple en cas d'un appareil mobile comme le montre la figure ci-dessus, le formulaire d'inscription sera déplacé plus bas.

---

**CONCEPTION ET DÉVELOPPEMENT D'UNE PLATEFORME  
E-COMMERCE/PIM POUR LE GROUPE ARCELORMITTAL À BASE DE LA  
SOLUTION HYBRIS**

---

***Résumé***

Le présent rapport est l'aboutissement du travail effectué dans le cadre du projet de fin d'études au sein de l'équipe AMDS et AMC de l'agence SQLI Oujda, pour l'obtention du diplôme du master Systèmes intelligents et réseaux. L'objectif principal des deux projets auxquels j'ai participé consiste à la réalisation d'une plateforme e-commerce B2B et un PIM à base de la solution Hybris 6.3. Durant les deux projets, nous avons commencé par une analyse fonctionnelle et technique des spécifications du client afin de bien saisir la problématique, puis nous avons entamé l'étape de conception pour finir avec la mise en œuvre tout en respectant l'architecture JavaEE et les conventions de développement Hybris. Pour la mise en place des deux projets, et pour assurer l'agilité et la réactivité avec le client, le choix est tombé sur Scrum comme méthodologie de gestion de projet, avec un cycle de développement itératif et incrémental.

***Mots clés : AMDS, AMC, Hybris, PIM, B2B***

---

**DESIGN AND DEVELOPMENT OF AN E-COMMERCE/PIM  
PLATFORM FOR ARCELORMITTAL GROUP BASED SOLUTION ON THE  
HYBRIS SOLUTION**

---

***Abstract***

This report is the result of the work done as part of the end-of-studies project within the AMDS and AMC team of the SQLI Oujda agency, to obtain the diploma of the Master's degree in Intelligent Systems and Networks. The main objective of the two projects is the realization of a B2B e-commerce platform and a PIM based on the Hybris 6.3 solution. During both projects, we started with a functional and technical analysis of the customer's specifications in order to understand the problem, and then we moved to the design to finish with the implementation while respecting the JavaEE architecture and the Hybris development agreements. For the implementation of both projects, and to ensure agility and responsiveness with the customer, the choice fell on Scrum as a project management methodology, with an iterative and incremental development cycle.

***Keywords: AMDS, AMC, Hybris, PIM, B2B***

**MASTER SYSTÈMES INTELLIGENTS & RÉSEAUX  
DÉPARTEMENT D'INFORMATIQUE  
FACULTÉ DES SCIENCES ET TECHNIQUES DE FÈS  
A.U. 2017 - 2018**