



l'école d'ingénierie
informatique



Dossier Professionnel

Expert en Informatique et Système d'Information
Titre RNCP niveau7 (bac+5) – 2022-2023

Organisme de formation : EPSI

Entreprise du contrat d'alternance : Ausy

Responsable formation :

DEMOIS Sophie

Auteur : FURNON Cyril

25/08/2023

Remerciements

Je tiens à exprimer toute ma reconnaissance à Mme Sophie DEMOIS, coordinatrice pédagogique, Mme. Céline KINGRAPHONE, responsable des relations entreprises, ainsi que leurs équipes pour leurs accompagnements et leurs aides tant professionnels que personnels tout au long de ma scolarité.

Je souhaite adresser toute ma gratitude à M. Nelson LAMBERT, Business Leader Department, et l'ensemble de la société Ausy pour m'avoir permis de réaliser mon alternance, pour leur accueil et leur sympathie tout au long de la période. Je tiens également à remercier de l'entreprise Ausy, M. Hassan OUZEGDOUH, Ingénieur développement Web et tuteur de mon alternance, puis M. Amine LHAIBA, Project Manager, pour mon accompagnement et ma montée en compétences techniques et sociales lors de cette alternance.

D'autre part, je voudrais évoquer le client de ma mission durant ces deux années : DomusVi en mentionnant M. Aurélien HUVETEAU, directeur du pôle Data, BI et développement, ainsi que son équipe « Portail » qui m'ont traité avec bienveillance et m'ont accordé leur confiance pour leur projet.

Je souhaite enfin remercier aussi bien l'EPSI que les entreprises Ausy et DomusVi d'avoir partagé leurs connaissances et leur savoir afin de faire évoluer mon projet professionnel.

Glossaire

EISI : acronyme du titre d'Expert en Informatique et en Système d'Information décerné par l'EPSI

ESN (Entreprise de Services du Numérique) : société experte en informatique et en nouvelles technologies.

Sprint : Période déterminée dans le cadre d'une méthode Agile d'un projet. Durant cette période, l'équipe s'engage sur un objectif et les tâches correspondantes.

Backlog : Ensemble des tâches qui constituent un projet. Cette liste est amenée à évoluer en fonction des besoins.

US : User Story : Description d'une fonctionnalité à développer. Elle est rédigée afin de faire ressortir le besoin de la fonctionnalité.

Spécification : Document listant le détail et le fonctionnement d'une ou plusieurs fonctionnalités.

CRM : Customer Relationship Management ou gestion de la relation client est un logiciel aidant les entreprises pour le suivi, la fidélisation et le développement de leur clientèle.

Environnement : en informatique, on parle d'environnement afin de simuler une application avec une configuration précise.

API : (Application Programming Interface) est une interface logicielle qui permet de « connecter » deux systèmes informatiques afin de faciliter l'échange de données et de fonctionnalités.

REST : (REpresentational State Transfer) est principe de structure et de contraintes principalement utilisé pour la gestion d'API.

SQL (Structured Query Language) : langage informatique permettant d'interagir avec les bases de données.

Front-end : partie visible d'une application ou d'un logiciel communiquant avec les utilisateurs.

Back -end : partie invisible d'une application ou d'un logiciel ne communiquant pas directement avec les utilisateurs.

Git : est un outil de gestion de versions décentralisée permettant la gestion mutualisée d'un dossier. Il permet la gestion de versions de ces fichiers en assurant la sauvegarde de plusieurs versions du répertoire. La récupération d'une version antérieure et la fusion de travaux sur un même fichier sont ainsi facilitées grâce à cet outil.

HTML (HyperText Markup Language) CSS (Cascading Style Sheets) : Langage informatique définissant l'affichage des pages web, l'HTML gérant les objets et le CSS gérant leur position, leurs couleurs...

IDE (IDevelopment Environment) : est un logiciel permettant d'éditer et d'automatiser des tâches de programmation. Leur interface regroupe souvent des outils pour compiler, déboguer et pour gérer la complétion de code.

PCA : Plan de Continuité d'Activité : Ensemble de procédures visant à anticiper des incidents et à prévoir des processus pour éviter tout arrêt de l'activité de l'entreprise.

PRA : Plan de Reprise d'Activité : Ensemble de procédure visant à reprendre l'activité de l'entreprise le plus vite après un incident. Le but est de comprendre les cas d'incidents pour répondre au mieux.

I. Sommaire

Table des matières

Remerciements	1
Glossaire	2
I. Sommaire	3
II. Introduction.....	4
III. Environnement professionnel	5
A. La structure d'Ausy.....	5
B. Le client de la mission : DomusVi.....	8
IV. Valorisation des compétences	10
A. Analyse du contexte de la mission	10
1. Contexte de l'entreprise Ausy	10
2. Gestion de la mission Domusvi	12
3. Présentation de la mission DomusVi.....	15
B. Organisation de l'équipe Portail.....	19
1. Méthode Agile	19
2. Méthode Agile adoptée par l'équipe Portail.....	21
3. Processus de gestion du développement.....	24
4. Environnement de travail.....	30
C. Projet MSAL :	36
1. Besoin du projet	37
2. Processus du projet.....	39
3. Fonctionnement de MSAL sur le projet Portail.....	41
D. Projet CRM	44
1. Présentation du CRM	44
2. Projet de refonte du CRM	45
3. Règles de gestion.....	48
E. MSPR OuiCar	50
1. Besoin du projet	50
4. La structure des données	52
5. Traitement des données	53
V. Conclusion	55
Table des figures.....	57
Webographie.....	58
Annexes	59

II. Introduction

Ce dossier rentre dans le cadre de la formation délivré par l'EPSI Lyon : Expert en Informatique et en Système d'Information, titre RNCP de niveau 7.

Il fait l'objet de document présentant la validation des compétences nécessaires à l'attribution du titre EISI en rendant compte de l'alternance que moi-même, Cyril FURNON, étudiant à l'EPSI, ai réalisé auprès de l'entreprise Ausy durant la période du 14 septembre 2021 au 22 septembre 2023.

Ausy est une ESN française acteur majeur de conseil et d'ingénierie en technologies dont le siège se situe à Issy-les-Moulineaux. Elle dessert des services s'étalant du conseil, de l'expertise et la sous-traitance jusqu'au consulting et l'ingénierie technologique. Forte de presque 35 années d'expérience, Ausy rejoint le groupe Randstad en 2017. Randstad est fondé en 1960 à Amsterdam et est l'un des leaders mondiales en services en Ressources Humaines avec presque 4 milliards d'euros de chiffres d'affaires et près de 30 000 entreprises clientes en 2022.

Crée en 1989, Ausy s'est spécialisée dans l'informatique technologique autour de trois principaux pôles : Paris, Lyon et Rennes. Elle s'est ensuite rapidement développée en France puis, elle s'est exportée au Luxembourg et en Allemagne et a renforcé son statut pour arriver aujourd'hui avec plus de 50 sites. En Europe, Ausy s'étend même dans des pays de l'Est, très tôt en Roumanie en 2006 et plus tard en Pologne. De plus, l'entreprise française s'est également implantée en Inde (2012) et a fait une acquisition aux Etats-Unis, montrant ainsi son impact mondial. Plus de détails peuvent être retrouvés sur l'*Annexe 2 : Historique de Ausy*.

III. Environnement professionnel

A. La structure d'Ausy

En France, Ausy possède une vingtaine d'agences dont une dizaine de « delivery centers » ou centre de production : ce sont des agences ayant des engagements (production ou services) auprès de client. La *Figure 1 : Carte des Delivery Centers en France* ci-dessous montre les dix delivery centers dont celui de Lyon. Toutes ses agences forment un socle de compétences allant de domaines techniques : IA & Big Data, cybersécurité, développement ... à la gestion de projets, à l'analyse et l'audit, l'optimisation et le conseil sur la gestion de projets...

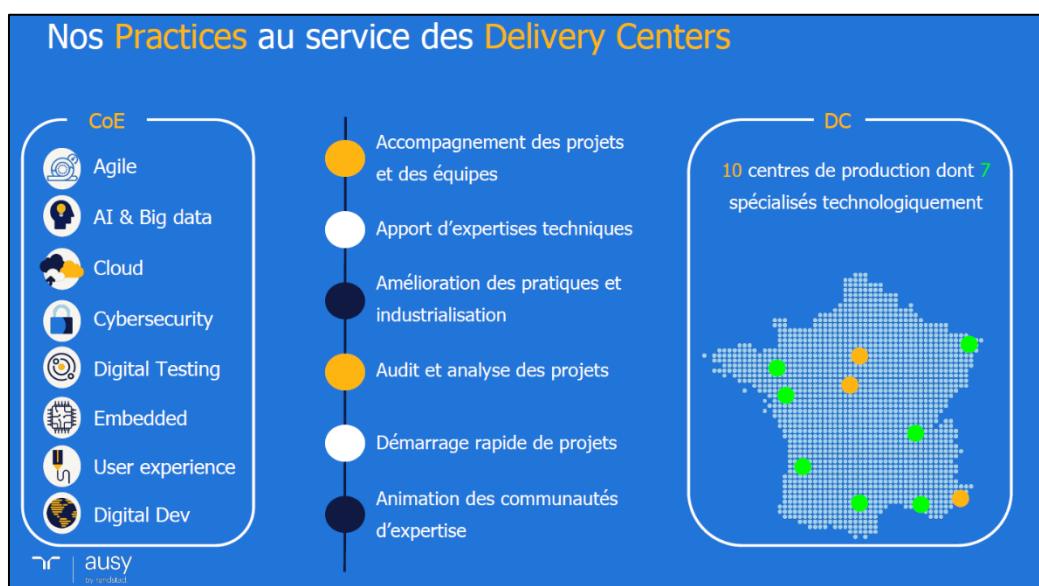
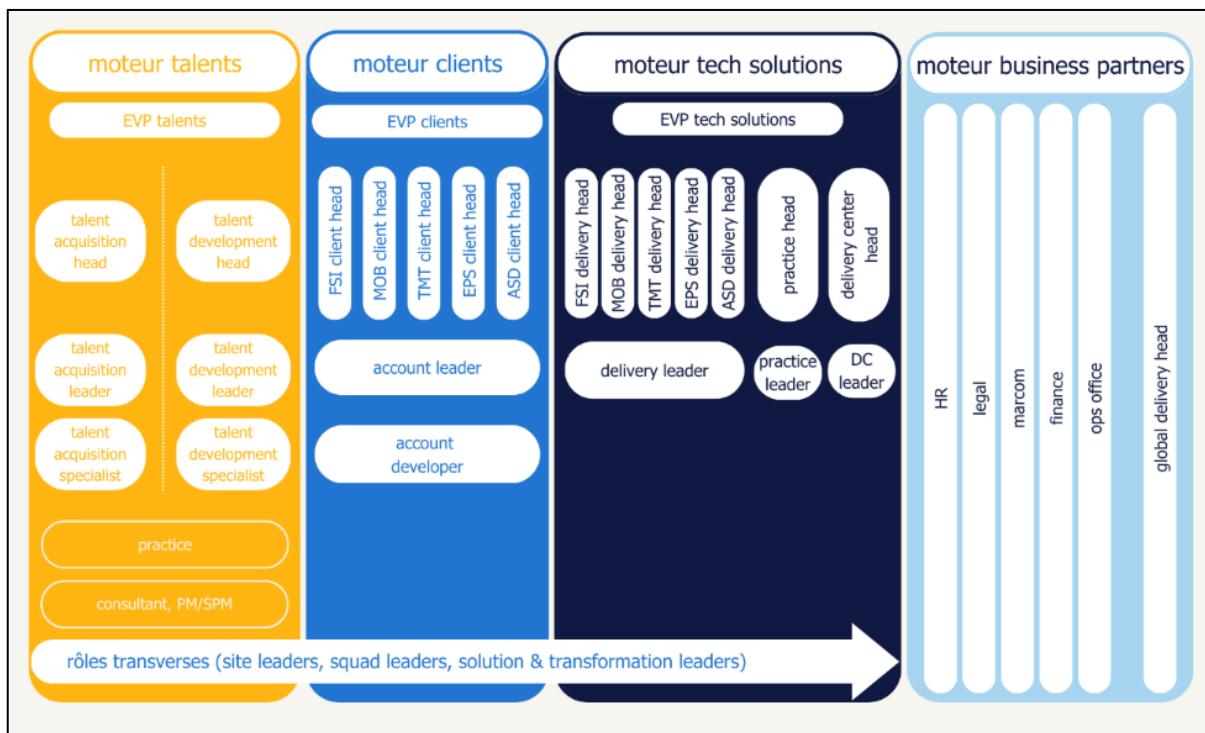


Figure 1 : Carte des Delivery Centers Ausy en France

Concernant son organisation, Ausy se structure autour de différents « moteurs », ces moteurs segmentent les employés suivant leurs activités :

- « Talents » : représente les personnes dans l'accompagnement, notamment lors de l'arrivée et dans l'accompagnement de l'évolution professionnel.
- « Business Partners » : correspondant au responsable de qualité, d'administration, des Ressources Humaines, des Finances, d'outillage ou de formation... des métiers génériques à la gestion d'entreprise.
- « Clients » : ce moteur regroupe les interactions avec les clients porteurs de projets, de demandes de moyens ou de ressources. Suivant un découpage en fonction de l'activité des clients, des rôles supplémentaires sont ajoutés au sein de ce pôle.
- « Tech solutions » : qui entoure toute production ou service visant à répondre au besoin d'un client. Cette partie est également découpée suivant l'activité des clients. C'est dans cette partie que mon rôle est situé.

L'organigramme que représente la *Figure 2 : Organigramme Ausy* plus bas, illustre bien ce découpage et des rôles transverses.



Nous nous intéressons au découpage suivant l'activité des clients. En effet, Ausy gère de nombreux clients et découpe ainsi les moteurs « clients » et « tech solutions » suivant leurs appartements aux clients. Les abréviations visibles sur l'image plus haute encadrent des domaines définis :

- « FSI » pour les métiers des finances, de la santé et de l'assurance
- « MOB » pour les entreprises liées à la mobilité et aux transports
- « TMT » pour les sociétés de télécommunications
- « EPS » correspond à l'activité d'énergie et du marché public
- Et « ASD » encadrent la partie défense aérospatiale

Plus tard, nous verrons que la mission sur laquelle j'ai été placé, rentre dans la catégorie « FSI »

Avant de détailler l'activité de Ausy, j'aimerais décrire un principe essentiel pour l'entreprise. Elle s'est engagée depuis 2009 sur le plan de la Responsabilité Sociétale d'Entreprise, c'est-à-dire qu'elle intègre dans ses stratégies les préoccupations environnementales, éthiques et sociétales. Je cite les mots d'un site interne à l'entreprise ...

« Dans le cadre de sa politique RSE, Ausy s'engage sur le plan social à :

- Lutter contre les discriminations et les inégalités Femmes-Hommes
- Développer l'embauche des jeunes et le maintien dans l'emploi des séniors
- Développer l'emploi des personnes en situation de handicap
- Veiller aux conditions de santé et de sécurité au travail des salariés

Sur le plan environnemental à :

- Réduire les émissions de gaz à effet de serre
- Économiser l'énergie
- Développer un parc automobile moins polluant
- Développer des actions de recyclage et de réduction de consommables

Sur le plan sociétal à :

- Développer une culture RSE
- Avoir un statut de fournisseur privilégié auprès de nos clients
- Encourager nos partenaires dans une démarche éthique »

... prouvant ainsi ses engagements pour l'égalité Homme-Femme, les seniors et le handicap, pour un plan pour l'économie d'énergie et contre la pollution ainsi que la continuité de ses engagements et de ceux de ses partenaires.

Synonyme de son engagement, je détaillerais la partie concernant le handicap au sein de la société. Depuis 2014, Ausy possède une cellule « Mission Handicap » afin de permettre l'embauche et l'intégration des personnes handicapées. Cela se concrétise grâce à des accords d'un an en 2015 et de trois ans en 2019 et 2022 avec l'institut public de « Direction régionale des entreprises, de la concurrence, de la consommation, du travail et de l'emploi ». Concrètement, cette cellule sensibilise l'ensemble de l'entreprise à ce principe et soutient le recrutement et l'intégration des personnes handicapées, leur aménagement d'environnement de travail et la contribution au projet au service du handicap.

Pour se recentrer sur l'activité de la société Ausy, nous pouvons regarder la *Figure 3 : Macro-processus de Ausy*, plus bas, qui propose un découpage des processus généraux de Ausy. Ces processus se regroupent en trois types : les « processus opérationnels » définissant les processus au cœur des activités métiers, le « Support » représentant les actions liées indirectement à l'opérationnel et les « processus managériaux » correspondant aux actions soit décisionnelles soit de pilotage soit en lien avec la qualité.

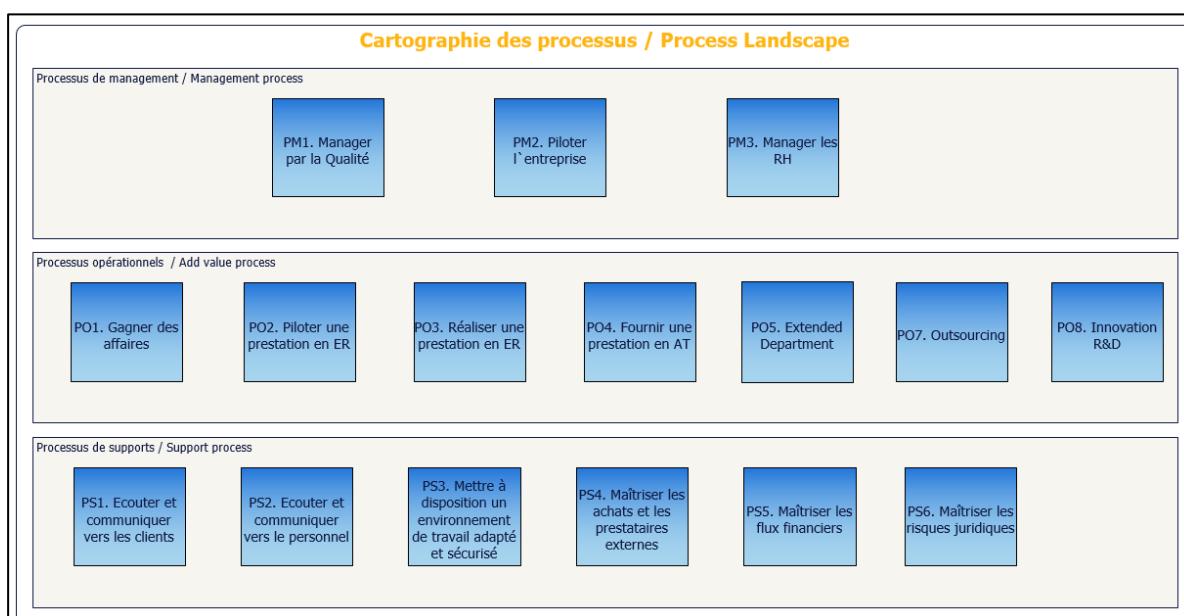


Figure 3 : Macro-processus de Ausy

Sur la figure ci-dessus, nous pouvons voir qu'Ausy reprend la définition des processus managériaux. La partie opérationnelle est plus concrètes et concerne les actions de production de service ou biens pour un client. Concernant le support, Ausy donne une importance à la communication et l'écoute de toutes parties prenantes et incorpore la gestion financière et juridique de son activité aux processus support.

Pour mon positionnement je participe à l'opération auprès d'un client en fournissant une prestation en « Assistance Technique ». C'est lors de mon arrivée en octobre 2021 que j'ai rejoint directement mon tuteur d'alternance M. Hassan OUZEGDOUH et l'équipe travaillant avec le client. Cette équipe Ausy était composée de trois consultants avant ma venue, rejoint plus tard par d'autres, tous pour du conseil et de l'assistance technique auprès du client.

Après cette première présentation d'Ausy, j'enchaînerai sur la présentation du client de l'unique mission de mon alternance : DomusVi

B. Le client de la mission : DomusVi

DomusVi est une entreprise créée en 1983 en France avec un siège à Paris, elle se spécialise dans les domaines de l'habitat, des services et des soins destinés aux personnes âgées. Présente dans huit pays, dont la France, l'Espagne, le Portugal, l'Allemagne, les Pays-Bas, l'Irlande, le Chili et l'Uruguay, DomusVi gère près de 500 résidences médicalisées et résidences services, dont plus de 240 en France. Elle possède également 60 agences d'aide à domicile en France sur plus de 100 au total. L'entreprise emploie près de 14 400 personnes en France et plus de 50 000 en tout dans le monde.

Fort de son engagement pour la société, DomusVi prône cinq piliers : l'humanité et la bienveillance, le sens de l'autre, la sincérité des émotions, la confiance partagée, ainsi que l'esprit pionnier.

De plus, l'entreprise s'est engagée à respecter les principes éthiques qui s'appliquent à son activité et à adopter un comportement socialement responsable envers l'ensemble de ses parties prenantes. Elle va jusqu'à encourager tous les collaborateurs à agir de manière responsable et éthique, en accord avec ses propres valeurs.

DomusVi propose des services web pour la gestion d'établissements, qu'ils soient des maisons de retraite médicalisées, des résidences seniors ou des agences d'aide à domicile. Parmi eux, nous retrouvons la gestion des contrats, des clients, des finances, des ressources humaines et l'apport d'outils d'administration pour les organismes publics.

Mais également de l'aide et de l'accompagnement pour toutes personnes destinées aux services fournies par ces établissements.

Cette présentation de DomusVi est inspirée des sites : <https://www.domusvigroup.com/le-groupe/> et <https://www.domusvi.com/domusvi-en-toute-confiance>.

La société s'est structurée autour d'une « équipe Transformation Client, Digitale et Technologique » comme le décrit la *Figure 4 : Organigramme de DomusVi* ci-dessous. Nous pouvons décomposer cette équipe en pôles principaux : le marketing digital, le SI et le pôle Innovation. Nous nous intéresserons au groupe « Data, BI & Développement » se trouvant dans le pôle SI (encadré en rouge sur la *Figure 4 : Organigramme de DomusVi*). Ce groupe gère trois projets dont le projet Portail où je me situe. Le détail de l'équipe « Portail » et son organisation sera détaillé plus tard dans le dossier.

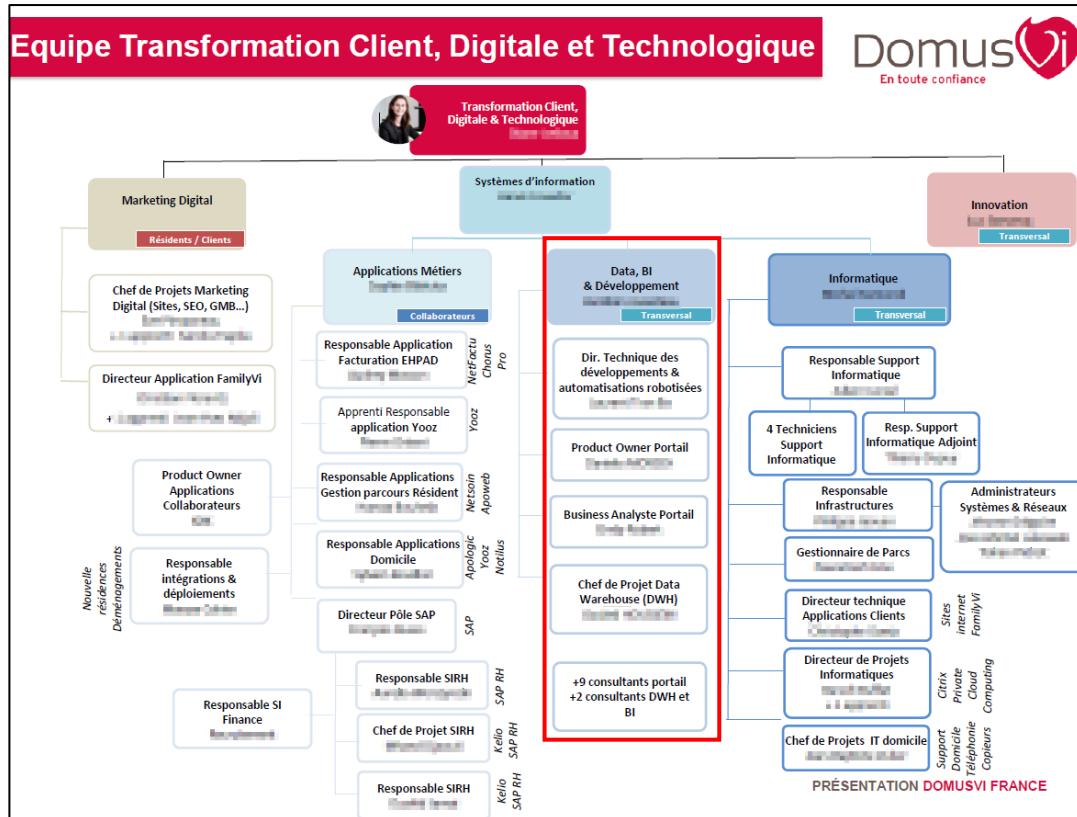


Figure 4 : Organigramme de DomusVi

IV. Valorisation des compétences

A. Analyse du contexte de la mission

Blocs de Compétences :

- A1 – Analyse et définition de la stratégie des systèmes d'information
- A2 – Pilotage d'un projet Système d'Information

Activité : Définition d'une stratégie, Suivi des réalisations d'un projet

Compétences choisies : A1C3, A1C4, A1C5, A1C6, A2C5, A2C6, A2C7

1. Contexte de l'entreprise Ausy

Comme vu plus haut, Ausy se définit comme une ESN : une entreprise de services et de conseil en informatiques avec des compétences techniques mais aussi avec des compétences de gestion de projets. Suivant son cadre, ses services et ses possibilités, j'ai dressé un SWOT, une matrice regroupant les forces, faiblesses, opportunités et menaces sujets à la société. C'est ce que montre la *Figure 5 : SWOT de Ausy*.

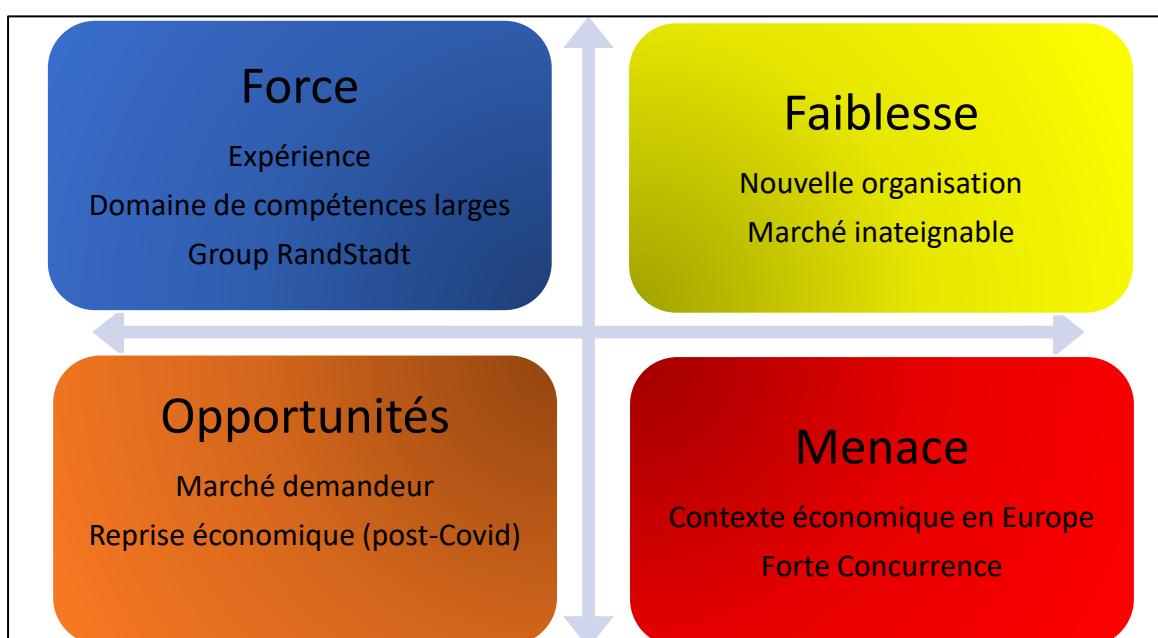


Figure 5 : SWOT de Ausy

En termes de force, Ausy se base sur près de 34 ans d'expériences pour répondre à ses clients et un domaine de compétences larges. De plus, depuis 2017 elle possède le soutien du groupe mondialement renommé : Randstad. Toutefois, la société vit une restructuration qui a besoin de temps afin de pouvoir montrer des résultats. Pour son essor, Ausy est bloquée par certains marchés restreints mais elle peut se rattacher à un marché de l'informatique très dynamique. Ensuite, la société peut à la fois profiter d'une sortie de crise sanitaire permettant une reprise économique mais est aussi confrontée à un contexte économique en Europe tendu. Enfin, nous avons dit que le marché est dynamique mais également au niveau des entreprises offrant des services informatiques ce qui peut apporter une certaine concurrence sur les projets.

Pour continuer sur l'approche de mon entreprise, je vais détailler les processus qu'Ausy peut présenter avec la figure suivante : *Figure 6 : Schéma des principaux processus de Ausy*. Cette image découpe les activités par deux niveaux : métier et fonctionnel.

(A noter : *Initialement, cette image contient deux niveaux supplémentaires : applicatif et infrastructure mais la confidentialité sujet à ce schéma empêche l'ajout de plus de détail.*)

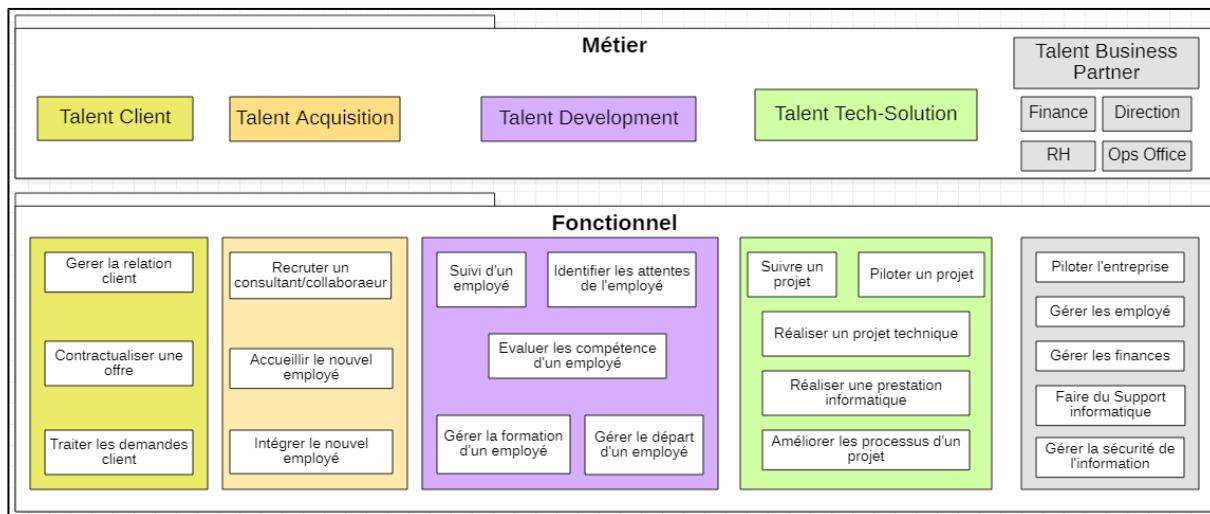


Figure 6 : Schéma des principaux processus de Ausy

Sur cette image, nous pouvons retrouver les différents moteurs développés dans la partie [3. A. La structure d'Ausy](#) avec le détail des activités que chacun des « talents » effectue au quotidien. Ceci décrit plus précisément la structure au sein de l'entreprise.

Concentrons-nous désormais sur l'agence de Lyon : avec un peu moins de 200 employés, elle se spécialise sur un centre de service en conseil et en expertise notamment en agilité et en développement digital avec les langages .NET et PHP. Ausy apporte aussi un accompagnement des projets et des équipes, une amélioration des pratiques et une industrialisation des gestions de projets. Enfin, elle propose d'auditer et d'analyser l'organisation des projets. Parmi le secteur d'activité des clients, nous retrouvons des acteurs connus dans l'assurance et la banque, dans l'énergie, le transport, le public, également le groupe Randstad et bien d'autres encore...

Grâce à son expertise en gestion de projet, le delivery center de Lyon possède de nombreux « project managers » qui s'occupent de la gestion, de l'accompagnement et du suivi des projets.

Nous pouvons découper l'encadrement du travail auprès de client en trois types majoritaires afin de définir les attentes entre le client et le prestataire :

Le plus simple à comprendre reste l'« Engagement de résultats », dont l'exigence se centre autour de la réalisation d'objectifs définis dans la mission. Suivant les attentes, qu'il est impératif de bien détailler, les prestataires s'engagent à livrer un résultat répondant à ses attentes. La présence de ressources comme des documentations est indispensable à la bonne définition du périmètre. Il est souvent questions d'une date de livraison qui engage également la responsabilité des prestataires en cas de non-respect.

L'Engagement de moyens est une autre possibilité : il met en avant la mise en œuvre des moyens pour la réalisation d'objectifs. Ces moyens peuvent être humains, techniques, matériels. Cependant, un résultat n'est pas garanti à la fin, si les moyens ne sont pas jugés suffisants ou si des événements

tiers viennent obstruer l'accomplissement de la mission, la responsabilité des prestataires ne peut être engagée.

Enfin, la dernière option est l'assistance technique ou prestation en régie : cette dernière ne définit pas forcément de résultats à obtenir mais engage des compétences, des conseils et une expertise pour aider le client à la réalisation de ses objectifs. Cependant, il est impératif de définir une période à cet engagement.

Le type n'est pas clairement défini pour tous les encadrements des missions et chacune d'entre elles est sujet à des négociations autour d'un périmètre et d'attentes uniques. Toutefois, il est possible de définir des indicateurs de performances dans chacun des cas qui vont permettre de contrôler l'avancement du projet et d'estimer les résultats finaux. Pour le suivi des projets, nous pouvons évoquer les KPI (Key Performance Indicators). En effet, ces indicateurs sont un moyen de contrôler le déroulé des projets. En les définissant au début de la mission, ils peuvent faire partie de critères concrets permettant d'illustrer son avancement et de valider les objectifs à la fin. Suivant les attentes des projets, les indicateurs peuvent s'orienter sur le respect des délais : délais de livraisons, respect des délais de la résolution d'incidents. Si les objectifs sont orientés sur la qualité, le taux de retours et le taux de disponibilité des services sont des indicateurs préférables.

En tant qu'ESN et malgré cette diversité de contrats, Ausy suit chacune des missions afin de pouvoir répondre au mieux aux besoins des clients. Etant même experte dans la gestion et l'accompagnement de projets, elle s'assure de mettre en place des échanges avec les clients comme les employés techniques afin de garantir un suivi et un certain contrôle. Comme échanges principaux, il y a :

- Le « Kick-off » définissant le lancement, il vise à rappeler les engagements des deux parties avec leurs objectifs. Un détail du périmètre et de certaines contraintes est énoncé avec les grandes étapes du déroulement de la mission.
- Les « COPROJ » ou comités opérationnels sont des réunions fréquentes donc le but est de suivre l'avancement des opérations. L'ensemble des projets et leur avancement est surveillé dans cette réunion, un suivi des potentiels livraisons et des facturations est fait. La discussion comporte également la gestion des risques avec leurs actions.
- Enfin, le « COPIL » ou comité de pilotage se détache de l'opération pour développer une synthèse de la période. Il est moins courant que le COPROJ et aborde un planning ou « roadmap » montrant l'avancement actuels des projets et détaillant les prochains projets. Un moment est accordé au suivi des risques et une attention est portée sur l'amélioration continue sur l'ensemble de la mission. Enfin, un bilan de la période est dressé pour conclure la réunion, l'utilisation d'indicateurs peut servir à la présentation du bilan.

Après ces explications de l'encadrement de projet par Ausy, nous pouvons détailler la gestion de la mission DomusVi.

2. Gestion de la mission Domusvi

Depuis près de cinq ans, Ausy et DomusVi travaillent ensemble en engagement de moyens. Concrètement, DomusVi demande des consultants pour apporter leur expertise et leur compétence afin de réaliser leurs objectifs. Ces objectifs seront détaillés dans la partie suivante : [A. 4. Présentation de la mission DomusVi](#). Concernant l'organisation, Ausy a la capacité et l'expertise pour

orchestrer la mission en mode « solution », c'est-à-dire en ayant la gestion du projet et du pilotage. Cependant, DomusVi assure ces parties depuis le pôle « Data, BI & Développement » (référence à l'organigramme visible sur la Figure 3). L'équipe Ausy travaille ainsi en conseil et assistance technique avec DomusVi. L'ensemble de l'équipe est appelé à intégrer l'équipe « Portail » avec laquelle elle est rattachée. Dans la majorité des cas, on demande aux consultants de se déplacer dans les locaux avec les autres membres de l'équipe.

Cependant dans notre cas, l'agence cliente est à Paris, il a donc été décidé de travailler à distance. Cela n'inclut pas un télétravail total et Ausy demande un minimum de jour de présence à Lyon afin de garder une proximité avec ses employés. Elle tient à garder un côté humain avec l'ensemble de son personnel. Certains points dérogatoires ont empiété et empiète encore sur la présence de l'équipe Ausy à l'agence de Lyon : la fin de la crise sanitaire Covid et la programmation de mise en production. Mais étant continuellement en distance avec le reste de l'équipe Portail, le télétravail ne s'immisce que très peu dans l'organisation des consultants Ausy. L'ensemble de l'équipe Portail a même définit des actions afin de pallier cette distance, je l'évoquerai plus tard dans la partie [B. Organisation de l'équipe Portail](#)

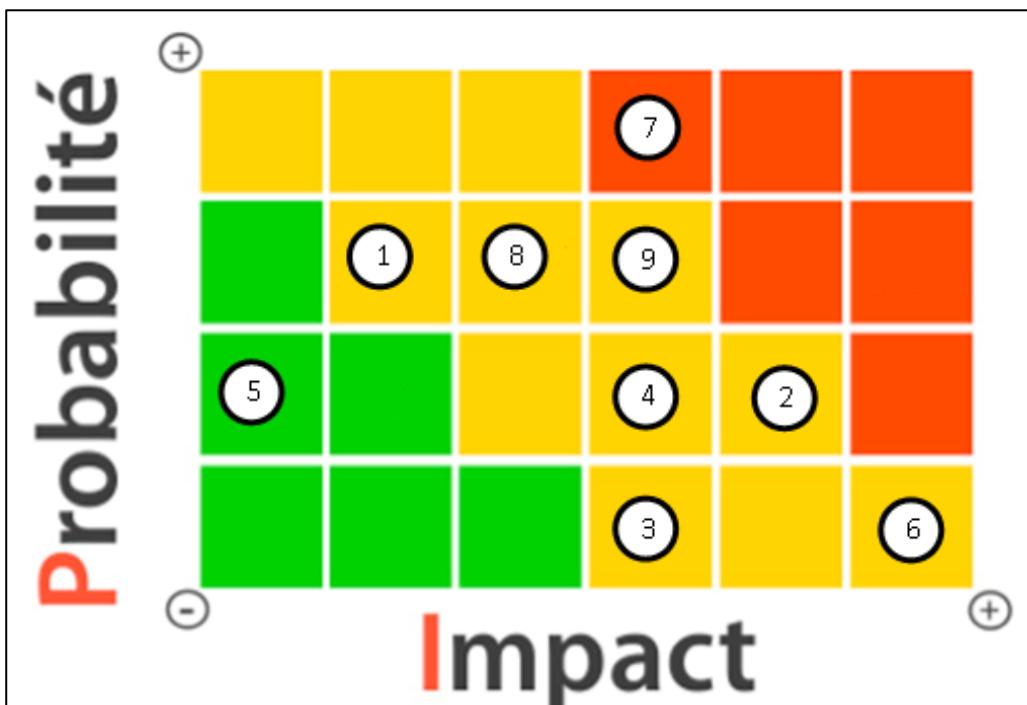
Bien que le pilotage et la gestion de projet soit effectué par DomusVi, le suivi de l'entreprise lyonnais s'organise avec les mêmes réunions : le COPROJ et le COPIL. Le comité opérationnel est moins fréquent que dans d'autres missions mais suit le même schéma. Chaque mois, il est rappelé les faits marquants de la dernière période et l'état d'avancement des projets. Ensuite, une discussion est ouverte pour la gestion des risques et les actions à prendre pour les prévenir. Enfin, après un point sur les finances de la mission, la discussion peut se diriger sur des sujets plus généraux : de potentiels rencontres physiques, les retours d'une nouvelle organisation de travail...

Quant au comité de pilotage, Ausy et DomusVi se rencontrent tous les trois mois pour cette réunion de synthèse et de prévisions. Un plan d'avancement et d'arrivée des projets est énoncé et un accent est mis sur les échéances à venir. Par ailleurs, il y a un suivi des risques et leurs actions avant un détail sur la partie financement. L'entretien se termine par un bilan et un retour de satisfaction du client afin de s'assurer du bon avancement ou de pouvoir réagir en cas mécontentements.

D'autre part, chaque semaine les consultants Ausy ont une réunion avec les project managers en lien avec le client. Ces points permettent de garder une communication avec Ausy. Le but principal est de connaître le ressenti de chacun et l'activité récente. La discussion découle aussi de potentiels réorganisation, remonté ou demandes du client ou à faire au client. Il est utile pour les résumés du COPIL après sa réalisation.

Enfin, Ausy met également un point à suivre individuellement chacun de ses employées. Le suivi se fait avec les retours des clients, des project managers mais également lors de PPR. Ces « Performance Project Reviews » visent examiner avec un project manager attribué l'ensemble des objectifs de l'employé, à les évaluer et faire une synthèse des compétences utilisées lors des missions. Le but est de construire des plans d'évolutions pour l'employé et suivre leurs avancements. Pour cela, il y a un PPR par trimestre.

Concernant les risques d'un projet, il existe des matrices de risques définissant la priorité de prise en compte. En prenant exemple sur la Mission DomusVi, j'ai créé une matrice des risques, elle est visible sur la *Figure 7 : Matrice des risques de la mission DomusVi*. Nous pouvons observer deux axes : la probabilité qui correspond à l'apparition possible du risque et l'impact qui fait référence à la répercussion qu'il peut avoir sur la réalisation de la mission. Elle est fournie avec une liste exhaustives de risques sur la mission.



1. Problèmes de connexion des consultants Ausy empêchement une connexion aux outils DomusVi (base de données, environnements informatiques de DomusVi...)
 ⇒ Cette probabilité est limitée car une connexion internet régit le besoin de connexion, l'impact serait cependant l'incapacité des consultants à travailler.
2. Arrêt de travail importants (plusieurs mois) de plusieurs développeurs
 ⇒ L'incapacité de travail de plusieurs développeurs (Aisy ou DomusVi) bloquerait grandement l'avancement des tâches et donc des délais des projets.
3. Conflit au sein de la mission
 ⇒ Même si les répercussions directes sont moindres sur l'avancement, l'ambiance de travail en sera dégradée, diminuant le dynamisme et l'envie de travailler de tous.
4. Perte de compétences critiques au projet
 ⇒ Si le départ de certains membres de l'équipe n'est pas préparé, c'est une perte de connaissances qui peut suivre et empiéter sur la mission.
5. Besoins et demandes mal formulés
 ⇒ Un besoin mal formulé équivaut à un résultat imprécis et des retards, une attention est à porter sur la rédaction des spécifications et des documentations.
6. Destruction d'un bien indispensable à l'accomplissement de la mission
 ⇒ Même si des solutions sont possibles pour amoindrir l'impact d'un tel incident, suivant le bien, l'avancement peut être bloqué voire la mission interrompue.
7. Non-respect des délais de livraisons
 ⇒ Des retards des livraisons peuvent être plus ou moins acceptés mais la répétition de retards peut engager la responsabilité de l'équipe et diminuer la satisfaction cliente.
8. Mauvaises gestions de bugs sur la version du site utilisé par les utilisateurs
 ⇒ La gestion des bugs est souvent une clé de la satisfaction client, le nombre et le temps de résolution des bugs peuvent être des indicateurs pour la mesurer.
9. Bugs sur la version du site utilisées par les utilisateurs
 ⇒ Les impacts dépendent de l'erreur en elle-même mais les corrections peuvent être prévenues par des outils et plusieurs solutions sont disponibles pour les corriger.

Ces risques ne sont pas exhaustifs mais il est impératif de pouvoir prévoir un maximum de risques afin d'être préparé à ces contextes. Imaginons un plan afin de prévoir le risque numéro 4 : Perte de compétences critiques au projet. Il est possible de faire une matrice des compétences de l'ensemble des membres de l'équipe dans le but de déterminer les compétences critiques. Ainsi pour limiter ces pertes, nous pouvons imaginer un partage de connaissances et des formations pour permettre la transmission de ces compétences critiques. Nous pouvons également prévoir du travail en binôme afin de partager les connaissances des personnes possédant une compétence critique.

Du point de vue d'Ausy, pour répondre aux besoins de ressources de DomusVi, il faut sélectionner les critères qui sont les plus précieux pour le client. Etonnamment la connaissance des technologies n'est pas le facteur le plus important car ce manque peut être comblé par des formations sur les sujets. Avec la distance séparant les entités, la capacité à bien communiquer est un axe plus essentiel pour répondre à la mission. Étant consultant, la place de la communication est nécessaire pour cibler les points d'attention. En complément, nous pourrions citer la prise d'initiative et la force de proposition, émanant d'une implication dans le projet, elles montreraient l'expertise et la capacité à répondre à des besoins ou des exigences sous-jacentes.

En tant que consultant Ausy, il faut être conscient que l'on représente son entreprise et qu'un consultant est un moyen de communication entre les entreprises. Il est important de répondre aux attentes et de conseiller le client mais c'est aussi la possibilité de proposer de nouvelles interactions entre les deux entités, par de nouveaux projets ou par l'élargissement de l'actuel. Les conseils et bonnes pratiques pour l'amélioration de l'organisation et de l'exécution du travail sont un point central de l'activité de consultant.

Nous continuons le contexte de la mission part la présentation de la mission en elle-même.

3. Présentation de la mission DomusVi

Avant les détails de la mission, il est nécessaire d'analyser quelques points importants à l'entreprise afin de pouvoir mieux répondre à ces attentes. Comme cité lors de la présentation du Group DomusVi, la société met un point sur son engagement et sa responsabilité envers la société et sur l'environnement. En effet, le service principal du groupe correspond à un engagement envers les personnes seniors, c'est donc un « aspect de survie » (citation d'un des directeurs techniques de DomusVi) qui est portée entièrement par la direction générale. L'entreprise communique en externe sur ce sujet et est porteuse de formation sur le RSE (Responsabilité sociétale et environnementale) au sein de son entreprise. Enfin, la société aborde des sujets environnementaux et met en action des plans de recyclage de certaines matières propre au domaine de la santé.

Un autre point important à aborder est la politique de sécurité de DomusVi, dont les Responsables de la Sécurité de l'Information ont la tâche. La politique mise en place se veut très stricte avec un logiciel de sécurité commun à tous et une demande continue de justifications et de précautions. Par ailleurs, il y a une limitation du Shadow IT très rigoureuse. Le Shadow IT fait référence à tout outil informatique dont le responsable de la sécurité n'a pas la connaissance ou le contrôle.

DomusVi est également HDS (Hébergeur de Données de Santé) qui est une certification concernant la sécurisation des données de santé. En outre, nous pourrions citer la mesure qui nécessite que les bases de données soient sauvegardées et stockées sur deux datacenters séparés de plus de 10km. Enfin, DomusVi possède un PCA (Plan de Continuité d'Activité) et un PRA (Plan de Reprise d'Activité) leur permettant d'anticiper et de réagir en cas d'incident sur leur entreprise. Ces plans prennent en

compte de nombreux cas allant du simple oubli de verrouillage d'une porte et allant jusqu'à la cyberattaque ou la destruction de locaux.

Après ces points essentiels au contexte de l'entreprise, concentrons-nous plutôt sur le périmètre de la mission : le Portail DomusVi.

Depuis 2017, DomusVi est un groupe dont l'activité majeure est la gestion de l'exploitation des entreprises d'aides aux personnes âgées. Un des outils essentiels de son SI est le « Portail DomusVi » qui est un portail d'applications web adressé au personnel des établissements ainsi qu'au personnel siège de l'entreprise. Ce portail héberge un ensemble d'applications spécifiques développées en interne et relie d'autre part des applications SAAS aux application du Portail DomusVi.

Dans un premier temps, la *Figure 8 : Page d'accueil du site portail DomusVi* montre l'accueil du site.

The screenshot shows the main dashboard of the DomusVi portal. On the left, there's a sidebar with navigation links like ACCUEIL, RÉSIDENTS, MARKETING, RÉSIDENCE, RH, GESTION, and OUTILS. The main area has several sections: 'Actualités' (News) with a link to 'L'OFFICIEL' (Official), 'REMERCIEMENTS' (Acknowledgments), and 'ÉVÉNEMENTS DANS VOTRE ÉTABLISSEMENT' (Events in your establishment). A large red box highlights the right-hand sidebar, which displays various performance metrics: 'flux organismes OUVERT' (2 action(s) commerciale(s) aujourd'hui), 'nouveaux contacts' (0), '221 actions dans le PAH' (117 en cours, 8 en validation, 106 en retard), 'visites à valider' (4), and 'actions à proscrire(s) aujourd'hui' (0). Below this is an 'Espace Managers' section. The bottom right features a grid of icons for various external services like COVID, net soins, SAP, etc.

Figure 8 : Page d'accueil du site portail DomusVi

Sur les coins supérieurs de l'image et donc du site se trouve le contexte : à gauche, l'établissement pour lequel nous consultons le site et à droite, le profil utilisateur sur lequel nous nous connectons. Sur cet accueil se trouve nombre d'informations sur l'état actuel de la résidence : le nombre de places restantes, le prix... mais aussi une partie prospection et fidélisation de clients (encadré en rouge) et un agenda d'activité de l'entreprise situé au milieu de la page. Il regroupe les visites et contacts avec les clients. Cette courte description de l'accueil permet d'introduire certains points du site mais il en regroupe beaucoup d'autres.

L'illustration ci-dessous (*Figure 9 : Système d'information du Portail DomusVi*) regroupe l'ensemble des thèmes que parcourt le site, les thèmes sont regroupés par bloc que l'on appelle domaine. Un domaine permet de découper la mission en termes de gestions fonctionnelle, technique et organisationnelle. Ils sont ensuite décomposés en applications qui regroupent des fonctionnalités de thèmes. Nous nous attardons sur certains mais cela permet de souligner la diversité et le nombre de sujets composant ce site.

Systèmes d'information

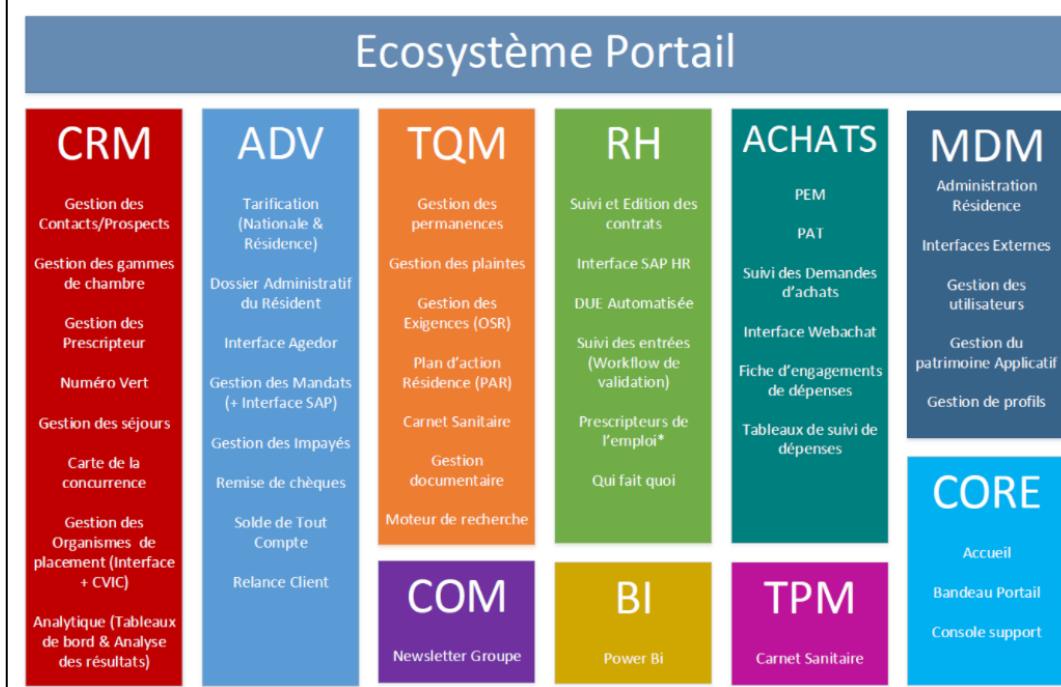


Figure 9 : Système d'information du Portail DomusVi

Nous allons rapidement parler de la partie CRM avec la gestion des clients et la prospection, le site a également développé des fonctionnalités importantes pour faciliter cette gestion aux utilisateurs. Cette partie peut être décrite par la *Figure 10 : Domaine CRM du portail Domusvi* plus bas. Les sujets centraux (encadrés en noir) regroupent la gestion des prospects avec la partie communication, la « Multi-Orientation » qui correspond aux placements des clients dans les établissements séniors. Puis la gestion des séjours qui illustre la gestion des contrats des personnes résidant dans les agences séniors.

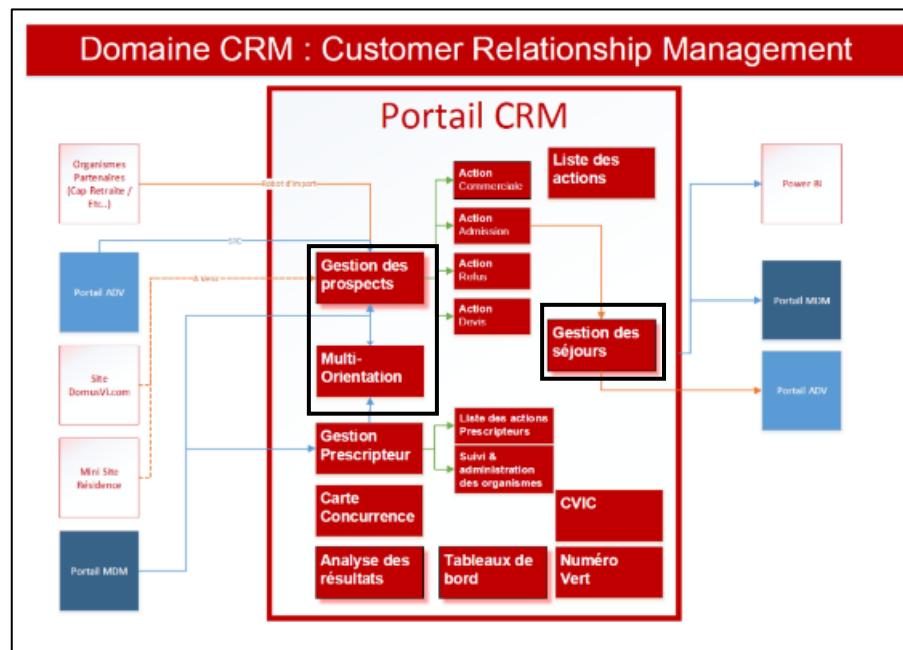


Figure 10 : Domaine CRM du portail DomusVi

Nous présenterons également le Domaine MDM : Master Data Management avec la *Figure 11 : Domaine MDM du portail DomusVi*.

C'est ce domaine qui gère l'administration des établissements et des sociétés. En effet, nous retrouvons les informations des groupes mais aussi la gestion des utilisateurs des résidences avec leurs gestions des accès et des droits. Depuis cette partie, une résidence peut gérer l'ensemble des accès aux différents pages (Gestion des Utilisateurs et Gestion des Profils de droits) de chacun des profils.

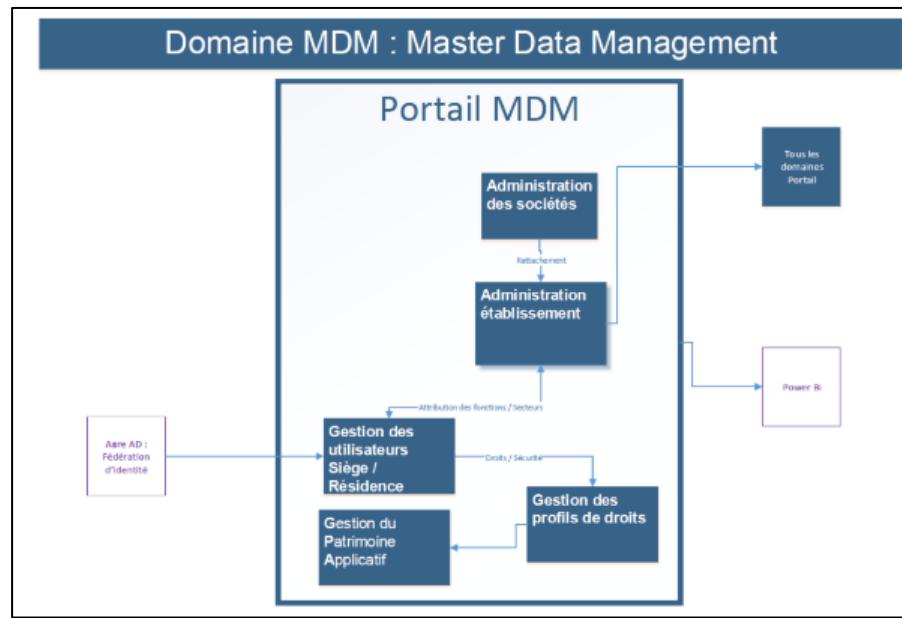


Figure 11 : Domaine MDM du portail DomusVi

Pour le reste du périmètre, nous pourrions citer le domaine couvrant la gestion des ventes : ADV (administration des ventes) qui gère la facturation des fournisseurs, des résidents et la gestion financières avec divers organismes. La gestion des Ressources Humaines (RH) est également présente sur ce site ainsi qu'une partie de la gestion financière des entreprises (ACHATS). Puis, il y a une partie qualité (TQM) traitant des plaintes et des normes auxquelles un établissement doit répondre.

Tout au long du document, nous utiliserons le terme application pour parler de l'ensemble des fonctionnalités regroupées autour d'un acronyme tel que « USR » gérant la partie d'accès et de droits des utilisateurs décrite un peu plus haut.

Après cette présentation du périmètre, nous évoquerons l'équipe Portail : pour rappel, elle se situe dans le groupe « Data, BI & Développement » se trouvant dans le pôle SI (encadré en rouge sur la *Figure 4 : Organigramme de DomusVi*). Cette équipe est responsable de la maintenance, la disponibilité et du développement de ce site web.

L'équipe de DomusVi chargée du projet de maintenance du site et du développement de nouvelles fonctionnalités est composée huit développeurs, d'un scrum master, d'un directeur technique, d'un consultant, d'un product owner et d'un business analyst. En complément de cette équipe Portail, il y a plusieurs équipes de testeurs fonctionnels (testeurs Métier) qui communiquent et font l'intermédiaire entre les clients finaux et l'équipe. Ils sont un point essentiel pour toute la partie validation des fonctionnalités ou corrections d'erreurs. Ils permettent de donner un retour rapide sur l'ajout de fonctionnalités et la résolution des bugs.

Nous utiliserons tout au long du document les termes « personnes Métier », « équipe Métier » ou « testeurs Métier » pour les désignés.

Nous pouvons terminer cette partie par un extrait de l'ordre de mission Ausy pour le client :

« La mission :

Les tâches comprennent la mise à jour de la sécurité du site, la connexion à une API, l'ajout d'éléments à la base de données, le déploiement sur des environnements, la mise à jour de l'UX Design, ainsi que la gestion des priorités liées à une cinquantaine d'applications, notamment les requêtes AJAX, GET et POST.

Les technologies utilisées dans cet environnement technique comprennent C#, React JS, JS, .NET Framework, Visual Studio, WebService, ISS, Azure DevOps, Oracle, PL-SQL, Windows, Agile Scrum, architecture Repository et Git. ».

Pour résumer, la mission concerne la mise à jour du site Portail DomusVi en termes de fonctionnalités, de données, de code et d'affichage. Il précise un nombre important de composants et détaille une « connexion à une API » signifiant la communication du site avec des applications externes au site. Enfin, on notera l'agilité Scrum qui est une méthode de gestion de projet.

Ces détails sont assez vastes et font appel à de nombreuses compétences et technologies que nous décrirons plus dans la partie [B. 4. Environnement de développement](#).

B. Organisation de l'équipe Portail

Blocs de Compétences :

- A2 – Pilotage d'un projet Système d'Information
- A3 – Management d'un projet informatique dans un environnement agile en collaboration avec les parties prenantes
- A5 – Développement d'une solution applicative spécifique et métier selon le projet de développement S.I.

Activité : Mise en œuvre et Suivi d'un projet, Management et Accompagnement des équipes projet, Intégration continue

Compétences choisies : A2C1, A2C2, A2C3, A2C4, A3C2, A3C3, A3C4, A3C5, A3C6, A3C7, A3C8, A5C6

1. Méthode Agile

L'équipe a adopté la méthode Agile Scrum : le principe de la méthode Agile se base sur des valeurs pour sa méthodologie : la communication avec les utilisateurs grâce à des retours réguliers et une communication forte des équipes internes. On retrouve aussi une certaine flexibilité et adaptation du projet : les nombreux retours permettent de réajuster en temps réel les décisions et la direction du projet. Une solution fonctionnelle doit être livrée pendant le projet et évolue tout au long du projet afin que toute l'équipe et les clients puissent avoir une visibilité sur l'avancement. Dans cet environnement, on retrouve toujours un « Product Owner » qui représente les clients et les utilisateurs. Il définit les priorités des tâches et prend les décisions sur les problématiques fonctionnelles. Un « Scrum master » est aussi un élément indispensable car il est le point central de

la méthode, il veille à la bonne communication, la bonne coordination dans l'équipe et au respect des principes Agile Scrum. Et enfin l'équipe de développeurs qui est chargée du développement.

Dans la méthode agile, on regroupe les tâches à réaliser dans un backlog ou périmètre produit qui correspond aux attentes de la solution. Pour l'avancement du projet, il y a un découpage du temps en période ou itérations de deux à quatre semaines : on parle de sprint. Lors d'un sprint, il y a aussi un regroupement de certaines des tâches du backlog dans un périmètre d'itération. Il traduit les tâches que l'équipe doit réaliser durant la période.

Dans ce processus, illustré par la *Figure 12 : Schéma de la méthode Agile Scrum* ci-dessous, nous pouvons voir sur la gauche un « Périmètre produit » ou « backlog » représentant le projet avec ses fonctionnalités non développées. Ce périmètre est ensuite divisé en « Périmètre d'itération » qui rassemble les tâches ou US (User Story) à réaliser durant un sprint. Lors de cette période, l'équipe s'engage terminer ces tâches. Enfin, ces périodes se répètent indéfiniment jusqu'à la fin du projet.

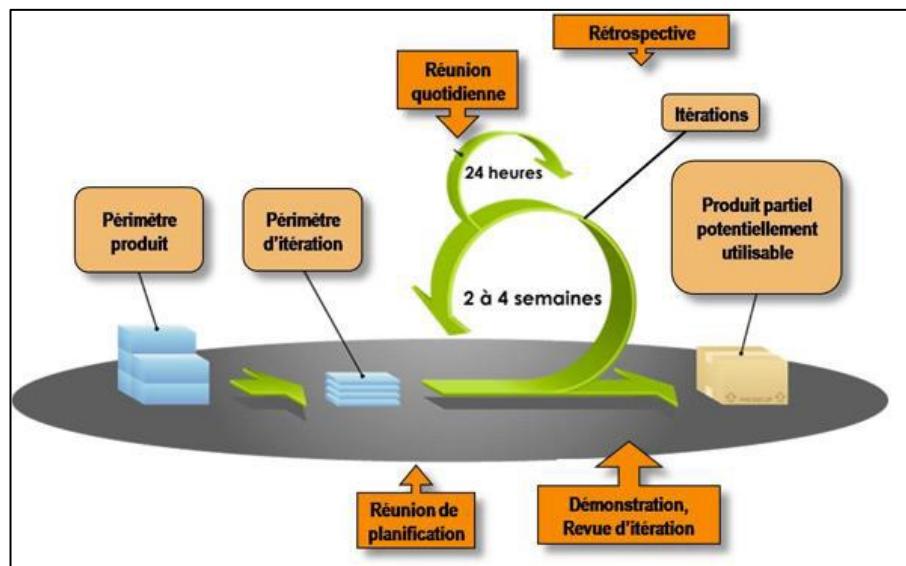


Figure 12: Schéma de la méthode Agile Scrum

Source : <https://all-it-network.com/methode-agile/>

Un sprint commence par une première réunion c'est le sprint planning : l'ensemble de l'équipe du projet planifie, définit et présente le périmètre d'itération. D'autre part, tous les jours un « Daily Meeting » est mis en place pour rendre compte de l'avancement des tâches. Chaque membre de l'équipe évoque les tâches qu'il a réalisé, les problèmes éventuellement rencontrés et les tâches qu'il fera dans la journée. Deux réunions concluent chaque période : la « Review d'itération » et la « Rétrospective ». La première permet de faire une démonstration des ajouts du sprint et de présenter le résultat à tous, et notamment à l'équipe validant les ajouts. La seconde est un retour sur le sprint qui vient d'avoir lieu afin que chacun fasse un bilan. C'est également un moment où il est possible de dialoguer sur des éléments de la méthode, des façons de travailler, de communiquer, d'échanger... dans le but d'améliorer l'efficacité de l'équipe.

Il existe également une réunion dite « poker planning » dont le but est d'évaluer l'effort de chaque tâche à accomplir. Elle est primordiale pour définir le périmètre d'un sprint. Elle regroupe l'ensemble de l'équipe scrum et c'est un moment d'échange afin de bien comprendre les enjeux de la tâche.

La fin des sprints est souvent synonyme de livraison de la version puisque les clients doivent voir l'évolution du produit afin de faire part de leurs retours, tout au long du projet.

2. Méthode Agile adoptée par l'équipe Portail

Lors de mon arrivée dans le projet Portail, la méthode Agile Scrum était récente et comportait la majorité des principes. Tout d'abord, l'équipe est rythmée par des sprints de trois semaines et les réunions sont toutes présentes :

Le sprint planning démarre les sprints avec une présentation des sujets importants du sprint et des tâches et US. Chacune des US sont présentés majoritairement par le Scrum Master ou le Product Owner. Chaque US apporte un échange de l'équipe afin de soit détailler certains points d'attentions ou incompris soit d'anticiper certains impacts qu'elle pourrait produire. Une fois, l'analyse des tâches terminée, chaque développeur se positionne sur des sujets.

L'équipe rythme également ses sprints par un DSM (Daily Scrum Master) tous les jours afin de faire état d'avancement des tâches.

Par ailleurs, l'équipe se réunit chaque lundi pour effectuer un « Start week », sauf les lundis de sprint planning. Cette réunion permet de revenir sur les objectifs du sprint afin de vérifier qu'il n'y a pas d'écart, elle traite également les sujets sensibles et critiques qui apparaissent au cours de la période. La discussion peut également amener certains échanges autour de bugs qui doivent être prises en compte et priorisées.

Cette réunion peut servir de moment pour chiffrer des US, telle la réunion de poker planning. Un poker planning se déroule de la façon suivante : après la présentation d'une nouvelle US, un échange s'ouvre sur le sujet, sur les points techniques pour comprendre les impacts, le développement attendu et sur les points fonctionnels en recentrant le besoin... Puis chaque développeur donne un chiffrage correspondant à l'effort qu'il pense que cette tâche nécessitera. Une moyenne est faite afin d'avoir une idée du temps nécessaire à la réalisation de cette US. Cette valeur est essentielle pour l'organisation des sprints.

Enfin, les sprints se concluent par une Rétrospective lors de laquelle nous revenons sur le sprint pour faire un bilan de la période et connaître l'avancement de toutes les tâches. Puis, nous faisons un tour de table pour connaître le ressenti de chacun sur le sprint écoulé. Il est fait état des bonnes pratiques, des problèmes rencontrés et des points d'améliorations. Il existe également des réunions ponctuelles qui peuvent être ajoutées lorsqu'il y a besoin de points techniques, qu'il a un nouveau projet sur le portail ou un point organisationnel.

Enfin, concernant les livraisons, l'équipe a adopté un principe de livraisons par regroupement de sujets : lorsqu'une ou plusieurs fonctionnalités ont besoin d'être ajoutée à une application. Elles suivent un process de développement, test et déploiement sur des environnements différents avant d'être implémentés sur la version en production accessible par les utilisateurs.

Commentaires

Cette organisation de la méthode agile a évolué lors de ces deux années de présence et continuera d'évoluer même après mon alternance. Nous allons donc décrire quelques points et parler des évolutions qu'il y a eu et qu'il est possible d'imaginer.

Tout d'abord, lors du sprint planning, nous avons évoqué « le positionnement des développeurs sur un sujet précis » : cela peut soulever une certaine dérive de la méthode. En effet, il ne devrait pas

avoir besoin de positionner les développeurs, suivant ses disponibilités, chacun devrait prendre la première US située dans le sprint backlog.

Cependant, le nombre important d'applications différentes sur le projet (plus d'une soixantaine) rend difficile voire impossible la maîtrise de l'entièreté du périmètre, que ce soit une maîtrise fonctionnelle : comment interagir avec l'ensemble du site, arrivé à tel cas précis ; ou une maîtrise technique : l'architecture du code et des données, le schéma des objets représentants les informations affichées... Pour faciliter l'avancement des tâches, lorsqu'un sujet a de nouveaux besoins, il n'est pas rare de voir un développeur ayant déjà travaillé dessus. Mais certaines fois, cette assignation précède la priorité des tâches et l'ordre de priorisation des US fait lors du sprint planning n'est pas respecté. Ce fait a été remonté lors de rétrospective et l'équipe met un point d'attention au partage de connaissances. Des démonstrations sont donc effectuées durant le sprint afin de présenter à l'équipe les nouvelles versions et les points techniques les plus essentiels.

Concernant les DSM, lors de mon arrivée dans l'équipe Portail, ce rituel commençait à 11h pour des préférences organisationnelles mais durait plus d'une heure, là où en théorie un daily meeting doit se limiter à une quinzaine-vingtaine de minutes. Ces réunions commençaient normalement : chaque personne de l'équipe énonçait son activité de la précédente journée mais les discussions dérivaient sur des échanges techniques ou fonctionnels qui prolongeaient la durée de la réunion. Ce point a été remonté durant les rétrospectives et au fur et à mesure des sprints, nous avons corrigé cela. Aujourd'hui, certains DSM peuvent dépasser 30 minutes mais cela reste minoritaire et peut s'expliquer par le nombre de participants dans l'équipe (presque une douzaine). D'autre part, avec la distance et le télétravail, cette réunion située avant la pause de midi peut devenir moins formelle et peut être le moyen de discuter d'autres sujets comme des retours clients, des validations de versions, de prochains projets, de modifications d'organisation...

Tout au long de mon alternance, les rétrospectives ont évolué pour être de plus en plus pertinentes et efficaces. Alors qu'au début la majorité était orale avec une prise de note légère, l'apparition de support pour la réunion a facilité les interactions. Tout d'abord, des systèmes de météo, d'humeurs et d'autres solutions ont permis d'exprimer et d'illustrer plus facilement le ressenti de chacun. Nous pourrions de même citer la prise de note qui est passée à un bloc note commun permettant à chacun d'écrire les points positifs et négatifs de la méthode. Concernant les points négatifs, des discussions ouvertes les solutionnent et des actions concrètes sont énoncées et mises en place pour corriger les problèmes potentiels. Il manque qu'une présence systématique des rétrospectives avec l'ensemble de l'équipe pour continuellement améliorer l'agilité.

Un des points que l'on pourrait également remettre en cause sont ces réunions « StartWeek » : elles pourraient être perçues comme chronophages et redondantes. Cependant, cette réunion permet de corriger les écarts et de recalibrer des ressources sur des sujets plus prioritaires. N'étant pas encore tout à fait mature concernant l'agilité Scrum, les Start Week sont un bon moyen de garder une certaine communication et de remémorer les objectifs attendus. Enfin, depuis l'augmentation de la présence du télétravail et du fait de la distance qui sépare l'ensemble de l'équipe, son existence en tant que cérémonie permet de garder le contact mais le contenu et l'animation pourrait être améliorés : certains points disgressent en point techniques.

En termes de KPI (Key Performance Indicator), l'équipe organise ses sprints autour de la vélocité. Dans la méthode agile, la vélocité est un indicateur de performances qui détermine la quantité de travail que peut fournir une équipe agile lors d'un sprint. Pour DomusVi, elle se calcule suivant le nombre de développeurs et leur disponibilité sur le sprint, le scrum master n'étant pas inclus. De

plus, dans l'équipe, un développeur est assigné à corrections de bugs, de même que le scrum master, il n'est pas retenu pour le calcul. En rattachant cette vélocité avec le chiffrage des US qui est expliqué plus haut, l'équipe peut adapter le nombre d'US des sprints. Enfin, lors de la réunion Review en fin de sprint, elle sert également de bilan.

Malgré l'ajout récent de la méthode et la distance, l'équipe possède une forte communication. Depuis mon arrivé, plusieurs logiciels sont en place et ont pris de plus en plus d'importances avec le temps :

Les Outils de communication et de l'agilité

Discord qui est une plateforme de communication, elle fonctionne avec un principe de serveur qui regroupe des salons de messageries instantanées et des salons de discussions vocales (rarement utilisés dans notre cas). L'accès à ses salons nécessite d'appartenir au serveur. Chaque membre de l'équipe a donc accès à ses salons et peut donc communiquer sur n'importe quel sujet. Dans notre cas (exemple sur la *Figure 13 : Salons de discussions présents sur le serveur discord DomusVi*), les salons possèdent des thèmes afin de regrouper les discussions sur un même sujet. Il y a des salons sur la gestion des bugs, les bugs urgents à prendre en compte, les déploiements (modifications d'un environnement du site) sur chaque environnement... C'est l'outil utilisé pour communiquer rapidement avec le maximum de personnes.

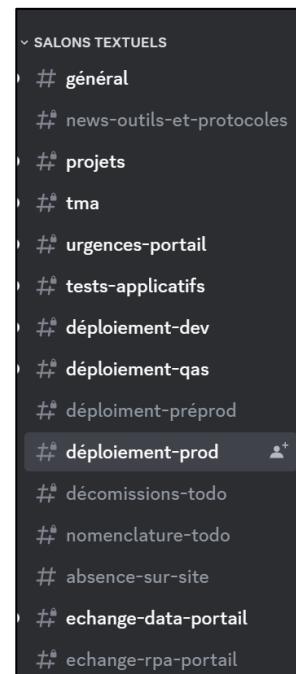


Figure 13 : Salons de discussions présents sur le serveur discord DomusVi

Microsoft Teams : Malgré l'accès à des salons de discussions vocales sur Discord, l'équipe utilise Teams. Ce logiciel est aussi une plateforme de communication mais plus orienté sur les discussions vocales. Au sein de l'équipe Portail, avec la distance séparant les différentes parties, toutes les réunions passent par cet outil : du DSM, au Start Week, en passant par les Sprints Planning, les Rétrospectives, les points techniques et même les appels individuels.

Azure DevOps anciennement TFS (Team Foundation Server), est un logiciel de collaboration qui permet la gestion de code source, d'intégration, de documents et de gestion de projet. C'est sur cet outil que tout le backlog, les sprints, les tâches et les bugs sont présents. Azure DevOps facilite énormément la méthode agile avec ses affichages, comme nous pouvons l'observer sur la *Figure 14 : Backlog de sprint de l'équipe Portail* plus bas.

En effet, l'image montre un exemple de sprint : nous retrouvons l'ensemble des tâches attachées à la période. Les premières avec un symbole d'avion rose correspondent aux mises en production que nous détaillerons plus tard, les symboles de fiches vertes font référence à des tests et les symboles de fiches bleus à des US. Pour chacune des tâches, il est possible de voir la priorité, l'effort prévu pour la faire, le statut et la personne assignée.

The screenshot shows a backlog board with the following tasks:

Order	Title	Priority	Effort	Assigned To
9	↗ [HRC-2.3.1] Impact CCO DUE	1		Hassan OULAOUDOUH
10	↗ [ETAB-1.15.2] Dispositions collectives	1		Hassan OULAOUDOUH
11	> [CRM-3.0.2] Retour à la suite US beauté		21	
12	> 🔍 [DAC-1.0] Etude technique - Etat des lieux du projet de migration DAR-3.0 (report du sprint-07)		100	
13	➡ [DAC-1.0] Evolution du MPD	2	12	Khalil REBAAM
14	➡ [CAC-1.4] Décaler l'heure du rdv dans la même journée	2	30	Hassan OULAOUDOUH
15	> [PIMS-1.4] Test du choix du workflow + La liste restrictive		1	
16	> [GPE-1.0] Retours QAS		1	
17	➡ [IMP-2.6] Retours métier : Ajout d'un champ commentaire		1	mohammed.lahbib-
18	➡ [HRC-2.3] Statut erroné lorsque la DPAE est KO			Hassan OULAOUDOUH
19	> [CSUP-1.3] Tests - Clôture de masse des prospects		1	Cyril ROBERT
20	> [ETAB-1.15.2] Administration des Dispositions collectives	2	13	Hassan OULAOUDOUH

Figure 14 : Backlog de sprint de l'équipe Portail

Il sert de support technique à la réalisation des US et aux corrections des bugs car il est possible de renseigner des informations sur chaque tâche. Il permet également de faire des liens entre elles et de gérer les tests. De plus, sur Azure Dev Ops, il est possible d'afficher la gestion du code source, que nous détaillerons plus tard, l'intégration et le déploiement continu.

Enfin, dans l'équipe Portail les membres ont tous le même poids d'influence et de pertinence. Malgré la distance et les réunions par vidéo, l'équipe a su s'adapter et grâce à un respect de chacun et une bonne organisation des prises de parole lors des réunions, le télétravail n'entrave aucunement l'avancée du projet. La présence des caméras est aussi un moyen de garder une certaine proximité et de rapprocher de l'équipe.

3. Processus de gestion du développement

Au sein de la mission, la gestion du développement peut être découpée en 2 parties : l'ajout d'une fonctionnalité et la gestion des TMA ou bugs. Notons qu'un environnement en informatique correspond à une version de l'application avec une installation et configuration précise.

Ajout d'une fonctionnalité

En plus de la méthode Agile, il existe un processus définissant le cheminement d'ajout d'une fonctionnalité. Nous pouvons l'illustrer par le schéma suivant : *Figure 15 : Processus d'ajout d'une fonctionnalité*.

Préparation

- Réception du Cahier des charges ou Expression de besoin
- Dépot de documents (Expression du besoin, specifications, cas de tests...)
- Préparation des outils de la méthode Agile
- Création de la version, des US, des sprints

Réalisation

- Etudes techinques et création des versions (applications, branches git...)
- Documentation technique (maquette, schéma, diagramme de BDD...)
- Développement et déploiement sur l'environnement de développement

Validation

- Tests techniques (revue de code, respect architecture et nomenclature...)
- Déploiement sur l'environnement de test
- Test fonctionnels
- S'il y a des tests non-valides => Correction des retours et redéploiements

MEP

- Mise à jour des outils et documents
- Préparation de la Mise En Production
- Mise en Production

Figure 15 : Processus d'ajout d'une fonctionnalité

Préparation

La réception d'un besoin se traduit immédiatement par la création de documentation exprimant le besoin lui-même, les spécifications fonctionnelles et autres documents d'informations. S'ensuit alors l'initialisation du projet :

La mise en place des outils de suivi et de projet, principalement avec Azure Dev Ops qui permet l'ensemble du suivi et son affichage. Nous pouvons voir un exemple sur la Figure 16 : Exemple de découpage des objets dans Azure DevOps ci-dessous.

Level	Category	Sub-Categories	Work Items
Top Level	Domaine		ACHAT
Level 1	Apps		DEP-Tableau de bord - Suivi des dépenses, GDA-Gestion des DA, PAT-PAT
Level 2	Version		[PAT-1.1] Intégration Facture et ERRD et Montants oubliés, [PAT-1.1] Ajout d'un bloc de suivi des factures sur... (under PAT-PAT)
Level 3	Fonctionnalité		[PAT-1.1] Suivi des montants oubliés, [PAT-1.1] Revue du calcul du solde si commentair...
Level 4	Mini Fonctionnalité		PEM-PEM, RFR-Réclamations Fournisseur

Figure 16 : Exemple de découpage des objets dans Azure DevOps

Le découpage suit des règles en partant des « Domaines » qui regroupent les applications par thème, les applications elles-mêmes « Apps » sur la figure. Puis les « versions » des applications qui regroupent les « fonctionnalités » à ajouter. C'est en divisant encore les "fonctionnalités que l'on retrouve les US ou « mini fonctionnalités ». Nous pouvons noter ensuite un premier « Macro-chiffrage » qui détermine le temps nécessaire à une analyse détaillée du sujet.

Réalisation

Après cette préparation, l'US d'analyse doit être placée dans un sprint pour sa réalisation. Cette étude doit permettre de mettre en perspective les modifications à venir, les impacts potentiels et surtout doit être documentée par des maquettes, des schémas et des diagrammes de modèles de données afin de comprendre au mieux les enjeux de la fonctionnalité. Ensuite, vient une création des US permettant la réalisation de l'étude menée. Elle est suivie d'un chiffrage de chacune de ses US afin de pouvoir prévoir le temps de développement dans un prochain sprint.

Dans ce processus d'ajout de fonctionnalités se trouve un principe de statut des US qui permet le suivi de leurs avancements.

Statut de l'US	Étape du processus
New	US connue et placé dans le backlog
En développement	US assigné à un développeur en développement
En Test DEV	Développement fini, en attente de revue de code et de validation technique et déployé sur l'environnement de Développement
Validé DEV	Revue de code faite en attente de test fonctionnel et déployé sur l'environnement de Qualité
Done	Validé par les tests fonctionnels
Removed	US dont le besoin n'est plus à jour

Figure 17 : Tableau des statuts des US

Comme le montre la *Figure 17 : Tableau des statuts des US* ci-dessus, toutes les tâches passent par ces états :

- « New » qui signifie que le besoin fonctionnel a été formulé et documenté.
- « En développement » comme son nom l'indique, la tâche a été placé dans un sprint, a été assigné et commencé par un développeur.
- « En Test DEV » lorsque la personne en charge a terminé la tâche et qu'il l'a implémentée sur l'environnement de développement, il doit d'abord passer par un autre développeur afin qu'il valide les modifications et qu'il teste sur l'environnement.
- « Validé DEV » survient après la validation du second développeur. Les modifications doivent alors être déployées sur la version de « Qualité ». C'est sur cet environnement que les tests fonctionnels vont être réalisés.

- « Done ». Le statut « Done » d'une US est très important en agilité car elle fixe les attentes en matière de développement, de tests et d'organisation d'une équipe pour l'accomplissement d'une tâche. Dans notre cas, il faut passer par toutes les autres étapes et valider les tests fonctionnels
- « Removed » est utilisé lorsque l'objet de l'US a été redéfini ou n'est plus d'actualité.

Validation

Dans l'étape de réalisation, il est demandé aux développeurs de créer eux-mêmes des cas de tests pour les étapes de tests avec les comportements attendus pour l'ensemble des cas. On peut voir donc que ces statuts permettent également de faire état des étapes de validation des tâches. Lors de ces étapes, tout manquement technique ou fonctionnel empêchera le passage au statut suivant et demandera de retourner aux étapes précédentes avec de nouveau un développement et des vérifications.

Mise En Production

Pour la mise en production, il est demandé de préparer une « fiche MEP » qui sera disponible sur le site Azure DevOps. Le faire sur le site permet de rapidement faire des liens vers les versions des applications mises à jour, avec les US, les bugs en lien avec les modifications.

Comme le montre la *Figure 18 : Exemple de fiche de Mise En Production* ci-dessous, une fiche de mise en production doit permettre de tracer et de visualiser les modifications apportées. Sur l'image, nous pouvons observer les applications concernées (colonne « CODE APP-version»), les deuxième et troisième colonnes permettent de faire les liens vers les version, tâches et bugs. Les deux colonnes encadrées en rouge sur l'image signalent quels fichiers ont été modifiés qu'ils proviennent du système de données (SQL et BDD), des fichiers des codes, des fichiers de paramétrages... Enfin, il y a une colonne qui permet de connaître la personne « en charge de préparation » : qui est responsable d'effectuer cette mise en production.

CODE APP-version	EPIC APP-version ou n°TMA	AVION	Applications + SQL	Paramétrage des accès et pools	En charge de préparation	Notes
ETAB	23207 [ETAB-1.17] sécurité associée + exports + clé API NS + Code GLN ● Done	27637 [ETAB-1.17] Ajout du code GLN + export + périmètre ● Done		Backend_establishement establishment.webclient SQL [1 - TMA-1.17] [2 - TMA-1.17]	@Pascal DEMARCHE	
HRC	31505 [HRC-2.4.2] Annexe jobvi ● En Production	31511 [HRC-2.4.2] Annexe jobvi ● Done	Backend_HRC.WebApi		@Pascal DEMARCHE	
TMA HRC	31637 [HRC] - Défaillance Forcer la génération du contrat Portail SAP ● Done		PKG_HRC : [1 - TMA-1.17]	Backend_HRC	@Pascal DEMARCHE	
TMA IMP	29281 [IMP] - Erreur JS dans la console (test chrome 101) ● Done			Facturation.webclient	@Pascal DEMARCHE	

Figure 18 : Exemple de fiche de Mise En Production

Concernant la MEP en elle-même, elle prévoit donc la mise à jour de l'environnement de production. Il a été convenu de toujours faire ces dernières en début de soirée lorsque normalement plus aucun utilisateur n'est censé être connecté, il est aussi convenu de ne jamais en faire le vendredi soir pour éviter de laisser plusieurs jours sans surveillance et sans support au cas où des bugs ou problèmes apparaîtraient sur le site.

Le déroulé d'une mise en production prévoit une sauvegarde de tous les fichiers qui vont être mise à jour. De plus, elle peut être précédée d'une mise en « pré-production » (sur un environnement similaire à celui de production) afin de s'assurer de la validité des modifications.

La gestion des TMA

Les TMA (Tierce Maintenance Application) relèvent de tout ce qu'on pourrait qualifier de « bugs ». Cela peut amener à l'arrêt d'une fonctionnalité, des problèmes de build, de compilation, des références d'url inconnues ou inexistantes. Mais il est question aussi des données, lorsque des données sont incohérentes, fausses ou absentes. A partir des causes que j'ai citées au-dessus, nous pouvons découper les TMA en 3 types : les Evolutions, Exploitations et les Bugs.

La première catégorie regroupe toutes les TMA dont les modifications sont assez critiques et importantes pour créer une nouvelle version de l'application ou de la fonctionnalité. Elles sont souvent repriorisées et ajoutées dans un sprint avec d'autres évolutions sur le sujet. Les Exploitations sont des TMA ne touchant souvent qu'aux données : il y a eu une mauvaise entrée par un utilisateur et il ne pas peut pas résoudre le problème lui-même. Souvent, le cas vient de donner incohérentes, manquantes ou fausses qui nécessite une intervention au niveau de la Base De Données (BDD). Enfin, les autres TMA rentrés dans la catégorie des bugs et nécessite une certaine attention afin de connaître leur impact et les criticités.

Après découverte de la TMA par un utilisateur ou un testeur Métier, elle doit être analysée pour connaître son impact fonctionnel et doit être priorisée : c'est à dire avoir une note de 1 à 4 sur sa sévérité et sa priorité. C'est à partir de cette notation que les tickets peuvent être pris en compte et résolus. Tout comme les US, les TMA suivent un schéma de statut pour pouvoir faire état à tout moment de leurs nombres, de leurs importances et de leurs avancements. D'après la *Figure 19 : Tableau des statuts des TMA*, les statuts suivent également un processus pour pouvoir avoir une visibilité.

Statut de la TMA	Étape du processus
New	TMA connue dont la priorité a été détaillée
A préciser	Après analyse, manque de donnée pour la correction de la TMA
En Test DEV	Correction apportée à la TMA en attente de validation
Recette	TMA en attente de validation déployée sur l'environnement de Qualité
Validé Métier	TMA validé par les personnes Métier en attente de Mise en Production
Done	TMA déployé sur l'environnement de production
Removed	TMA dont le besoin n'est plus à jour

Figure 19 : Tableau des statuts des TMA

On retrouve les statuts :

- « New » qui signifie que la TMA a été produite et qu'elle a été priorisée (principe de notation).

- « A préciser » comme son nom l'indique, ce statut traduit une demande de précisions de la TMA si elle n'est pas assez claire, qu'il manque des informations ou des documents.
- « En Test DEV » lorsque la personne en charge de la TMA a corrigé et dont les modifications ont été ajoutées sur l'environnement de Qualité, un autre développeur valide les modifications et teste sur l'environnement. Cette étape n'est pas toujours présente si le problème est minime (simple erreur sur une donnée, légère modification du code...)
- « Recette » : ce statut demande à un testeur fonctionnel de vérifier les modifications. En cas de refus, la TMA repasse au statut new et la boucle recommence.
- « Validé Métier » survient juste après la validation d'une personne Métier et sera donc en attente de mise en production.
- « Done » : Une TMA « Done » signifie qu'elle a été résolue et que les modifications sont présentes sur l'environnement de production.
- « Removed » est utilisé lorsque la TMA n'a pas été reproduite ou n'est plus d'actualité. Il est également possible qu'elle ait été résolue par une version d'application.

Ces statuts sont également visibles depuis notre outil Azure DevOps, il permet de regrouper l'ensemble des TMA et de les classer suivant plusieurs catégories. On peut voir un exemple sur la Figure 20 : Dashboard du suivi des TMA.

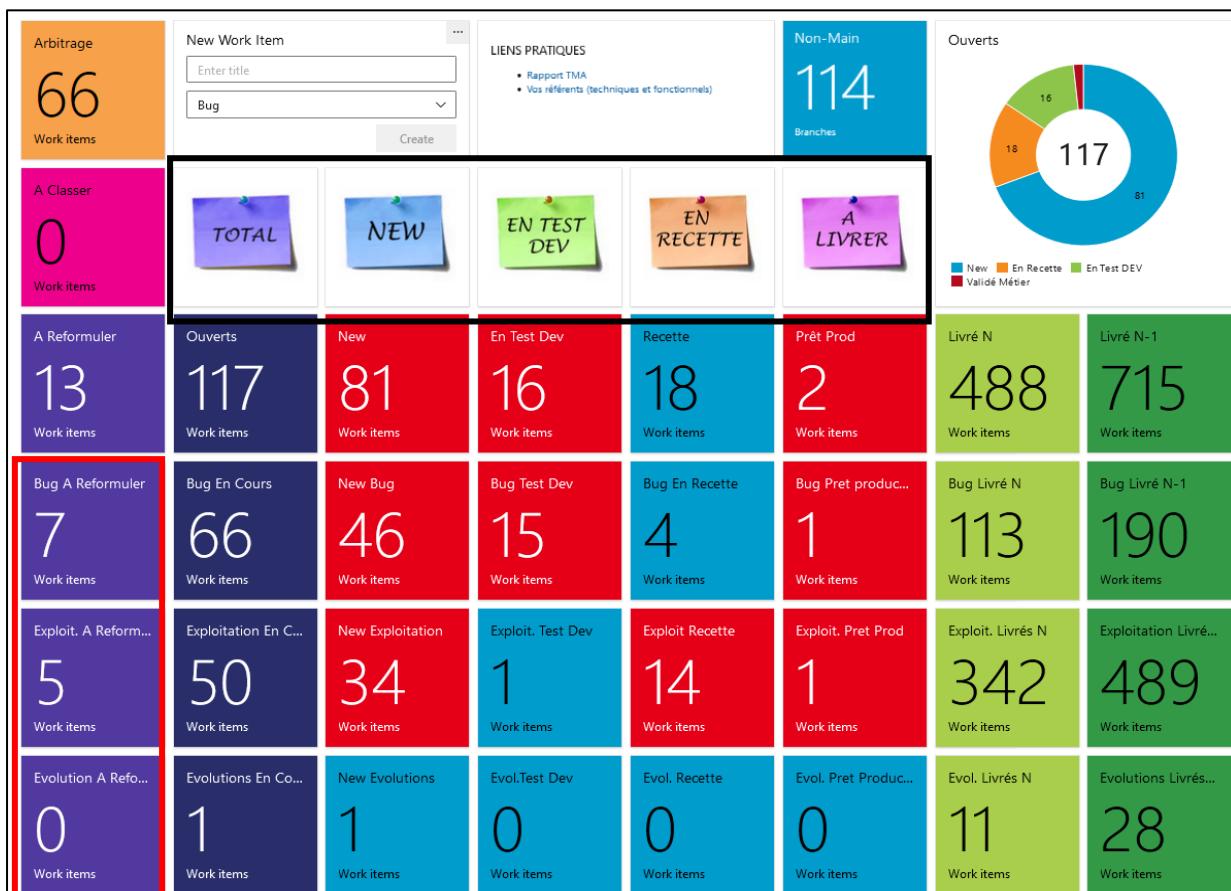


Figure 20 : Dashboard du suivi des TMA

Nous pouvons observer un découpage vertical en fonction des statuts dont nous avons parlé plus haut (encadré en noir) : « New », « En Test Dev », « En Recette » et « A Livrer » correspondant à « Validé Métier ». Et un découpage horizontal (encadré en rouge) en fonction du type de la TMA aussi expliqué plus haut : « Evolution », « Exploitation » et « Bug ».

Ce tableau de bord permet de visualiser facilement l'avancement et le nombre de TMA présentes sur le projet. C'est un des points d'attention de l'équipe qui peut nécessiter une réquisition de ressources si besoin. C'est un autre point de la méthode agile : savoir rediriger les ressources en fonction du besoin client. Il n'y a pas de règle explicite mais l'attention du scrum master et du product owner est aussi dirigée sur cette gestion des TMA. D'une part, une personne minimum par sprint est réquisitionnée pour la gestion des TMA, cette personne change de sprints en sprints. D'autre part il n'est pas rare d'avoir plusieurs personnes sur ce sujet, lorsque le nombre est jugé trop important que ce soit sur l'arrivée de nouveaux incidents (statut New) ou sur la validation technique (statut En Test Dev) car l'équipe a aussi cette charge de test à faire. Enfin, l'un des derniers cas possibles est lors de fin de sprints, lorsque tous les sujets ont été traités ou sont en traitement par une personne, le développeur ayant terminé sa tâche précédente va se diriger vers la maintenance du site et sur la résolution de bugs.

4. Environnement de travail

Comme vu plus haut, le Portail DomusVi est un ensemble d'applications développées en C# et en Javascript. L'ensemble des applications a été migré en .NET Framework 4.8 pour la partie back-end avec une architecture Repository. Le front-end quant à lui, se structure également par une application .NET Framework 4.8 pour faciliter les échanges entre les logiciels. Cependant, il est complété par un affichage qui varie suivant les applications. Le « React-Js » concerne les applications les plus récentes ou celles qui ont été migrées récemment. Pour les autres, le front-end peut être en Javascript ou en ASP.NET (qui lui remplacera la structure front-end). Enfin, l'ensemble des données sont stockées sur des bases de données Oracle. Pour le développement des applications, Visual Studio et Visual Studio Code sont les IDE majoritaires et pour la gestion de base de données, SQL Developer et Toad sont utilisés.

Environnement

Avant d'entrer dans les détails, je vais rapidement parler des environnements mis en place pour ce projet. Pour DomusVi, la définition d'environnement informatique comprend des configurations différentes pour le système d'authentification, les bases de données, les fichiers de codes et le serveur. Ils ne sont pas totalement différents mais leurs modifications sont gérées de manière indépendante.

Développement (DEV)
<ul style="list-style-type: none"> • Version commune à tous les développeurs
Qualité (QAS)
<ul style="list-style-type: none"> • Environnement pour les testeurs fonctionnels pour valider les modifications
Pré-Production (PPRD ou Pré-Prod)
<ul style="list-style-type: none"> • Ensemble du système de tests similaire à la Production
Production (PRD ou Prod)
<ul style="list-style-type: none"> • Ensemble du système mis en service pour ses utilisateurs

Figure 21 : Liste des environnements informatiques du Portail DomusVi

Pour ce système d'information nous avons donc quatre configurations différentes décrites par la *Figure 21 : Liste des environnements informatiques du Portail DomusVi* ci-dessus.

Cette variété de configuration permet également de posséder et de manipuler plusieurs versions du système, il est donc possible de résoudre des bugs tout en développant de nouvelles fonctionnalités.

Les Bases de données

Pour rappel, les bases de données sont stockées grâce au système de base données Oracle stockées sur des serveurs internes. Il y a deux schéma dans le système : MIGA et PORTAL qui font références aux bases de données possédant les mêmes noms ainsi que d'autres objets liés à elles (procédures, packages, vues...). Ces bases de données structurent donc les données du site et de l'ensemble du système du Portail. Elles possèdent donc des tables avec des colonnes permettant de sauvegarder les données. En complément de ces structures de données, il existe des fonctions et des procédures stockées avec les bases de données et permettant de modifier la structure ou les données elles-mêmes. Dans notre cas, l'équipe Portail traite utilisant le langage PL/SQL. Par ailleurs, l'ensemble des procédures sont regroupées dans des packages PL/SQL. Un package est simplement un regroupement de lignes de code destinées à être lues par d'autres programmes. Ils sont nommés suivants les applications web et fonctionnalités auxquelles ils sont rattachés.

A noter que chacune des bases de données et packages existent en versions modifiées pour chaque environnement cité plus haut.

Les Back-end

En remontant dans l'architecture, nous arrivons au back-end qui fait l'intermédiaire entre l'affichage d'un site (front-end) et la base de données. L'architecture repository dont nous parlions plus haut peut se schématiser comme sur la *Figure 22 : Schéma d'une architecture Repository* ci-dessous.

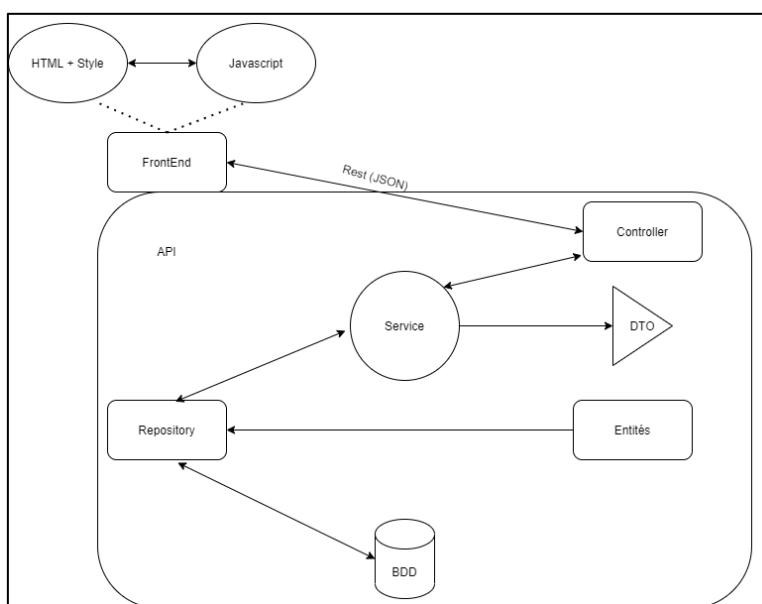


Figure 22 : Schéma d'une architecture Repository en lien avec un front-end

Dans cette architecture, le back-end se structure autour de cinq types de fichiers : les « Entités », les « Repositories », les « DTO » (Data Transfer Object), les « Services » et les « Controllers ».

Sur le schéma, nous pouvons observer les échanges entre les différents types de fichiers. Pour commencer, nous utilisons un outil : « dotConnect for Oracle » permettant de faire « le mapping objet-relationnel » des données relationnelles. C'est-à-dire qu'il permet de faire une certaine liaison entre les objets de la BDD (Base De Données) et le back-end. Ce qui facilite la gestion et les échanges des données entre le back-end et la base. Dans le système du Portail, il est utilisé afin de gérer l'interactions des procédures et des fonctions PL/SQL avec les objets du back-end : les « Entités ». Un exemple est visible sur la *Figure 23 : ModelDvi du projet CRM (1)* et la *Figure 24 : ModelDvi du projet CRM (2)*. Elles sont issues de l'IDE Visual Studio et représentent l'outil permettant l'interaction entre le back-end et la BDD.

La première image visible sur la droite représente les objets du schéma « PORTAL ». Dans la partie encadrée en vert sur l'image, on retrouve une partie des packages. Parmi eux, le « PKG_CRM » est déplié et affiche les procédures stockées à l'intérieur.

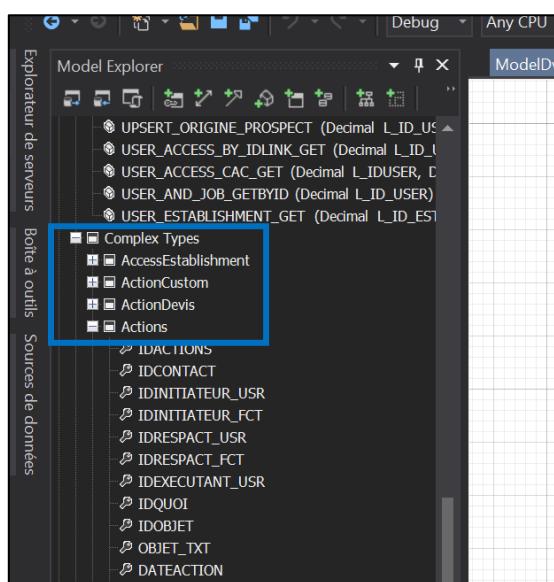


Figure 24 : ModelDvi du projet CRM (2)

Ainsi pour palier à la différence des objets provenant de la base de données et du back-end, docConnect for Oracle transforme le résultat des requêtes SQL en Entité.

Pour reprendre le schéma de la *Figure 22*, les copies des procédures sont stockées dans les fichiers de types « Repository » et les objets permettant la réception des données de ces procédures sont regroupées dans les fichiers « Entité ». La couche supérieure représentée par les fichiers « Services » se voit la tâche de transformer les « Entité » en « DTO » (Data Transfer Object : objets typés et plus faciles à manipuler) et peut également faire du traitement de données. Il ne reste donc que les

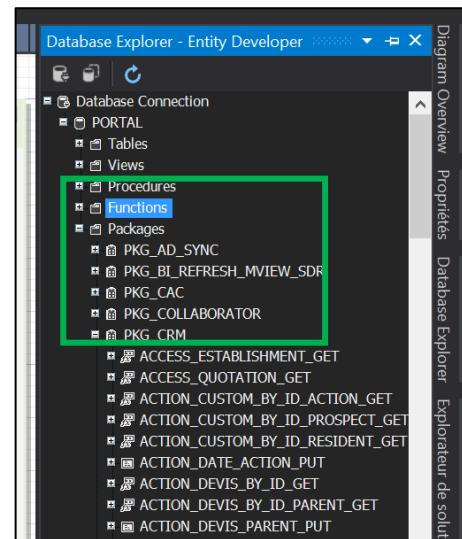


Figure 23 : ModelDvi du projet CRM (1)

Sur la *Figure 23 : ModelDvi du projet CRM (2)*, on retrouve l'affichage des objets du back-end en lien avec les procédures de la BDD. Les procédures correspondent à des requêtes PL/SQL, nous avons donc besoin d'objets permettant de recevoir le résultat de ces requêtes, c'est le rôle de ces « Complex Types » ou « Entité » encadré en bleu sur l'image. Le « Complex Type » « Actions » est déplié et nous pouvons observer qu'il contient des variables ou attributs qui font le lien entre les données d'une requête.

fichiers de type « Controller » qui peuvent également faire du traitement mais qui servent principalement d'intermédiaire pour le front-end grâce à des échanges de données via API REST.

Les Front-end

Plus haut, nous évoquions que les applications front-end peuvent avoir des technologies variées mais ici, nous allons détailler l'architecture d'applications la plus courante dans le projet : le React Js.

« React (aussi appelé React.js ou ReactJS) est une bibliothèque JavaScript libre [...]. Le but principal de cette bibliothèque est de faciliter la création d'application web monopage, via la création de composants dépendant d'un état et générant une page (ou portion) HTML à chaque changement d'état. » Source : <https://fr.wikipedia.org/wiki/React>

Pour expliquer plus précisément, cette bibliothèque utilise un cycle de vie de données pour afficher et mettre à jour ces données. React utilise deux notions pour cela : le « store » et le « state » (état en français). La première représente l'ensemble des données du site. La seconde peut se définir par les données d'une page ou un morceau de page à un moment donné : d'où le nom état.

Nous compléterons ces deux notions avec la *Figure 25 : Schéma du cycle de données en React Js* ci-dessous. Dans ce cycle de vie des données, on retrouve le « store » qui, comme dit plus haut, représente toutes les données. L'arrivée sur une page amène la création d'un state pour cette page. Par exemple, lors de l'affichage d'une page profil d'un utilisateur, cela correspond aux données de l'utilisateur : nom, prénom, numéro, email, adresse... Dans le schéma, cette arrivée sur la page est symbolisée par la fonction « getState » et l'affichage en lui-même par le bloc « UI » (User Interface).

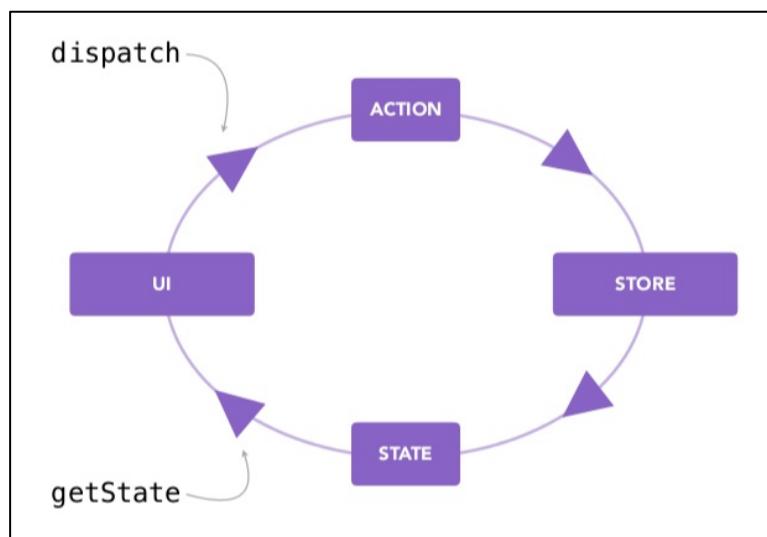


Figure 25 : Schéma du cycle de données en React Js

Lorsque l'utilisateur modifie un champ sur sa page de profil, l'adresse par exemple, il va pour l'instant modifier le « state » de la page profil et non le « store ». Cela évite d'avoir à recharger l'entièreté des données pour que le changement s'effectue. C'est ensuite lors de la validation et sauvegarde des champs (souvent avec un bouton « enregistrer les modifications ») qu'une nouvelle fonction va s'exécuter : « dispatch ». Son objectif est d'appeler une « Action » qui est une fonction aussi qui va modifier le « store » et donc l'ensemble des données du site.

Après l'explication du cycle des données, nous allons voir la structure du React au sein du portail DomusVi. Tout comme les applications back-end, elle se structure autour de différents types de

fichiers. Sur la *Figure 26 : Schéma du fonctionnement de React Js* ci-dessous, nous pouvons voir quatre nouveaux types de fichiers :

- Le « DataManager » dont le but est de gérer la communication avec les autres applications, sur le schéma il est rattaché avec un back-end.
- Les « Containers » comme leurs noms l'indiquent, font appel à d'autres types de fichiers : les « Component », pour gérer l'affichage des pages.
- Les « Components » contiennent donc des lignes de code pour gérer les objets à afficher : comme des champs d'édition, des icônes, des menus...
- Et enfin le « Reducer » qui regroupe des fonctions et permet la mise à jour du store. Dans la *Figure 26 : Schéma du cycle de données en React Js* vu plus haut, il pourrait s'intercaler entre l'« Action » et le « Store ».

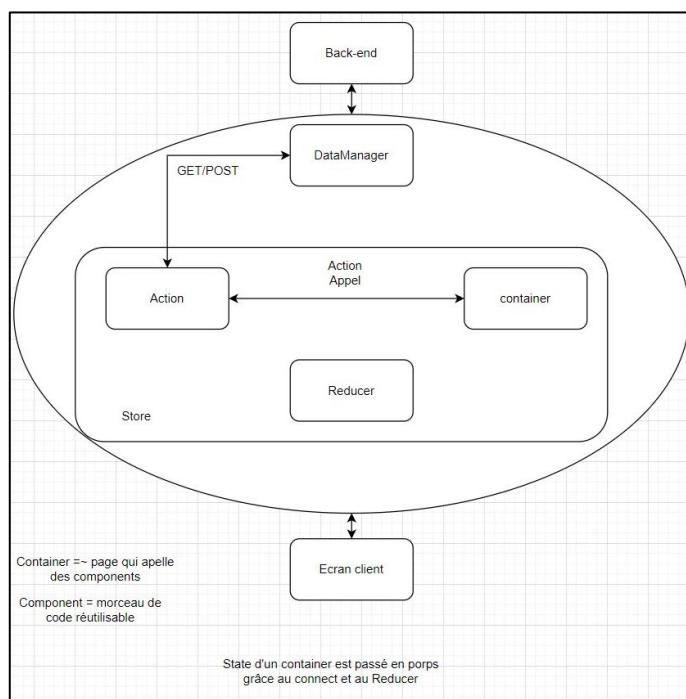


Figure 26 : Schéma du fonctionnement de React Js

Pour terminer sur la partie front-end, nous évoquons qu'elle se structurait avec une application .NET Framework. En effet, même lorsqu'un front-end se construit avec React-Js, il est créé avec une application .NET Framework. Cela permet de simplifier les interactions entre les front-ends et back-ends et de pouvoir gérer la sécurité de ces échanges. Pour permettre la compatibilité du .NET Framework et du React Js, nous allons expliquer le principe de bundle et minimisation :

- Le bundle est un principe de regroupement des différents fichiers en un seul. Pour une bibliothèque tel que React Js possédant plusieurs types de fichiers (Component, Container, Action, Reducer...), regrouper ces fichiers en un seul permet de limiter les échanges avec le navigateur et d'optimiser les performances.
- La minification quant à elle est aussi un principe d'optimisation des performances. Ce principe va simplifier en termes d'espace un fichier de code pour minimiser le nombre de caractères réduisant ainsi son contenu donc la taille du fichier.

Ainsi avec une bibliothèque complexe comme React, le bundle et la minification compressent l'ensemble des fichiers en un seul appelé « bundle » qui permet un gain considérable de

performances. Il ne reste qu'au front-end .NET Framework de gérer l'ensemble des échanges avec le navigateur pour afficher le site. Un exemple est montré dans la partie [D. 2. Projet de refonte du CRM](#), avec l'affichage du répertoire de fichier.

En termes de gestion d'environnements pour les applications back-end et front-end, il existe tout comme les bases de données, des versions de chacune des applications : DEV, QAS, PPRD et PRD. Chacune des versions du site est stockée sur un serveur que l'on peut modifier avec un déploiement. Avant de parler de déploiement, nous allons introduire l'outil Git qui est un gestionnaire de code source, il permet la gestion de multiples versions de fichiers de code grâce à une sauvegarde de ces versions. Il est donc possible de jongler entre plusieurs états des fichiers et de mutualiser ou fusionner ces versions. Pour l'équipe Portail, chaque application possède plusieurs versions indépendantes des environnements et facilitant les modifications de fichiers. Cet outil est utilisé pour le site Portail DomusVi et est stocké sur le site Azure DevOps utilisé par l'équipe. Cependant l'utilité d'Azure DevOps est également présente pour l'intégration continue et le déploiement continu.

L'intégration et le déploiement continu (CI/CD)

L'intégration continue est le principe consistant à vérifier l'intégrité du code lors d'un changement afin d'éviter toute régression (erreur suite aux modifications) du programme. On inclut en général la gestion de tests et l'amélioration du code source. La continuité signifie l'automatisation de ces actions et assure une certaine performance de l'équipe. Nous pouvons illustrer ce principe par *la Figure 27 : Schéma d'une chaîne d'intégration continue et de déploiement continu*.

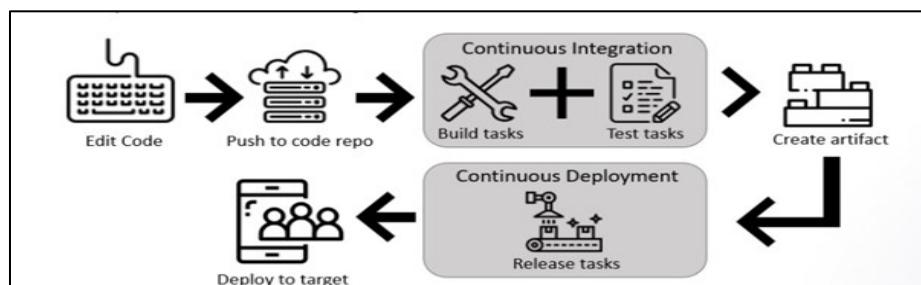


Figure 27 : Schéma d'une chaîne d'intégration continue et de déploiement continu

Source : <https://learn.microsoft.com/fr-fr/azure/devops/pipelines/get-started/pipelines-get-started?view=azure-devops>

Dans un processus traditionnel, un développeur modifie des fichiers de code puis va « push » c'est-à-dire qu'il va modifier une version du code stocké. Pour l'équipe Portail cela correspond à modifier l'une des versions disponible grâce à Git. Elle s'exécute automatiquement, après une modification et va donc procéder à des tâches définies sur Azure DevOps. Celles-ci comprennent une compilation, comme peut le faire un IDE classique et une tâche SonarQube qui est un outil d'analyse statique du code et permet certains tests automatisés. En cas d'échecs de la compilation ou de l'analyse SonarQube, l'intégration de la chaîne est arrêtée.

Ensuite, il est courant de voir un déploiement automatisé. Un déploiement est une modification d'un environnement, on peut traduire un déploiement continu par une modification d'une version lorsque qu'un développeur apporte des modifications à une version git correspondante à cet environnement. A titre d'exemple, sur la mission Portail, un changement du code source de la version Git portant le nom « develop » procédera à une modification de l'environnement de développement.

Commentaires

Cette présentation étant concise mais complexe, certains points méritent d'être éclaircis.

Tout d'abord, la gestion des procédures dans les gestionnaires de base de données peut sembler inhabituelle. En effet, gérer l'ensemble des fonctions et procédure PL/SQL dans des packages alors que l'outil Oracle permet et facilite la gestion de ces dernières peut être sous optimisé. Cependant, avec les deux bases de données et la quantité importante d'applications, donc de code et de procédures, il est important de pouvoir gérer la modification en temps réel. Lorsqu'une modification est apportée à une procédure ou une fonction PL/SQL, la compilation et le changement apporté à la version de l'environnement peut être regroupé par package, facilitant et optimisant son temps d'exécution.

D'autre part, il est à noter qu'il y a deux bases de données. En effet, la structure de la base de données MIGA posant quelques soucis pour certaines fonctionnalités, le schéma de donnée PORTAL a été implémenté afin de remplacer totalement MIGA et d'être autonome. Ce projet a été commencé il y a plusieurs années et nécessite encore quelques années. Mais chaque nouvelle conception prend en compte ce projet.

Concernant la partie qualité, l'équipe a des projets afin d'améliorer cette dernière. En termes de tests, des formations et de l'accompagnement au sein de l'équipe, il a été mis en place pour permettre à chaque développeur de créer et d'éditer des tests unitaires. A terme, l'objectif est que chaque application possède des tests unitaires. Les équipes fonctionnelles travaillent avec des procédures et des plans de tests afin de lister les attentes et avec des grilles de validations.

La partie intégration continue possède aussi des projets d'améliorations avec des formations aux tests unitaires, l'intégration en sera enrichie et l'analyse de code pourrait également être améliorée avec des règles plus précises. Le déploiement continu quant à lui reste limité, la majorité des déploiements se font actuellement avec les IDE. Cela rentre dans le processus d'évolution et de contrôle des tâches expliqué plus haut dans la partie [B. 3. Processus de gestion du besoin](#). Pour la partie développement, en plus du plan de tests, il y a des règles de nomenclatures précises qui sont validées lors des revues de codes. Elles concernent l'ensemble d'environnement de travail : noms des tâches sur Azure DevOps, nom des branches et commit de Git, nom de variables et fichiers .NET et React, nom des packages, des procédures, des tables et colonnes PL/SQL...

C. Projet MSAL :

Blocs de Compétences :

- A1 – Analyse et définition de la stratégie des systèmes d'information
- A5 – Développement d'une solution applicative spécifique et métier selon le projet de développement S.I.
- A2 – Pilotage d'un projet Système d'Information

Activité : Veille technologique, Développement sur différents environnements, Organisation de tâches

Compétences choisies : A1C1, A2C1, A5C5

1. Besoin du projet

Le site Portail DomusVi est donc un ensemble d'applications, qui pour s'échanger des données, utilise le principe d'API, il faut donc sécuriser ces appels. C'est ici qu'intervient ADAL (Active Directory Authentication Library), c'est un système d'appel relié à Azure qui permet la récupération de jetons de sécurité. Mais Microsoft a annoncé que « Toute la prise en charge et le développement [...] (ADAL), y compris les correctifs de sécurité, se terminent en juin 2023 » (source : <https://learn.microsoft.com/fr-fr/azure/active-directory/develop/msal-migration>) et a mis à disposition une nouvelle librairie d'authentification : MSAL (MicroSoft Authentication Library). Sur la *Figure 28 : Diagramme de comparaison de ADAL et MSAL*, on peut voir l'une des raisons de ce changement : le système d'authentification a ouvert l'étendu des comptes qu'elle permettait d'identifier.

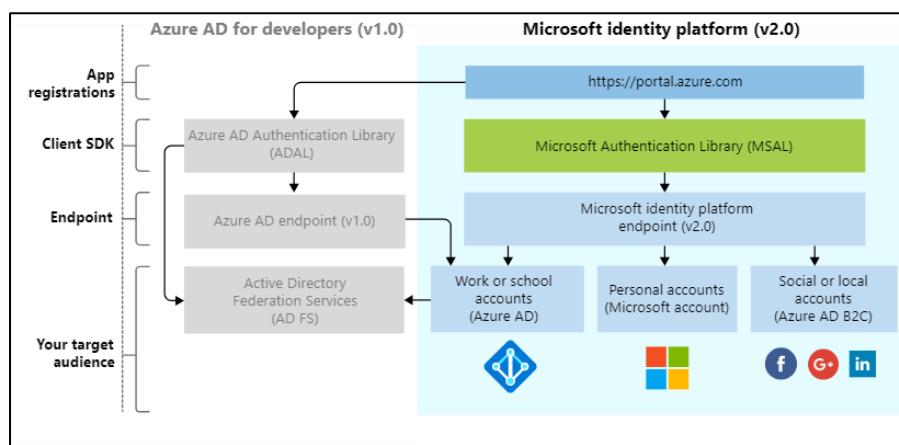


Figure 28 : Diagramme de comparaison de ADAL et MSAL

Source : <https://learn.microsoft.com/fr-fr/azure/active-directory/develop/msal-migration>

L'arrêt d'ajout de correctifs de sécurité est un problème des plus critiques car il met en péril la sécurité du système et peut ouvrir la porte aux cyber-attaques. Le pôle IT a donc réagi et a mis en place ce « projet MSAL » afin de mettre à jour l'ensemble des applications impactées. L'équipe Portail a donc été concernée par cette migration.

« La Bibliothèque d'authentification Microsoft (MSAL) permet aux développeurs d'acquérir des jetons de sécurité auprès de la plateforme d'identités Microsoft afin d'authentifier les utilisateurs et d'accéder aux API web sécurisées. Il peut être utilisé pour fournir un accès sécurisé à Microsoft Graph, d'autres API Microsoft, des API web de tiers ou vos propres API web. MSAL prend en charge de nombreuses architectures et plateformes d'application différentes, notamment .NET, JavaScript, Java, Python, Android et iOS. »

Source : <https://learn.microsoft.com/fr-fr/azure/active-directory/develop/msal-overview>

MSAL est donc une bibliothèque visant à remplacer ADAL afin de pouvoir garder le système d'authentification via des jetons de sécurité. Comparé à ADAL, elle permet la connexion de champs plus larges : compte utilisateurs du cloud. Elle gère, entre autres, le rafraîchissement du jeton de sécurité elle-même (un jeton peut être assimilé à un identifiant permettant la validité de l'identité d'un compte). Enfin, MSAL amène la possibilité de configurer et de gérer les erreurs plus facilement.

Avec le système ADAL, le système d'obtention de jeton de sécurité pouvait s'expliquer par la *Figure 29 : Schéma du fonctionnement d'ADAL sur le projet Portail*.

On retrouve sur cette figure trois types de logiciels : le « Front Web » correspondant au front-end des applications, l'« API (back-end) » qui regroupe tous les back-end des applications et l'API « Security Token » qui est une API externe permettant de gérer les jetons.

Note : API : (Application Programming Interface) est une interface logicielle qui permet de « connecter » 2 systèmes informatiques (majoritairement applications) afin permettre l'échange de données et de fonctionnalités.

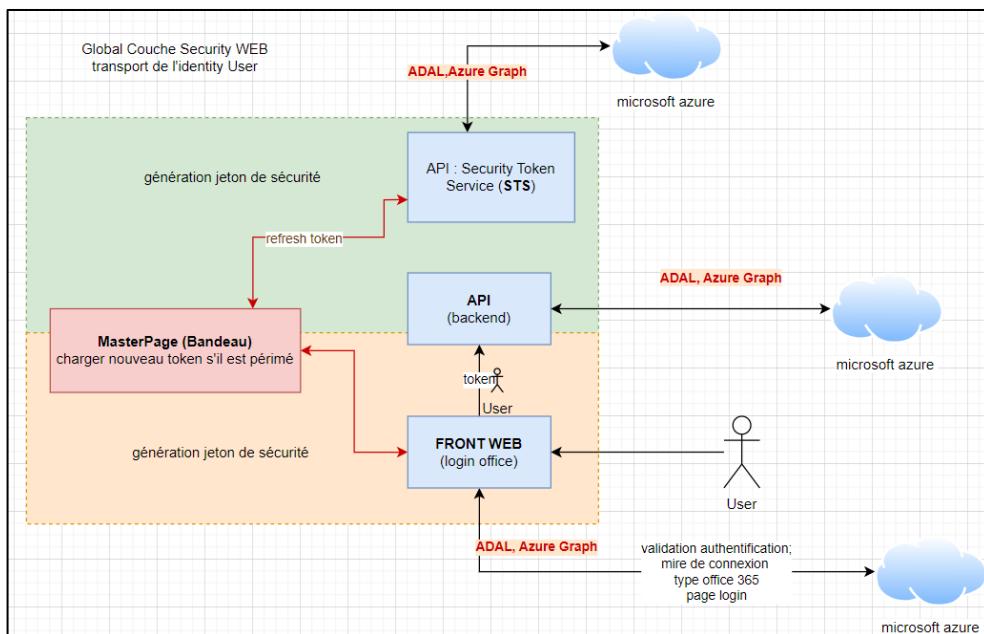


Figure 29 : Schéma du fonctionnement d'ADAL sur le projet Portail

Dans ce système, lorsqu'un utilisateur se connecte sur le site, il a besoin de s'identifier à son compte, ADAL utilise cette identification pour s'occuper de valider l'authentification et pour permettre aux échanges de données au sein du site de s'effectuer de manière sécurisée.

En utilisant le site, l'utilisateur charge des données qui proviennent des back-ends pour garder la sécurité lors de ses échanges entre le front-end et le back-end. Le jeton de sécurité est passé lors de l'échange et une vérification est également faite depuis le back-end.

Puis vient le moment où le jeton expire, c'est-à-dire que le temps alloué pour faire des échanges sécurisés est arrivé à son terme. C'est alors que pour rafraîchir le jeton, le front-end va demander à une certaine application : MasterPage de récupérer une nouvelle autorisation. Cette dernière va donc passer par l'API « Security Token » qui a pour mission de récupérer cette authentification depuis Microsoft Azure.

Le principe de d'ADAL est repris pour le fonctionnement de MSAL. Je l'expliquerai plus tard dans la partie [1. Fonctionnement de MSAL sur le projet Portail](#).

2. Processus du projet

Pour gérer le nouveau système, des recherches, des analyses et des études ont été nécessaires. Même si certaines n'ont pas abouti, des pistes ont été trouvées pour la conception du système.

On peut citer comme exemple l'analyse de la documentation Microsoft qui a permis de différencier les moyens d'acquisition du jeton de sécurité. (Exemple sur la *Figure 30 : Exemple d 'analyse du projet MSAL*).

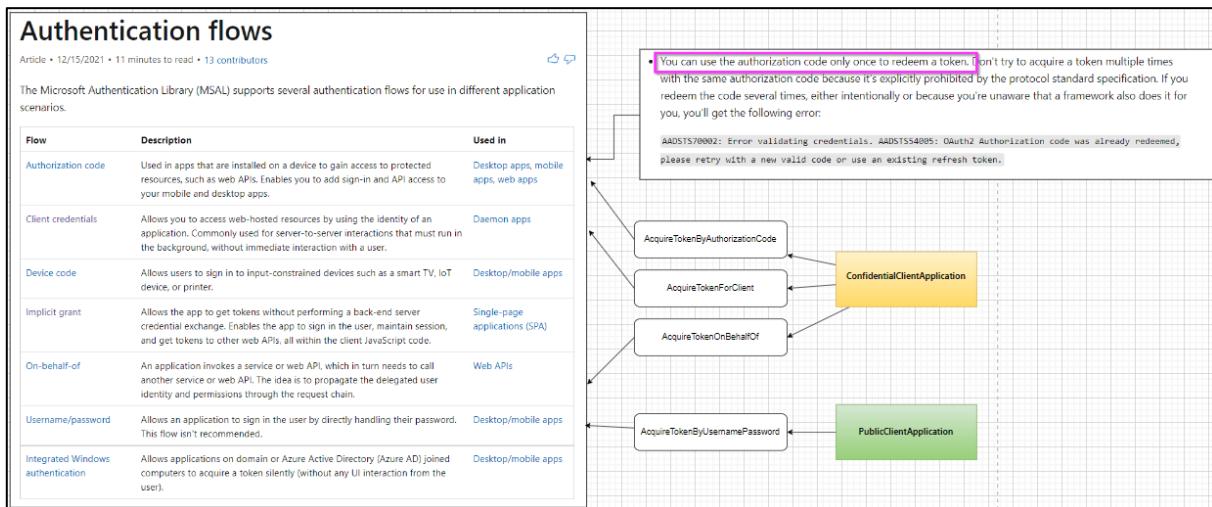


Figure 30 : Exemple d 'analyse du projet MSAL

D'autre part, pour garder le même fonctionnement qu'ADAL, on garde la même structure avec les trois types de logiciels. Il a fallu étudier pour chacun des cas le fonctionnement de MSAL comme le montre la *Figure 31 : Schéma des cas de fonctionnement de MSAL*.

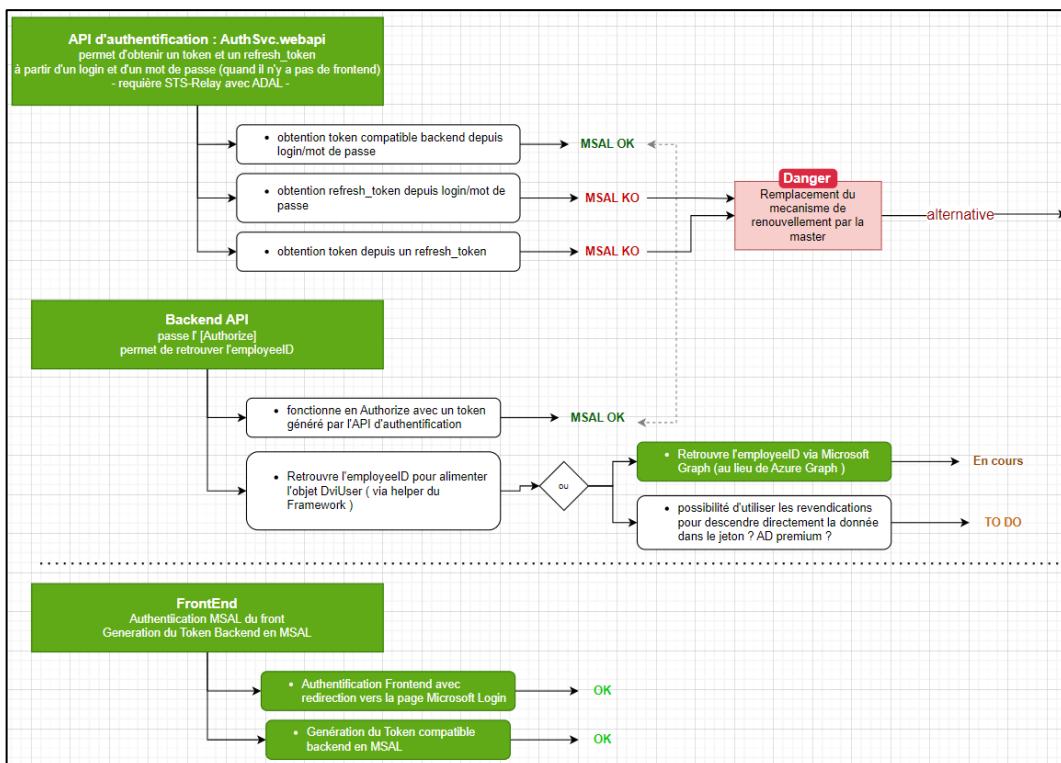


Figure 31 : Schéma des cas de fonctionnement de MSAL

Ainsi, les principales interrogations que soulèvent l'image concernent l'obtention du jeton de sécurité pour les types de logiciels depuis l'outil Microsoft. D'après ce schéma, on peut voir que les parties les plus délicates concernent le rafraîchissement du jeton là où l'authentification depuis les Front-end et Back-end ont trouvé une solution.

Ces premières analyses ont permis d'établir un MVP (Most Valuable Product) visible sur la *Figure 32 : Livrables attendus du projet MSAL*. Ce MVP traduit le minimum acceptable pour la validation de la version. C'est donc à partir de ce MVP que l'on peut approfondir les recherches.

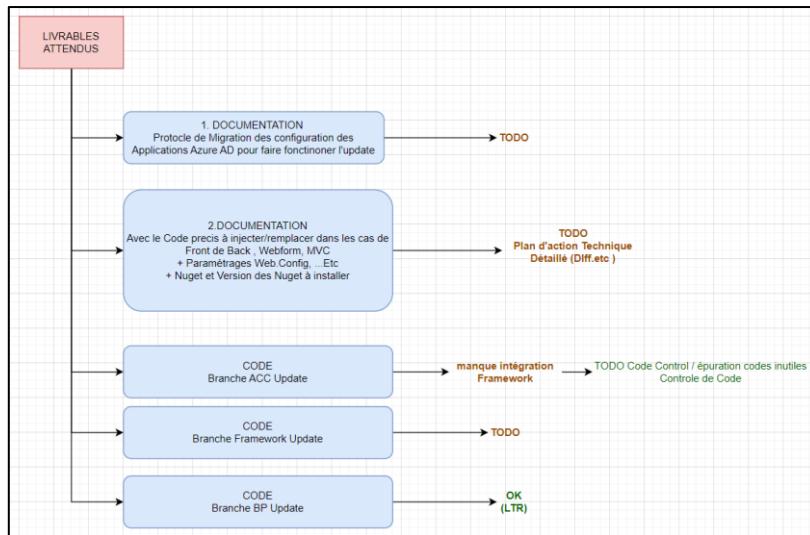


Figure 32 : Livrables attendus du projet MSAL

Nous pouvons voir qu'il y a des besoins de documentation sur les modifications sur le site Azure, les développements et les configurations suivant le types de l'applications : front-end, back-end, MVC... A noter que ACC et BP sont les abréviations des applications noyaux du site, gérant entre autres la page d'accueil du site et le noyaux des fonctionnalités communes à toutes les applications.

Une fois, cette étape d'analyse et d'étude terminée, le développement s'est découpé en deux parties : une première concernait toutes les modifications pour la gestion de la récupération des jetons ainsi que de leur rafraîchissement et la seconde comportait toutes les petites modifications à apporter sur toutes les applications (front-end et back-end) pour faire les appels à cette première partie de développement.

Après que la première partie ait été effectuée par une paire de développeurs, la seconde partie a demandé l'intervention de l'ensemble de l'équipe Portail. C'est là que la documentation technique a été très précieuse : elle décrivait l'ensemble des étapes à suivre pour les modifications : la création des branches Git, la récupération du code source et de tous les outils nécessaires, des modifications de code jusqu'à l'exécution du premier test. La documentation détaillait également l'ensemble du processus pour chacun des types d'architecture, des langages et des frameworks des applications.

L'avancée des modifications application après application a été suivie grâce à des réunions et grâce à un tableau Excel. En suivant les instructions de tests, nous pouvons également lister pour l'ensemble des applications la personne en charge, son avancement (étape de développement, étape de test techniques, étape de tests fonctionnels...) comme la gestion des statuts, vu dans la partie [B. 3. Processus de gestion du besoin](#). Ce fichier répertorierait également l'information des environnements sur lesquels la bascule vers le système MSAL avait déjà été réalisée. Enfin, les membres de l'équipe référaient également les erreurs et problèmes qu'ils rencontraient pendant le processus.

Ces problèmes pouvaient être de simples questions techniques et les erreurs étaient souvent signe d'un problème d'authentification rencontré lors de test.

Une fois l'ensemble des applications mises à jour, il a fallu procéder à un premier déploiement sur un environnement de Test (Qualité) pour permettre l'exécution de tests plus approfondis puis enfin les mises en production de l'ensemble des applications.

3. Fonctionnement de MSAL sur le projet Portail

Après ces dernières explications, il est temps de parler du fonctionnement de MSAL sur le périmètre Portail. Le principe est de pouvoir faire un appel au site Microsoft afin d'obtenir le jeton de sécurité. Nous pouvons illustrer le processus par la *Figure 33 : Processus d'obtention et de rafraîchissement de MSAL* suivante.

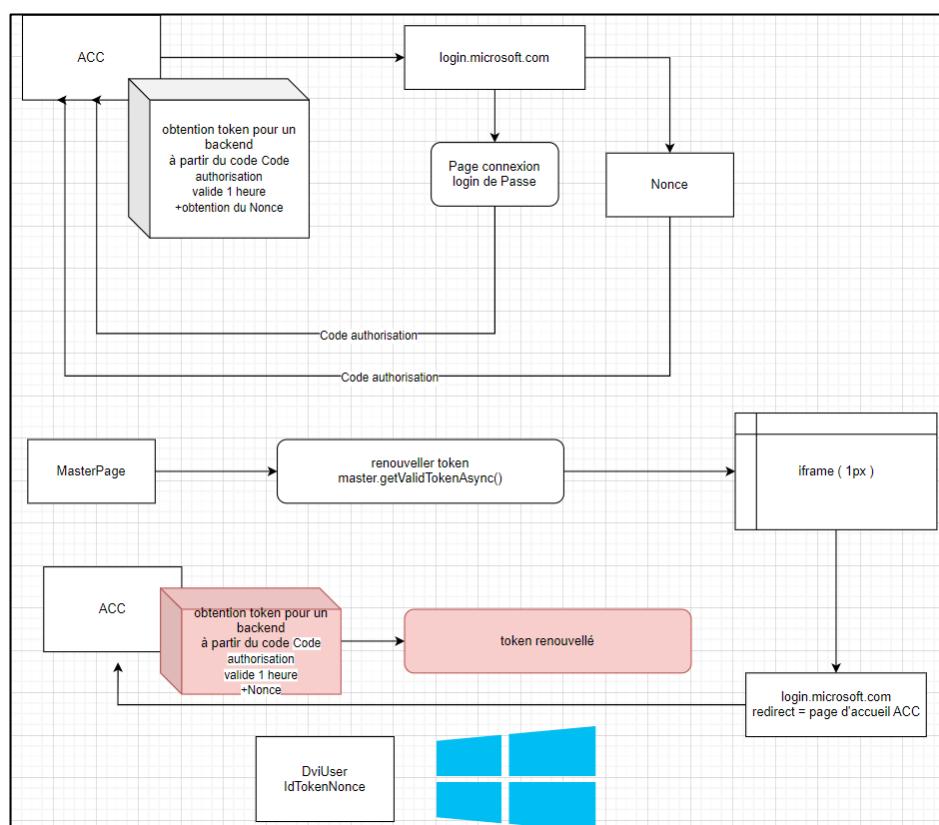


Figure 33 : Processus d'obtention et de rafraîchissement de MSAL

Sur cette image, il y a deux boucles, la première explique le fonctionnement de l'obtention. A noter que l'on retrouve les deux applications noyaux ACC et MasterPage dont nous avions parlé plus haut. C'est en effet l'application ACC qui est appelée pour l'obtention du token. Dans la boucle, on peut voir que ACC fait un appel au site Microsoft « `login.microsoft.com` » pour la demande d'authentification. Le site si ce n'est pas déjà fait, va demander une connexion via identifiant et mot de passe afin de s'assurer que l'utilisateur est enregistré sur le site DomusVi. Lorsque la connexion est validée, elle permet d'obtenir une authentification pour un certain temps ainsi qu'un objet « Nonce » qui est un identifiant à usage unique qui servira pour le rafraîchissement.

En effet, une authentification est pour une durée limitée, nous ne pourrions plus parler de sécurité si un jeton n'avait pas de limite de temps, il suffirait d'obtenir une seule authentification pour toujours être accepté. Cette durée d'expiration explique pourquoi il est nécessaire de rafraîchir le jeton.

La boucle inférieure quant à elle, montre comment le rafraîchissement s'opère. Ici, c'est l'application MasterPage qui déclenche le processus, comme elle est le noyau du site, elle est appelée par l'ensemble des autres pages et exécute en continue des fonctions. Cela lui permet donc de pouvoir savoir si un jeton est sur le point d'expirer et, si besoin, de relancer la boucle. Le principe est de créer sur la page un pop-up invisible (1 pixel sur 1 pixel) pour exécuter l'appel au site Microsoft qui lui redirige sur l'application ACC. Cette dernière vérifie l'objet « Nonce », récupéré lors de la connexion, ainsi que l'identification du compte afin de valider à nouveau l'authentification. Même lors du rafraîchissement, un nouvel objet « Nonce » est délivré dans le but de pouvoir réautoriser l'authentification plus tard.

Au niveau du code, cela se traduit par des appels avec le site Microsoft, l'application en cours d'utilisation, les deux applications noyaux ACC et BP et avec la réception des réponses. Il y a bien sûr des traitements de ces données et une sauvegarde pour pouvoir l'utiliser à tout moment et pouvoir gérer le rafraîchissement. Sur la *Figure 34 : Code de gestion de l'authentification*, nous pouvons observer une partie du code gérant l'obtention du jeton de sécurité. Les commentaires sont assez explicites : les deux premières lignes permettant de récupérer les informations du compte. Ensuite, on récupère les objets nécessaires : « Nonce », token pour le front-end et token pour le back-end.

```
// Récupération des information du compte
var accountAsync = Task.Run(async () => await clientapp.GetAccountAsync($"{tokenResult.UniqueId}.{tokenResult.TenantId}"));
var account = accountAsync.Result;

// Obtention du token pour la récupération de l'employeeId lié au compte.
var tokenCurrentAccount = Task.Run(async () => await clientapp.AcquireTokenSilent(scopes, account).ExecuteAsync());
var tokenCurrentUser = tokenCurrentAccount.Result;

#region GET Nonce
// Nonce: La valeur à usage unique correspond au paramètre inclus dans la requête /authorize d'origine au point de distribution d'émission.
// Cette valeur est utiliser lors du renouvellement du token dans getValidTokenAsync dans master.js de la MasterPage

_logger.Trace("AuthorizationCodeReceived -> IdToken: " + tokenCurrentUser.IdToken);
NonceFront = TokenManager.GetNonceFromIdToken(tokenCurrentUser.IdToken);
_logger.Trace("AuthorizationCodeReceived -> NonceFront: " + NonceFront);

#endregion GET Nonce

// variable global définit pour récupérer le token de l'utilisateur courant.
tokenValid = tokenCurrentUser.AccessToken;

_logger.Trace("AuthorizationCodeReceived -> Token Front: " + tokenValid);
//Obtention du token pour permettre la consommation du Backend
string[] scopesBackend = new string[] { backendScope };
var tokenSilent = Task.Run(async () => await clientapp.AcquireTokenSilent(scopesBackend, account).ExecuteAsync());

var token = tokenSilent.Result;

_logger.Trace("AuthorizationCodeReceived -> Token Back: " + token?.AccessToken);
TokenInfos tokenInfos = new TokenInfos()
```

Figure 34 : Code de gestion de l'authentification

L'une des autres problématiques du projet est le développement ou plutôt la mise à jour du logiciel permettant la récupération de jeton de sécurité : AAD Console. En effet, l'équipe a créé un logiciel uniquement pour la récupération de token de connexion. Ce dernier est indispensable lors de développement d'API ou de tests comme l'ensemble des applications nécessitent l'authentification pour manipuler les données. La *Figure 35 : Logiciel AAD Console* nous montre un aperçu du logiciel : les informations voulues se récupèrent grâce à un clic sur les boutons en bas. Il est également possible de choisir l'environnement pour lequel on souhaite un jeton, disponible par le biais de la liste déroulante dans le coin supérieur droit.

Web Api - Developer Tool

AUTHENTIFICATION STS-RELAY AAD SCHEMA EXPLORER TINY-POSTMAN

Authentification : NATIVE CLIENT => WEB API : 1 process authentification requis

AADConsole.DEVPRI.config
AADConsole.DEVPRI.config
AADConsole.QAS.config
AADConsole.PROD.config

MSAL

USER - ID DE L'OBJET	<input type="text"/>
USER - LOGIN AAD	<input type="text"/> ausy@ausy.com
USER - PASSWORD AAD	<input type="password"/>
ADD - INSTANCE	<input type="text"/> https://login.ausy.com/adfs/odata/v2/
ADD - TENANT	<input type="text"/> 88888888-0000-0000-0000-000000000000
ADD - ID TENANT	<input type="text"/> 00000000-0000-0000-0000-000000000000
WEB API - URI ID D'APPLICATION	<input type="text"/> https://ausy.com/api/auth/refreshToken
NATIVE APP CLIENTE - ID CLIENT	<input type="text"/> 00000000-0000-0000-0000-000000000000
FRONT APP CLIENTE - ID CLIENT	<input type="text"/> 00000000-0000-0000-0000-000000000000
FRONT APP CLIENTE - SECRET KEY	<input type="text"/> 00000000-0000-0000-0000-000000000000
TOKEN - HTTP-HEADER	<input type="text"/>

TOKEN **RECHARGER PARAMETRES** **COPIER LE TOKEN HEADER**

Figure 35 : Logiciel AAD Console

Cependant, le projet ne s'est pas vraiment arrêté là, tout comme les autres projets informatiques, on retrouve des évolutions et des bugs. L'un des problèmes les plus récents concerne la gestion du rafraîchissement : lorsque l'utilisateur n'utilise pas le site pendant un certain moment, des erreurs d'authentification empêchent toute action sur le site. Cet incident nécessita beaucoup de temps et d'analyse pour trouver la cause.

Certaines pistes n'ont pas trouvé la solution mais pendant le printemps 2023, la gestion du temps de validité entre les différents jetons est mise en avant. Pour rentrer dans les détails, le jeton de sécurité du front-end et du back-end n'ont pas le même temps de validité. Le rafraîchissement n'est donc pas forcément fait au même moment. Ainsi ce décalage crée un problème qui est illustré par la Figure 36 : Schéma du processus d'erreur de jeton de sécurité.

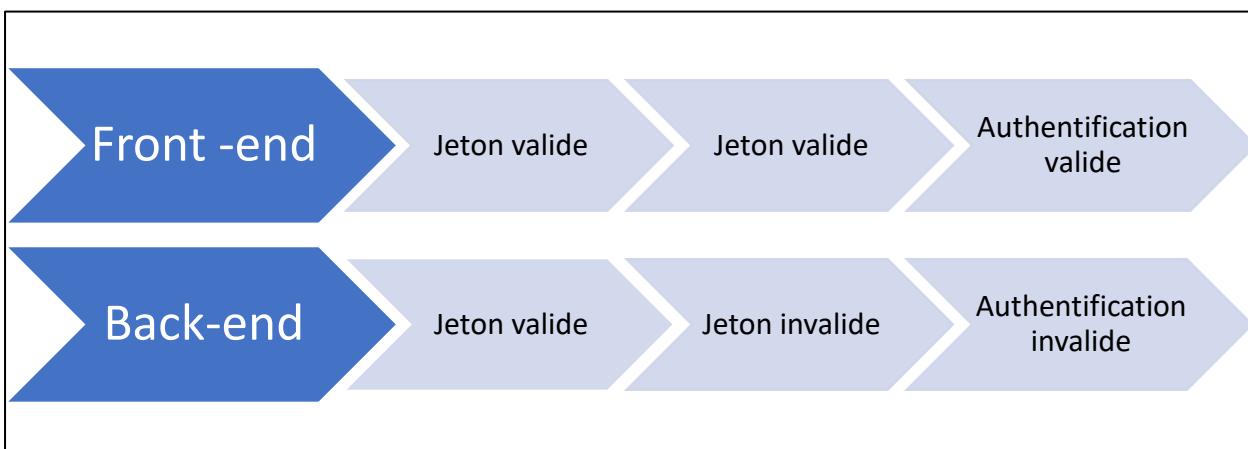


Figure 36 : Schéma du processus d'erreur de jeton de sécurité

Ainsi lorsque le jeton est valide sur le front-end, le site ne demande pas de nouveau jeton et lorsqu'une action demande un appel au back-end alors que lui possède un jeton invalide l'erreur survient.

Le développement n'est pas totalement fini mais la solution a nécessité de nouvelles recherches. Elle pourrait être résolue par la gestion du temps de validation des jetons définit lors de leur demande à l'outil Microsoft. Un projet est donc en cours pour gérer ce problème.

D. Projet CRM

Blocs de Compétences :

- A1 – Analyse et définition de la stratégie des systèmes d'information
- A2 – Pilotage d'un projet Système d'Information
- A5 – Développement d'une solution applicative spécifique et métier selon le projet de développement S.I.

Activité : Analyse des besoins, Développement sur différents environnements

Compétences choisies : A1C2, A2C8, A5C1, A5C2, A5C3, A5C4

1. Présentation du CRM

Dans la présentation du site faite dans la partie [A. 3. Le cas de DomusVi](#), nous avions évoqué le domaine CRM qui gère la partie prospection, fidélisation et la gestion des contrats des clients. Ce domaine a évolué plusieurs fois au fil de l'alternance et cette partie vise à expliquer quelques-uns des projets.

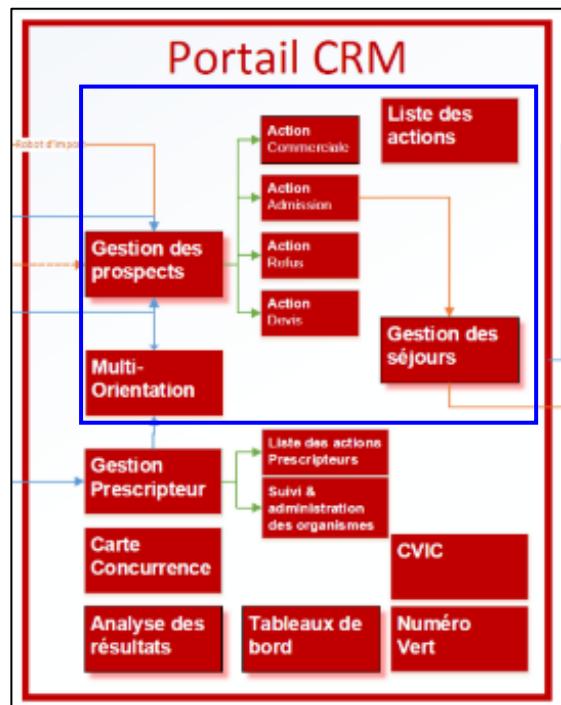


Figure 37 : Domaine CRM du portail DomusVi

En réutilisant la *Figure 37 : Domaine CRM du portail DomusVi*, nous nous intéresserons à la partie supérieure encadrée en bleu sur l'image.

La « gestion des prospects » contient la partie affichant les données des clients et la partie « Multi Orientation » met en contact un client avec plusieurs établissements. La gestion des prospects est complétée par la « Liste des actions ». Une action permet d'historiser les échanges et évolution du client, elles sont découpées en plusieurs catégories :

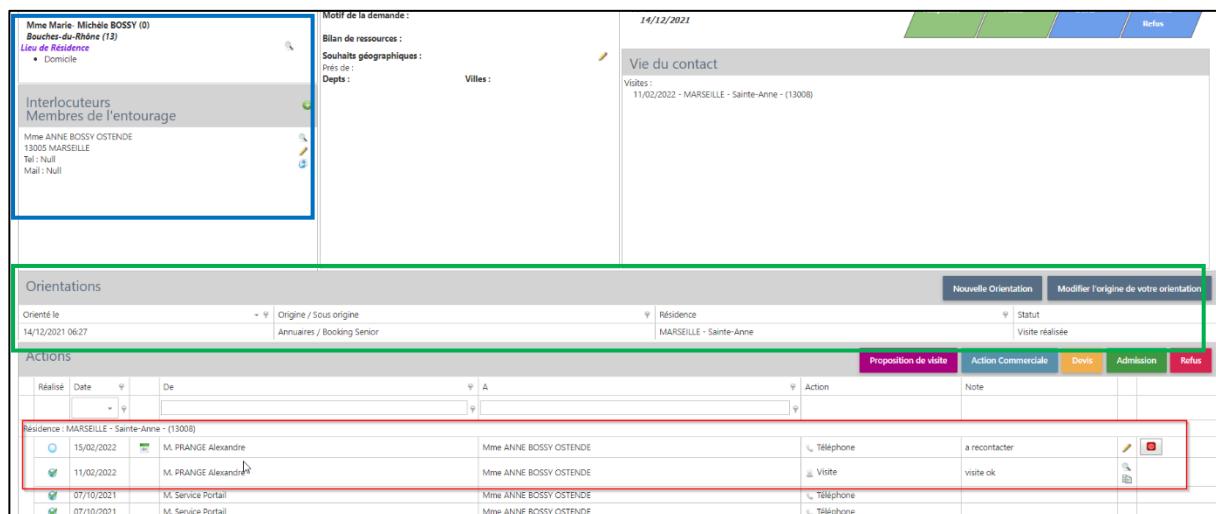
- Les « actions Commerciales », avec les contacts à distance des clients (téléphone, mail, courrier) et les contacts physiques avec les visites des établissements.
- Les « actions d'Admissions » dans les résidences.
- Les « actions de Refus » : refus des autres actions

- Et les « actions de Devis » qui permettent d'estimer le coût des services d'hébergement ou d'aide.

Une fois l'admission du client dans une résidence, la « gestion du séjour » prend le relais pour la gestion des contrats et des modalités de services pouvant évoluer au cours du temps.

Lors du début de l'alternance, l'affichage des données clients correspondait à une application nommée « dvdnet ». La *Figure 38 : Fiche prospect dvdnet* découpe la fiche prospect en plusieurs sections :

- La partie encadrée en rouge fait référence aux informations personnelles du prospect et de son interlocuteur : la personne étant en contact avec les résidences.
- L'onglet « Orientation » encadré en vert montre les contacts directs validés avec toute résidence. Dans le principe de « Multi-Orientation », il est possible de cumuler les contacts avec plusieurs résidences.
- Et enfin encadré en rouge, la partie « Action » listant les différents échanges faits avec le prospect.



Orientations		Actions	
Orienté le	Ortine / Sous origine	Réidence	Action
14/12/2021 06:27	Annuaire / Booking Senior	MARSEILLE - Sainte-Anne	Proposition de visite
			Action Commerciale
			Devis
			Admission
			Refus
15/02/2022	M. PRANGE Alexandre	Mme ANNE BOSSY OSTENDE	Téléphone
11/02/2022	M. PRANGE Alexandre	Mme ANNE BOSSY OSTENDE	Visite
07/10/2021	M. Service Portail	Mme ANNE BOSSY OSTENDE	Téléphone
07/10/2021	M. Service Portail	Mme ANNE BOSSY OSTENDE	Téléphone

Figure 38 : Fiche prospect dvdnet

2. Projet de refonte du CRM

Comme dit plus haut, le CRM a eu plusieurs projets. Le CRM 2.0 correspondait à l'un des premiers lors de mon arrivée sur la mission Domusvi. Ainsi, l'équipe Portail a été chargée des changements de technologies et d'une refonte visuelle de la fiche prospect. Le besoin était premièrement esthétique, la page devait être plus épurée et moins dense en informations avec des découpages plus marqués. Par ailleurs, le besoin était également technique : l'application dvdnet était construite en ASP.NET. C'était également la possibilité de faire évoluer l'application aux normes : un back-end et un front-end en .NET Framework et un complément front-end en React Js. Un autre point technique visait à supprimer tout utilisation de Bootstrap : un outil simplifiant le design de site web grâce à du code HTML et CSS.

Concernant l'aspect technique, un front-end en .NET Framework couplé au framework React Js peut se représenter avec la *Figure 39 : Architecture Front-end .NET Framework*. Nous nous intéresserons principalement aux encadrés en bleu, en rouge et en blanc sur l'image :

- Le premier cadre en bleu représente l'ensemble des configurations .NET de l'application avec la gestion des échanges avec les autres applications et avec le navigateur. A noter que le fichier « StartUp.Auth.cs » est le fichier gérant la sécurité des échanges dont les échanges MSAL vu dans le projet MSAL précédent.
- Les deux autres régions entourées en rouge et blanc montrent les fichiers gérant l'affichage. En rouge, nous retrouvons les fichiers « bundle » avec les principes de regroupement et de minification expliqués dans la partie **B. 4. Environnement de Travail**. Et la partie blanche fait référence aux fichiers .NET qui traduisent ce fichier « bundle » et créent les objets pour le navigateur internet.

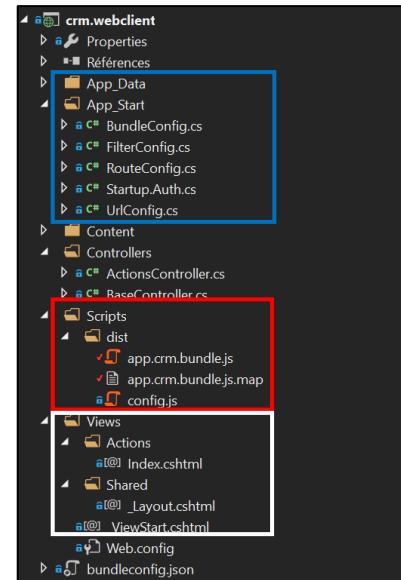


Figure 39 : Architecture Front-end .NET Framework

De plus, le projet visait à changer l'ensemble de l'affichage de la page de prospect. Nous pouvons retrouver certaines maquettes en annexe (*Annexe 3 : Chantier Ergonomique CRM*). Mais le principe du besoin était d'apporter un découpage des sections. Dans un premier, il est nécessaire de reprendre l'affichage des données personnelles, c'est le rôle de la zone entourée en bleu sur la *Figure 40 : Maquette de la refonte du CRM*. Ensuite, dans le coin supérieur droit apparaît deux boutons : pour « enregistrer » ou « annuler » les modifications. Enfin, nous pouvons noter l'apparition de champs d'édition (encadré en rouge) pour les données alors que sur l'ancienne version un bouton ouvrait une fenêtre d'édition des champs.

The image displays a 'PROSPECT' form interface. At the top, there's a header bar with the prospect's ID (3948213 - Mme Lefranc Jeannette), their email (M.Lefranc.Renaud@gmail.com), and a date (06/05/2014). On the right side, there are two buttons: 'ANNULER' (Cancel) in orange and 'ENREGISTRER' (Register) in green. The main form area contains several input fields: 'La recherche est pour lui-même' (checkbox), 'Civilité' (dropdown), 'Prénom' (text input), 'Nom' (text input), 'Département' (text input), 'Date de naissance' (date input), 'Age' (text input), 'Adresse' (text input), 'CP' (text input), 'Lieu de résidence' (dropdown), 'Pays' (dropdown set to France), 'Ville' (dropdown), 'Situation de famille' (dropdown), 'Nombre enfants' (text input), 'Téléphone perso' (text input), 'Email' (text input), and 'Téléphone autre' (text input). A large red box highlights the entire section from 'La recherche est pour lui-même' down to the 'Nombre enfants' field.

Figure 40 : Maquette de la refonte du CRM

Ces champs sont créés grâce à du code React Js visible sur la *Figure 41 : Ligne de code d'affichage de la fiche prospect*.

Chaque encadré en blanc correspond à l'un des champs visibles sur la maquette affichée plus haut et aux textes devant les champs : le premier encadré en blanc équivaut au prénom, le deuxième à l'adresse et le troisième au nom de famille.

Nous allons rapidement décrire l'un des champs : l'adresse entouré en bleu sur la figure 41. Le mot en rouge « input » correspond à la création d'un champ d'édition. « Value » fait référence au stockage de la donnée pour plus faciliter sa gestion et le terme « onChange » définit la fonction de modification du champ en lui-même.

```


<Label className="app-lib-left"
    tooltipContent={this.buildContent(historyProspect.firstname)}
    content={i18n.t("LBL_FIRSTNAME")} mandatory={true} >
  </Label>
  <div className="app-field-left">
    <InputWithMandatoryValidator className="form-control app-input-mandatory" ...
      disabled={isAdmins || rightEdit} |>
  </div>
  <Label className="app-lib-left"
    content={i18n.t("LBL_ADDRESS")} >
  </Label>
  <div className="app-field-left">
    <input className="form-control"
      value={this.props.editProspect.ADDRESS || ''}
      onChange={this.props.handleChangeTextInput}
      name="ADDRESS" maxLength="100"
      disabled={isAdmins} />
  </div>
</div>


<Label className="app-lib-right"
    tooltipContent={this.buildContent(historyProspect.lastname)}
    content={i18n.t("LBL_LASTNAME")} mandatory={true} >
  </Label>
  <div className="app-field-right">
    <InputWithMandatoryValidator className="form-control app-input-mandatory" ...
      disabled={isAdmins || rightEdit} />
  </div>
</div>


```

Figure 41 : Ligne de code d'affichage de la fiche prospect

L'ensemble des champs d'édition appartient à un « component » pour rappeler les notions vues dans la partie **B. 4. Environnement de Travail**. Mais il existe d'autres components qui permettent de gérer les interlocuteurs des clients ou la gestion des actions. Situés plus bas sur la même page que la fiche CRM, ces affichages ont subi quelques changements graphiques cependant le fonctionnement reste inchangé : depuis une pop-up (petite fenêtre) après un clic sur un bouton. Un exemple est visible sur la *Figure 42 : Bas de la page d'une fiche prospect* suivante.

Le déroulement de ses modifications a amené une certaine organisation. Le projet étant conséquent, les deux versions ont dû cohabiter afin de ne pas déranger la gestion actuelle des prospects. L'équipe a été dans l'obligation de jongler entre des corrections sur la version dvnet et le développement de la nouvelle version.

The screenshot shows a CRM interface for managing contacts. At the top, there's a header with tabs like 'Ajouter interlocuteur', 'Contact', 'Adresse', 'Principal', and 'Actions'. Below this, a contact card is displayed for 'Mme Titine TESTNUMVERT (Fille)'. The card includes fields for 'Nom', 'Prénom', 'Email', 'Téléphone', and 'Actions'. A large blue button labeled 'Ajouter interlocuteur' is at the top right. Below the contact card, there's a section titled 'ACTIONS' with a table showing tasks like 'Action commerciale', 'Devis', 'Admission', and 'Réfus'. At the bottom, there's a summary table with columns for 'Réalisé', 'Date', 'De', 'Action', 'Note', and 'Actions'.

Figure 42 : Bas de la page d'une fiche prospect

3. Règles de gestion

Bien sûr, l'apparition d'un nouveau design a nécessité de nouvelles règles de gestion. Ces règles de gestion apparaissent dès la création d'un prospect. L'une des premières fait référence à l'obligation de remplissage de certaines données pour valider l'enregistrement du prospect. Nous pouvons ainsi citer le nom, le prénom, le département, la présence d'un interlocuteur... mais il y a également des règles conditionnelles : le code postal doit correspondre au département s'il est renseigné. L'un des points notables est la case « la recherche est pour lui-même » que l'on peut voir sur la *Figure 40 : Maquette de la refonte du CRM*. Ce cas remplace la présence obligatoire d'un interlocuteur pour le client mais nécessite le remplissage de plus de champs, comme un moyen de communication (email, courrier, téléphone) ou la civilité.

D'autre part, la modification des graphismes a amené l'ensemble des pop-ups utilisées jusque-là. En effet, avec le changement de technologie, même si les textes ne changent qu'en partie, la création de ces derniers permet une évolution des processus.

Nous pouvons prendre l'exemple des doublons. Pour éviter la réplication de données, le Portail DomusVi possède une gestion des doublons. Mais ce système a rencontré un problème : appuyer sur la touche échap lors d'apparition de certains pop-ups provoquaient des actions anormales et non-voulues. Ce pourquoi il a fallu changer le processus et définir les actions possibles pour gérer tous les cas de figure.

Cela nous permet d'introduire ce processus avec la présence de types de profils : les profils Sièges et les profils Résidences. La principale différence entre ces deux profils est l'attribution des droits. Un profil Siège possède plus de droits et moins de restrictions c'est d'ailleurs visible sur la *Figure 43 : Schéma BPMN du processus de gestion des prospects doublons* car il peut créer un doublon là où un profil résidence ne peut pas.

Le schéma BPMN suivant prévoit deux lignes directrices : celle avec un utilisateur Résidence (bloc supérieur) et c'est d'un utilisateur Siège (bloc inférieur), tandis que le bloc du milieu représente le front-end de l'application.

Ce schéma traduit les différents cas en fonctions des actions des utilisateurs. Ce schéma fut utilisé pour la gestion de comportements anormaux et est arrivé après la refonte. Il a suivi le processus des TMA comme indiqué dans la partie [B. 3. Processus de gestion du besoin](#).

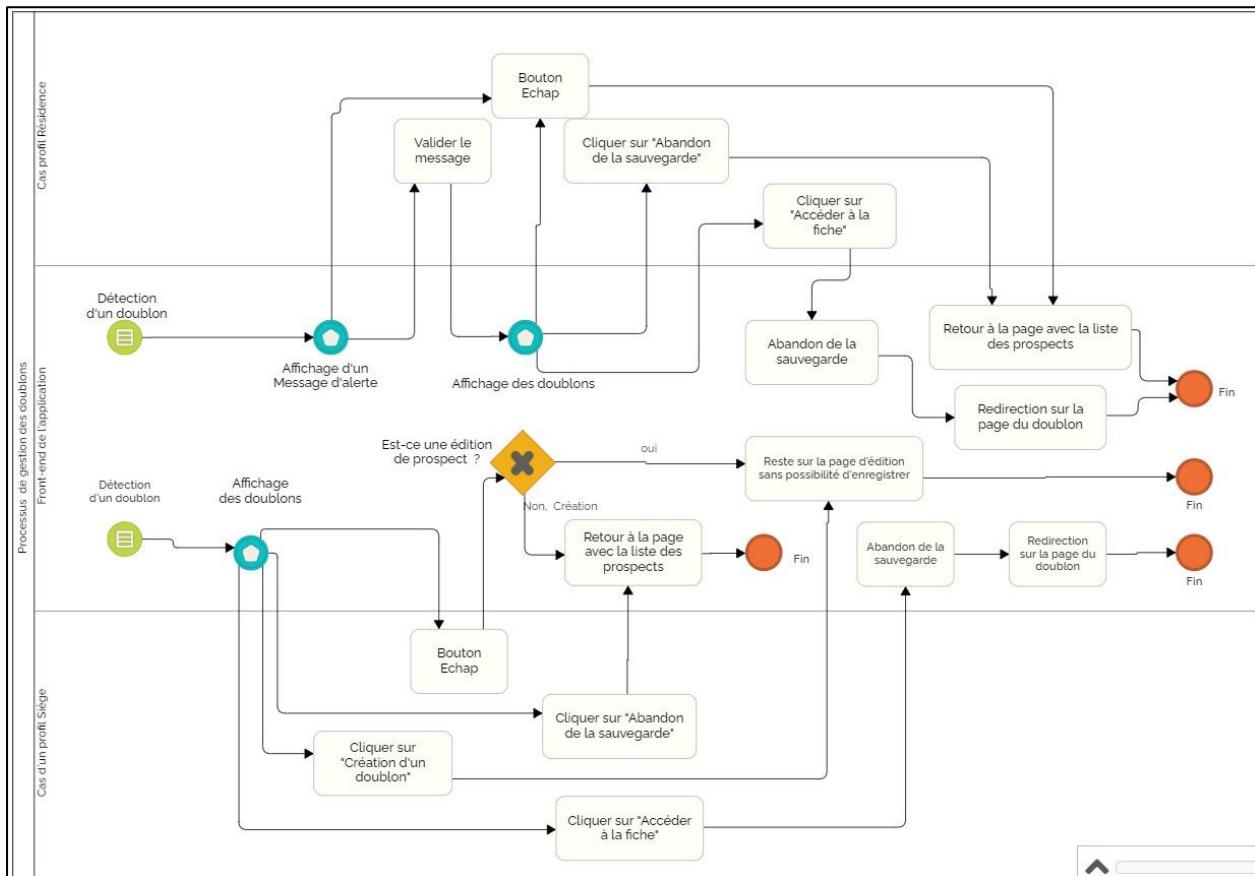


Figure 43 : Schéma BPMN du processus de gestion des prospects doublons

Avec la refonte, un besoin de créer une nouvelle gestion des statuts des prospects et des clients est arrivé. Le statut d'un client fait référence aux actions qu'il a fait auprès des résidences. Il peut être soit en contact avec des résidences, soit résident dans l'une d'elles soit sortie des résidences. Ces statuts sont décrits sur un schéma : la Figure 44 : Schéma des statuts des clients.

Sur cette image, nous pouvons souligner l'importance du contexte : « la résidence » définit la résidence propre à l'utilisateur et une « autre résidence ». Cette notion prend son sens lorsque l'on parle d'admission. « *Un client est-il admis sur ma résidence ou sur une autre ?* ». Ainsi sur ce diagramme, nous pouvons voir les statuts de « sortie » (encadré en rouge sur l'image) d'un client signifiant qu'il a été dans une résidence. Lorsqu'un client est admis dans la résidence de l'utilisateur, il est visible avec un statut « entré » tandis que les autres résidences le verront avec un statut « admis dans une résidence ». D'autre part, il y a la présence de statuts pour les contacts effectués : « visite prévue », « en cours » si le contact a été effectué mais sans visite prévue, « visite réalisée » et « refus ».

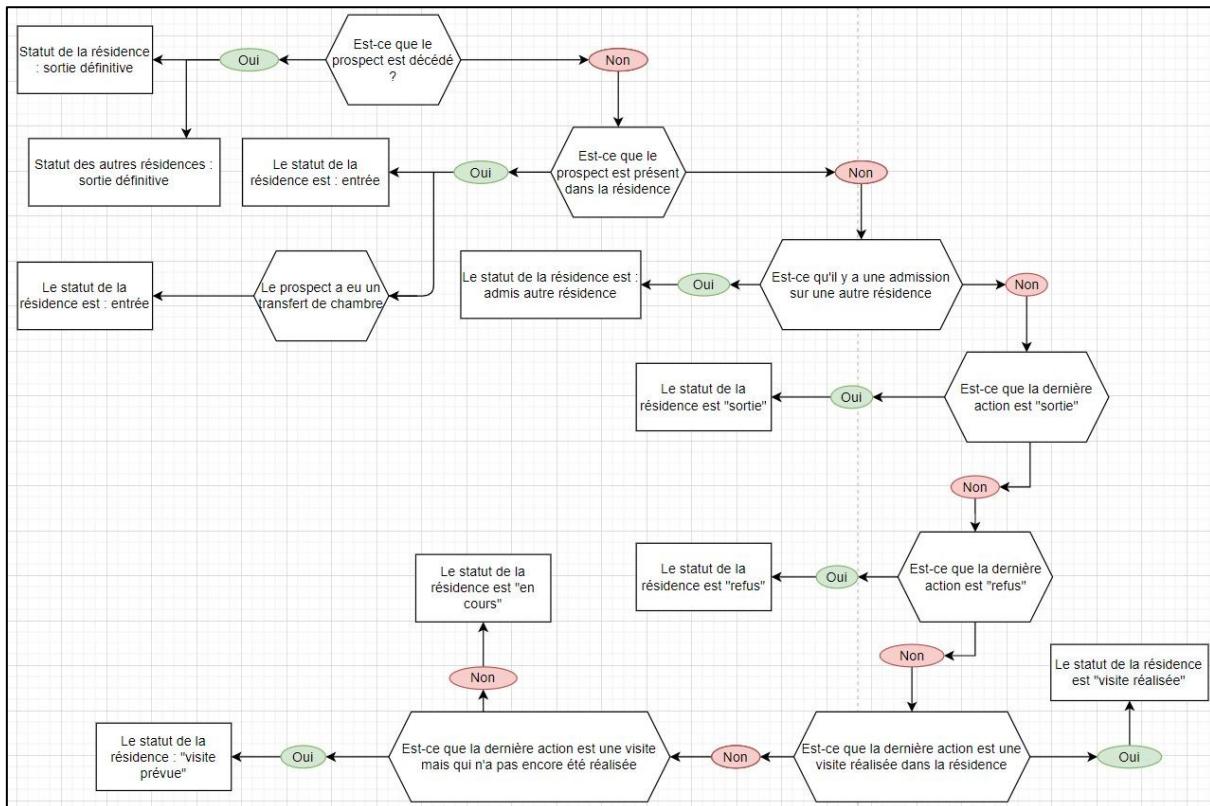


Figure 44 : Schéma des statuts des clients

E. MSPR OuiCar

Blocs de Compétences :

- A4 – Pilotage de l'informatique décisionnelle d'un système d'information (business intelligence & big data)

Activité : Définition d'une stratégie « Data » globale en alignement avec la stratégie « Business », Management des données structurées d'un SI

Compétences choisies : A4C1, A4C2, A4C3, A4C5, A4C6, A4C8, A4C9

1. Besoin du projet

Le projet qui suit a été réalisé dans le cadre du parcours EISI à l'EPSI, l'ensemble des informations réussis dans les explications sont tirées du travail produit lors de ce projet par groupe de 4 à 6. Il est possible de trouver des similitudes dans les dossiers de toutes les personnes ayant contribué à sa rédaction. De plus, l'entreprise est fictive et toute similitude avec une société est fortuite.

L'entreprise OuiCar, se spécialise dans le service de covoiturage et autopartage. OuiCar, en pleine croissance commerciale, souhaite concurrencer d'autres acteurs français et européens dans le domaine de la location et du covoiturage. C'est pourquoi, elle prévoit d'élargir son champ d'activité

avec la création d'une nouvelle application appelée OuiDispo. Cette application permettra aux utilisateurs de rechercher et de réserver des places de parking dans les grandes villes. L'application devra fournir toutes les informations nécessaires pour répondre à la demande croissante en places de parking dans les zones urbaines.

OuiCar a fait appel à notre société « Data Digital » afin de répondre à ses besoins. L'équipe qui était responsable de la conception de cette application se composait de quatre personnes.

Tout d'abord, la récupération du besoin en terme applicatif nous a permis de comprendre les enjeux et les besoins en termes de données.

Most Valuable Product

Dans un premier temps, nous pouvons présenter un diagramme Use Case (*Figure 45 : Diagramme des fonctionnalités de OuiDispo*) détaillant les principales fonctionnalités attendues de l'application mobile.

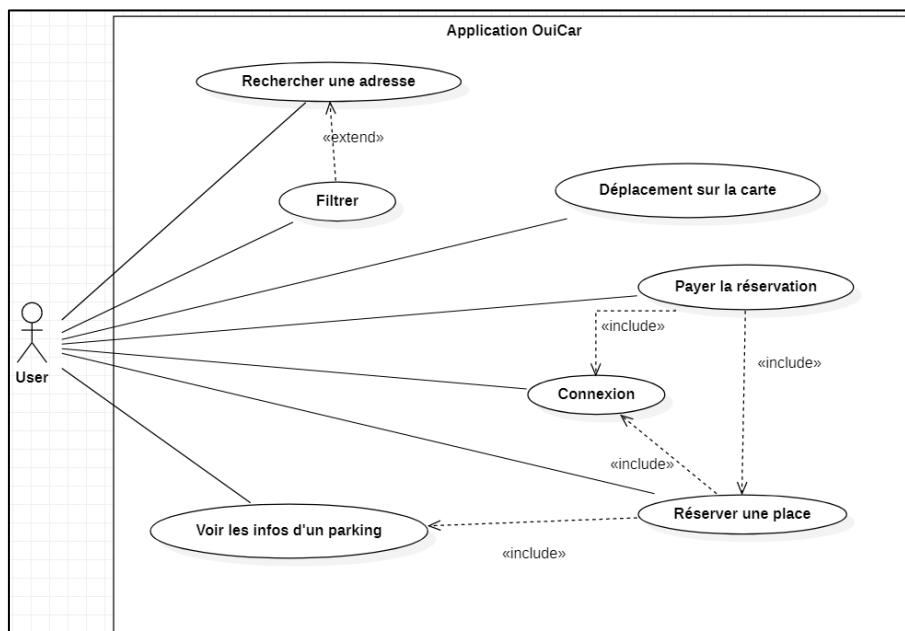


Figure 45 : Diagramme des fonctionnalités de OuiDispo

Une carte précise et fiable capable de localiser l'utilisateur ainsi que les parkings à proximité. Elle devra permettre la navigation sur la carte et devra afficher les détails du parking après un clic.

Une gestion de compte pour l'application est prévue pour la personnalisation de l'application et pour la gestion de la réservation.

Bien sûr, la réservation est la deuxième partie indispensable à l'application. Une connexion avec les systèmes de réservation des parkings est nécessaire pour mettre à jour le nombre de place disponible.

Par ailleurs, la gestion de compte pourrait se coupler à la partie facturation : l'utilisateur pouvant réserver à une date précise et pendant une certaine durée. Les clients doivent pouvoir payer et les sociétés propriétaires des parkings doivent pouvoir récupérer leurs contributions à la facturation de manière sécurisée. Un historique des réservations et des factures des utilisateurs est prévu.

depuis les gestions de compte. Pour cela, nous utiliserons un système externe que l'entreprise nous a fourni et utilise déjà pour son application principale de covoiturage et d'autopartage.

Ce diagramme correspond aux attentes minimum du projet pour sa validation d'autres améliorations pourront être discutées avec le client.

Afin de faciliter le développement multi-système d'exploitation mobile, nous avons choisi une technologie hybride pour la création de l'application. Le React Native semble également l'un des meilleurs outils pour combiner les échanges entre différents systèmes et la facilité d'affichage.

4. La structure des données

L'entreprise OuiCar nous fournit le jeu de données des stationnements en France pour baser notre système et structurer les données capitales à l'application. Mon principal apport dans ce projet concerne le modèle de données préparées pour l'application.

Après l'analyse du jeu de données, nous avons pu trier les données pour établir une liste des champs/attributs. Cette liste sert de structure aux modèles de données et décrit les noms donnés, leur type, leur provenance, leur objet de référence... c'est grâce à cette liste que nous présentons la *Figure 46 : Diagramme de classes du projet OuiCar*.

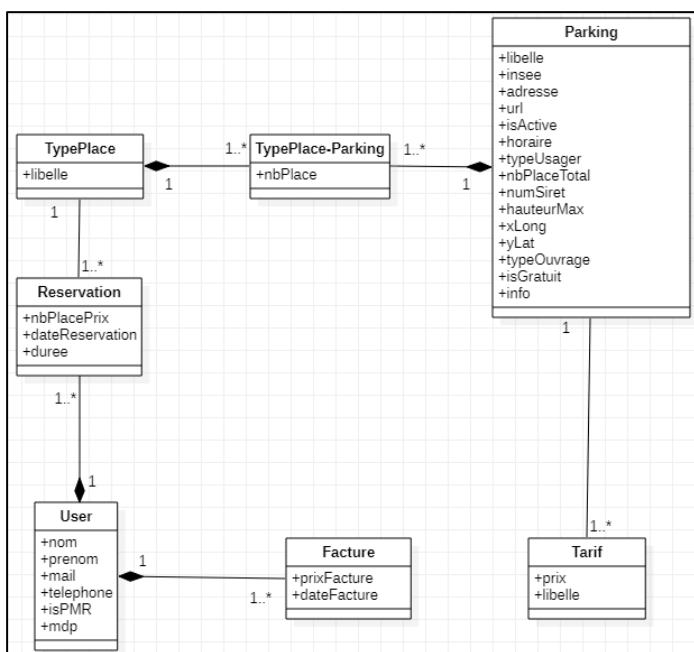


Figure 46 : Diagramme de classes du projet OuiCar

Cette figure nous a permis de créer la base de données nécessaire au fonctionnement de l'application.

Elle comprend différents objets : le parking avec toutes les coordonnées le définissant. Les réservations reliées avec le parking par deux tables pour gérer de potentielles évolutions avec différents types de places. Et l'utilisateur avec ses informations relié aux réservations et aux factures pour gérer l'historisation.

Pour la gestion des bases de données nous nous sommes dirigés vers PostgreSQL qui est un système de base de données relationnelles open source alimentée par une communauté. Il a l'avantage d'être performant et d'être mieux adapté aux gros volumes de données que d'autres système de base de données.

La première base de données est celle décrite par la figure plus haute, elle est utilisée par l'application afin de stockées les données indispensables aux fonctionnement de l'application. Cependant, cette dernière ne suffira pas, pour constituer la première base de données, nous avons besoin de données provenant de plusieurs sources. Nous avons besoin de récupérer le jeu de données des stationnements en France, au minimum chaque jour. De plus, plus différents parkings

devront nous envoyer certaines données pour pouvoir procéder aux différents échanges de données prévus dans OuiDispo. Cette deuxième BDD nous permet de pouvoir modifier la structure des données, de traiter les données pour de l'affichage sans impacter les utilisateurs de l'application.

5. Traitement des données

Comme dis plus haut, il y a plusieurs sources qui viennent alimenter notre système. Il est donc nécessaire de prévoir des flux. Tout d'abord, un programme vient mettre à jour chaque jour la table depuis le jeu de données des stationnements en France. Une gestion des doublons est également automatisée pour ne pas remplir inutilement la BDD.

Par ailleurs, les entreprises des parkings et notre système doivent pouvoir communiquer. Deux solutions sont possibles : la création d'une API de notre système définissant les informations nécessaires à la récupération des données pour les sociétés propriétaires des parkings. Cette solution nécessite aux entreprises de préparer leurs données pour que l'API puisse communiquer. L'autre solution vise à demander aux entreprises une API afin que notre système puisse récolter les informations nécessaires. Cette solution demande plus de travail pour les entreprises car en plus de l'API, une documentation sera nécessaire pour l'accès à leurs données. Afin de pouvoir gérer tous ses flux externes à l'application, nous avons mis en place un serveur Node. Js pour l'ensemble des services que nous avons sur le projet. Réputé optimisé et flexible, Node. Js est l'outil le plus adapté à nos besoins. Dans notre système, il permet de récupérer depuis les différentes sources les données et d'implémenter les bases de données.

Dans tous les cas, le système nécessite un « assainissement » des données, c'est-à-dire un processus de nettoyage, de correction et de mise à jour des données. Comme exemple du processus, concernant le jeu de données des stationnements en France, il faut le nettoyer en corriger les potentielles erreurs et changer le format des données puisqu'aucun des accents n'est lisible. Puis une certaine normalisation des données a été nécessaire pour faire correspondre les données aux bons types de données. L'un des autres procédés vise à enrichir les données avec d'autres sources. L'exemple de la gestion de la localisation des parkings. En effet, les données ne sont pas complètes et il est important de récupérer l'ensemble des numéros et noms des départements afin d'avoir des données plus lisibles.

L'ensemble de ces processus est facilité par le logiciel Power BI qui est un outil d'analyse et de traitements de données gérant également la visualisation de données. C'est grâce à cet outil que nous rédigeons des rapports et effectuons la surveillance. L'un des tableaux de bord créé pour le projet est visible sur la *Figure 47 : Exemple de dashboard du projet OuiCar*.

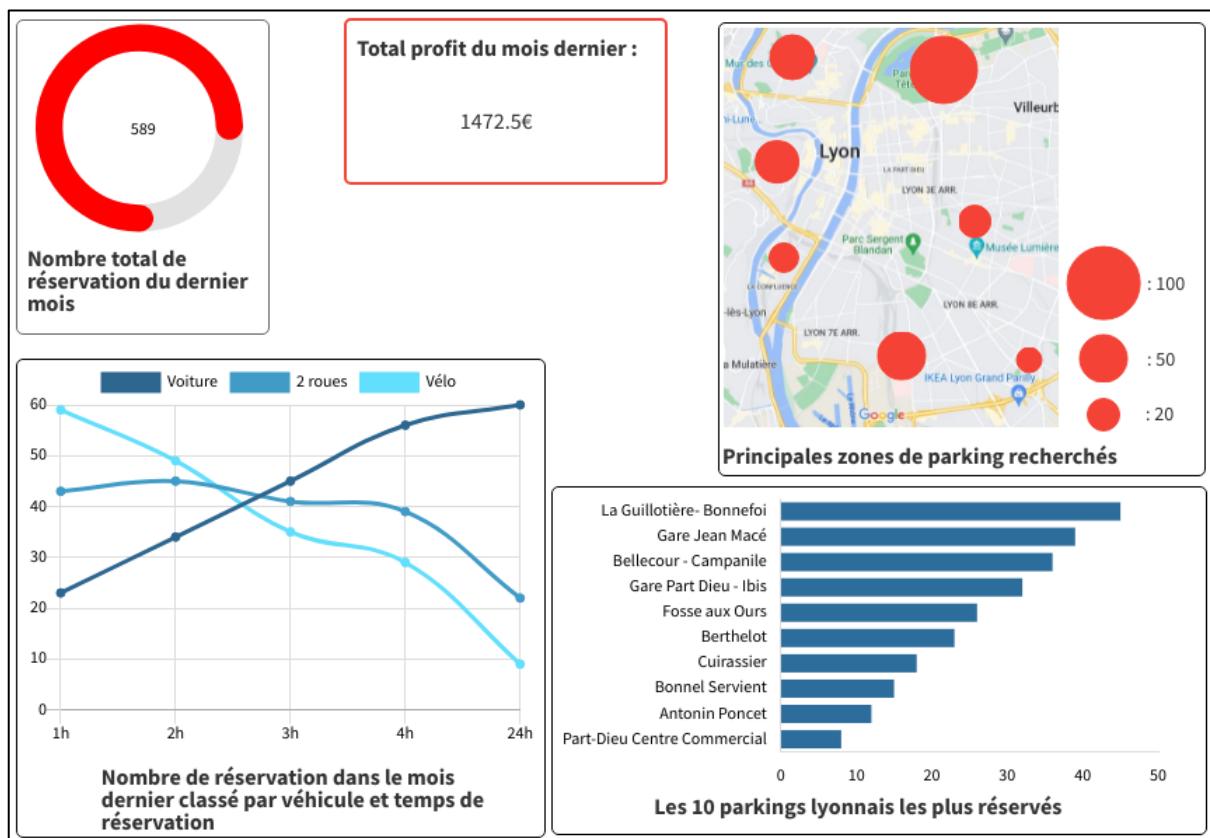


Figure 47 : Exemple de dashboard du projet OuiCar

Enfin en termes de sécurité, tout échange avec entre systèmes de données est crypté. Les données sensibles sont constamment cryptées avec l'algorithme sha256. Une protection sur les failles XSS est aussi présent afin d'éviter les attaques par injection de code depuis le navigateur. Nous comptons également sur une Politique de Sécurité du Système d'Information bannissant toute clé USB, régissant les accès et droits de manière stricte et adaptée au compte utilisé et définissant également la gestion des mots de passe : minimum de caractère, avec tout type de caractère, renouvellement courant...

Afin de garantir les données, il existe deux serveurs en réPLICATION permettant en cas de coupure de l'un, au deuxième de prendre le relais et empêcher tout arrêt du fonctionnement du système. D'autre part, ces serveurs internes à l'entreprise sont séparés de plus de 10 km afin de respecter la norme de sécurité préconisée.

Les normes RGPD sont également appliquées au système avec les droits en utilisation des données utilisateurs à leurs propriétaires, la possibilité de suppression des données...

V. Conclusion

Ausy se place en tant qu'ESN renommée avec beaucoup d'expériences dans de nombreux domaines tant en conseil qu'en technique. Elle bénéficie d'un contexte propice à son essor, le secteur de l'informatique est très dynamique et la sortie de la crise sanitaire a fait évoluer et augmenter le nombre de clients. Cependant, le marché est également concurrentiel et le contexte européen actuel est tendu et a des répercussions économiques sur tout la région ; région qui représente la majorité de son activité aujourd'hui. Enfin, Ausy entame une nouvelle phase pour répondre à des besoins organisationnels tout en ayant le support du groupe Randstad.

Le service informatique d'Ausy appelé « delivery » a une place clé dans la société, il est au centre de l'activité, devant répondre aux clients et donc satisfaire pour pouvoir subsister. De potentielles évolutions sur la gestion managériale pourrait optimiser encore l'efficacité des équipes. De plus, certaine évolution pourrait amener plus de simplicité quant à l'approche de nouveau projet et créer des échanges extra-projets apporterait une amélioration globale du service. Comme exemple, je citerai la création de discussions pour des échanges de bonnes pratiques. Mais le service est aujourd'hui performant notamment grâce à sa flexibilité et sa communication avec les clients pour répondre au mieux à leurs besoins.

En termes d'évolutions, cette alternance a été bénéfique tant sur le plan professionnel que personnel. Tout d'abord, c'est une première expérience professionnelle dans une ESN. La structure et le rapport concernant les développeurs sont différentes de ceux de mes précédentes expériences. Etant au centre de l'activité de l'entreprise, les relations avec les autres services et la communication fonctionnent autrement par rapport à ma dernière alternance. Par ailleurs, la taille de la société est aussi nouvelle pour moi, ce point est renforcé par la présence du groupe Randstad qui n'est pas discret et possède une implication grandissante. Pour parler plus de la mission que l'on m'a assignée, je note un développement de ma capacité de polyvalence et d'adaptation. En effet, c'est un nouveau domaine que j'aborde : la santé. Comme je l'ai aussi expliqué, le périmètre du Portail DomusVi est très vaste, tant fonctionnellement que techniquement. Je parle de gestion de client, de gestion de contrat pour des établissements séniors, de gestion de profils et de droit d'accès... Un autre point auquel j'ai dû m'adapter est la diversité des tâches que j'ai dû accomplir : tests fonctionnels, revues de code, développements front-end, back-end, gestions de BDD, déploiements sur divers environnements...

J'ai aussi pu étendre mon champ de compétences techniques : avec une nouvelle architecture applicative décrite dans ce dossier. Cette période a consolidé mes connaissances sur des langages informatiques : PL/SQL, .NET, React.Js mais j'ai également découvert le fonctionnement en entreprise de l'intégration et du déploiement continu avec Azure Dev Ops.

Cette alternance m'a personnellement apporté, notamment sur ma capacité à communiquer. L'équipe étant principalement à distance, cette communication a été cruciale pour la mission avec DomusVi. En tant que consultant Ausy pour un client, la relation n'est pas la même pour les entreprises où j'ai effectué des stages ou ma première alternance. Etant encore jeune, cette expérience développe aussi mon professionnalisme et d'une certaine manière mon implication.

Arrivant en fin d'étude, j'ai dû préparer ma complète intégration dans le monde du travail et donc la possibilité d'intégrer Ausy. Je peux ainsi décrire de manière plus précise mon projet professionnel. En premier lieu, je prévois une consolidation de mes compétences techniques : développement, architecture, déploiement et intégration d'applications afin de m'orienter sur une expertise en développement. A moyen terme, j'imagine me rapprocher du côté technico-fonctionnel : la partie

de conception architecturale des applications m'intéresse de plus en plus. Enfin, avec la formation EISI, un poste avec plus de responsabilités sur la gestion de projet pourrait répondre à mes attentes sur le long terme.

Table des figures

Figure 1 : Carte des Delivery Centers Ausy en France	5
Figure 2 : Organigramme Ausy	6
Figure 3 : Macro-processus de Ausy.....	7
Figure 4 : Organigramme de DomusVi	9
Figure 5 : SWOT de Ausy	10
Figure 6 : Schéma des principaux processus de Ausy	11
Figure 7 : Matrice des risques de la mission DomusVi	14
Figure 8 : Page d'accueil du site portail DomusVi	16
Figure 9 : Système d'information du Portail DomusVi	17
Figure 10 : Domaine CRM du portail DomusVi.....	17
Figure 11 : Domaine MDM du portail DomusVi	18
Figure 12: Schéma de la méthode Agile Scrum	20
Figure 13 : Salons de discussions présents sur le serveur discord DomusVi.....	23
Figure 14 : Backlog de sprint de l'équipe Portail.....	24
Figure 15 : Processus d'ajout d'une fonctionnalité	25
Figure 16 : Exemple de découpage des objets dans Azure DevOps.....	25
Figure 17 : Tableau des statuts des US.....	26
Figure 18 : Exemple de fiche de Mise En Production	27
Figure 19 : Tableau des statuts des TMA	28
Figure 20 : Dashboard du suivi des TMA.....	29
Figure 21 : Liste des environnement informatiques du Portail DomusVi	30
Figure 22 : Schéma d'une architecture Repository en lien avec un front-end.....	31
Figure 23 : ModelDvi du projet CRM (1)	32
Figure 24 : ModelDvi du projet CRM (2)	32
Figure 25 : Schéma du cycle de données en React Js.....	33
Figure 26 : Schéma du fonctionnement de React Js	34
Figure 27 : Schéma d'une chaîne d'intégration continue et de déploiement continu.....	35
Figure 28 : Diagramme de comparaison de ADAL et MSAL	37
Figure 29 : Schéma du fonctionnement d'ADAL sur le projet Portail	38
Figure 30 : Exemple d 'analyse du projet MISAL	39
Figure 31 : Schéma des cas de fonctionnement de MSAL.....	39
Figure 32 : Livrables attendus du projet MSAL	40
Figure 33 : Processus d'obtention et de rafraîchissement de MSAL.....	41
Figure 34 : Code de gestion de l'authentification	42
Figure 35 : Logiciel AAD Console	43
Figure 36 : Schéma du processus d'erreur de jeton de sécurité	43
Figure 37 : Domaine CRM du portail DomusVi.....	44
Figure 38 : Fiche prospect dvdnet	45
Figure 39 : Architecture Front-end .NET Framework	46
Figure 40 : Maquette de la refonte du CRM	46
Figure 41 : Ligne de code d'affichage de la fiche prospect	47
Figure 42 : Bas de la page d'une fiche prospect	47
Figure 43 : Schéma BPMN du processus de gestion des prospects doublons	49
Figure 44 : Schéma des statuts des clients.....	50
Figure 45 : Diagramme des fonctionnalités de OuiDispo.....	51
Figure 46 : Diagramme de classes du projet OuiCar	52
Figure 47 : Exemple de dashboard du projet OuiCar	54

Webographie

<https://www.lesechos.fr/idees-debats/leadership-management/trois-piliers-du-changement-agile-1246961>

<https://www.cegos.fr/solutions/formation-sur-mesure/sur-mesure-projet/infographie-projet-changement-agile>

<https://blog-gestion-de-projet.com/manifeste-agile-valeurs-et-principes/>

<http://agile-grenoble.org/agile-grenoble-2021-supports-et-videos/>

<https://all-it-network.com/methode-agile/>

https://fr.wikipedia.org/wiki/Oracle_Database pour ne citer qu'une page du site <https://fr.wikipedia.org/>

<https://stackoverflow.com/questions/53690172/why-and-where-all-do-we-need-bundle-js> pour ne citer qu'une page du site <https://stackoverflow.com/>

<https://aurelienbamde.com/2016/07/01/obligation-de-moyen-et-obligation-de-resultat/>

<https://www.microsoft.com/fr-fr/download/details.aspx?id=17851>

<https://learn.microsoft.com/fr-fr/azure/active-directory/develop/msal-migration>

<https://learn.microsoft.com/fr-fr/dotnet/framework/data/adonet/entity-data-model>

<https://learn.microsoft.com/fr-fr/skypeforbusiness/plan-your-deployment/modern-authentication/plan-adal>

<https://learn.microsoft.com/fr-fr/aspnet/mvc/overview/performance/bundling-and-minification>

pour ne citer que ces pages du site <https://learn.microsoft.com>

<https://grafikart.fr/tutoriels/module-bundler-979>

<https://medium.com/point-js/webpack-un-bundler-pour-les-gouverner-tous-1-3-c13cd8205f45>

Annexes

Annexe 1 : 2022-2023 Annexe4 AttestationNonPlagiat



epsi®
École d'Ingénierie
Informatique



pro alterna

ATTESTATION DE NON-PLAGIAT

(À inclure dans le dossier professionnel)

Je soussigné(e)

Nom : FURNON Prénom : CYRIL

Auteur du dossier professionnel pour la Certification Niveau 7 – Expert(e) en Informatique et Système d'Information - RNCP 35584

Titre du dossier professionnel :
FURNON_Cyril_EISI_2023.pdf

Déclare :

- Que ce dossier professionnel est un document original, fruit d'un travail personnel
- Avoir obtenu les autorisations nécessaires pour la reproduction d'images, d'extraits, de tableaux, figures ou graphiques
- Ne pas avoir contrefait, falsifié, copié tout ou partie de l'œuvre d'autrui afin de la faire passer pour mienne ;

Atteste :

- Que toutes les sources d'information utilisées pour ce travail de réflexion et de rédaction sont référencées de manière exhaustive et claire dans la bibliographie/webographie de mon dossier professionnel
- Être informé(e) et conscient(e) que le fait de ne pas citer une source ou de ne pas la citer clairement et complètement est constitutif de plagiat, et que le plagiat est considéré comme une faute grave et sanctionnée par la Loi.

Date et Signature :

Fait le : 29/07/2023

À :

LYON

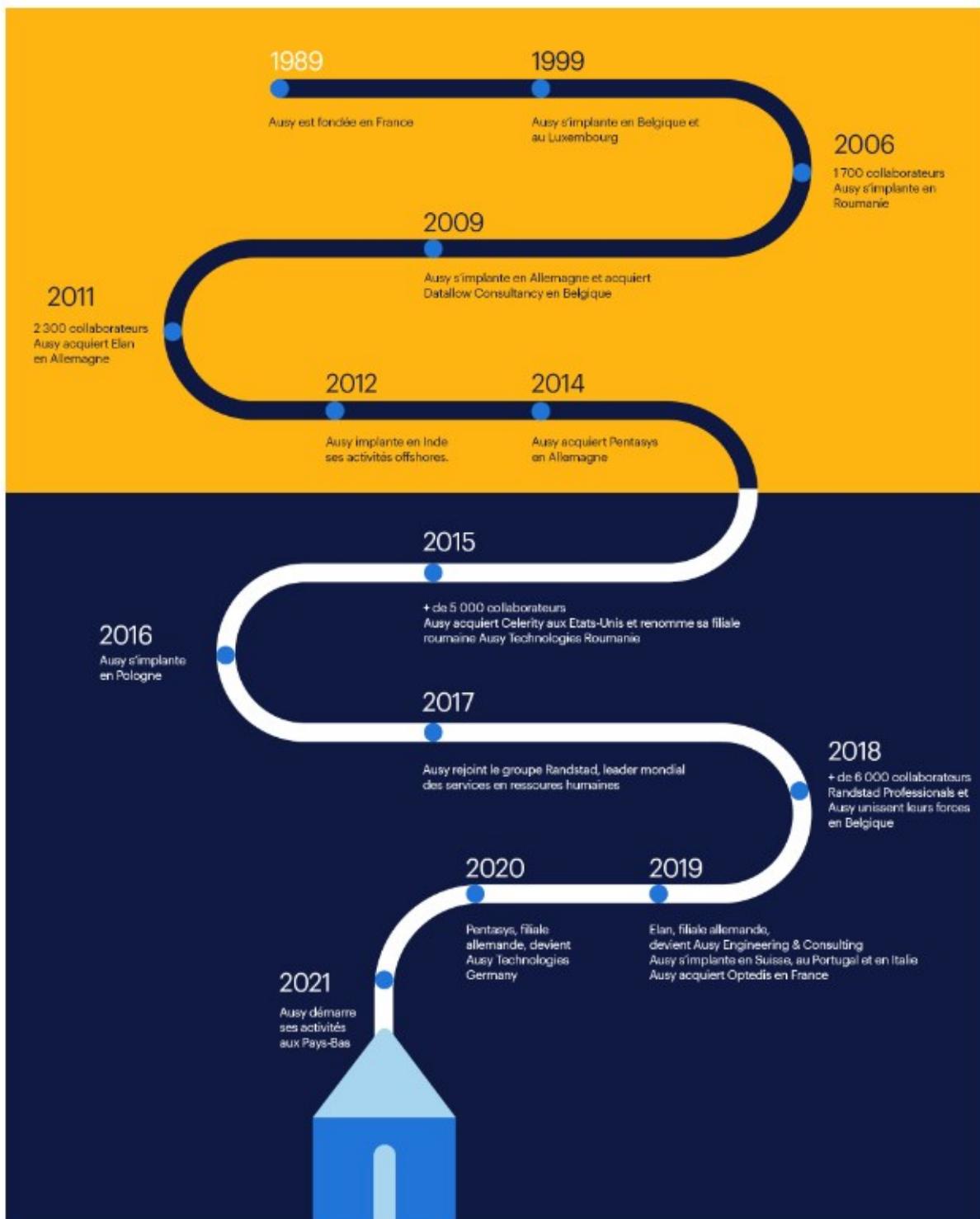


Signature :

Décret n°2016-1022 du 11 juillet 2016 relatif à la certification professionnelle et à la formation continue.

Annexe 2 : Historique Ausy

Source : <https://www.ausy.fr/fr/a-propos-d-ausy/notre-histoire/>



Annexe 3 : Chantier Ergonomique CRM

La création

- **Le bandeau :**
 - o Revoir le cartouche :
 - Déplacer le cartouche à gauche
 - Le redéposer et aligner les informations
 - Distinguer les informations du Prospects (sur 2 lignes) et celles de l'Interlocuteur (sur 3 lignes) dans le cartouche pour que ce soit bien compréhensible
 - Mauvais comportement du bandeau en mode responsive
 - o Le Workflow :
 - A intégrer dans le cartouche

- Passer uniquement les icônes sur une ligne qui sera intégrée en bas à gauche du cartouche
- Donner une signalétique sur les couleurs : par ex une couleur « grise » pour inactive, une couleur pour « actif » dans la résidence, une couleur pour « actif ailleurs »... Affichage d'un tooltip quand on passe dessus avec information de la résidence (factoriser, remplacement du « I » qui semble inutile. A voir pour le trait.

- o Mode Création : afficher le bandeau au fur et à mesure de la saisie

- **Les catégories :** séparateurs de paragraphe (ex PROSPECTS / INTERLOCUTEURS... :

- o Uniformiser les présentations des catégories : revoir la forme et couleur par ex

- **Prospect :** revoir la disposition des informations :
 - o Regrouper les informations obligatoires sur la gauche : civilité, nom, prénom, département et les 2 téléphones et l'email (cas où la coche « Lui-même » est cochée)
 - o Tabulation : de haut en bas : toute la colonne gauche puis la droite
 - o Modifier le libellé « lui-même » par « la recherche est pour lui-même » (écran de création, dans l'écran de l'édition il n'y a pas cette coche)
- **Les listes déroulantes :**
 - o Pb de couleur pour les champs obligatoires :

- Pb d'encadrés : uniformiser avec les autres données -> passer à encadrés pour les listes déroulantes
- Pb de dimension pour le département :
- Pb de couleur lorsque l'information est obligatoire, ne respecte pas la rose

Nom * [Text input]

Département * [Text input] Rechercher un département...

Lieu de résidence [Text input]

Pays France

- Date de naissance : revoir le format de la date à l'identique de celui de l'admission
- INTERLOCUTEURS :

Téléphone perso [Text input]

Email test@test.fr

INTERLOCUTEUR

Contact	Adresse	Principal	ACTIONS
Mme Christine DELAPORTE (Fille)	RUE DU TILLEULS cdelaporte@orange.fr	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

1 - 1 de 1 éléments

- Pb d'alignement de la grille : à aligner sur les libellés des champs du prospect,
- Pb pagination de la grille des interlocuteurs
- Ne pas afficher le téléphone et le mail s'il n'y a pas d'information.
- Reporter les mêmes principes sur la pop-up « Ajouter » interlocuteur : les listes, les couleurs, l'ordre avec les champs obligatoires à gauche et la tabulation sur les champs de gauche

Création interlocuteur

Personne à contacter

Civilité * [Text input]

Prénom * [Text input]

Nom * [Text input]

Pays [Text input]

CP [Text input]

Adresse [Text input]

Ville [Text input]

Lien de parenté [Text input]

Email * [Text input]

Tél. autre * [Text input]

Annuler Enregistrer

- Revoir les marges : recentrer car à droite c'est pas beau !
- Revoir le titre des POP-UP : gras, framboise écrasée. A voir si en majuscule ou pas.

- **ORIGINE :**
 - o Idem les autres catégories :
 - Titre à revoir,
 - Données obligatoires en rose
 - Liste déroulantes
- **ACTION en « création » :**
 - o Manque la catégorie « Action »
 - o Mêmes remarques :
 - Titre à revoir,
 - Données obligatoires en rose
 - Liste déroulantes
 - Dates en rose + reprendre le format de l'admission
 - Modifier le libellé « Action » en « Action de relance »

Il faudrait, quand on a fini de saisir un prospect, après l'enregistrement, aller directement sur le motif sans passer par l'édition.

→ **Piloter la section sur laquelle on veut arriver**

- **MOTIF :**
 - o Appliquer les principes d'affichage : les sections, framboise écrasés, encadrés....
 - o Ajouter un bouton « SUIVANT » pour passer, lorsqu'on est en mode édition, au « bilan des ressources ».
- **Fonctionnement des boutons ANNULER / ENREGISTRER :**
 - o « griser les boutons » tant qu'il n'y a pas de saisie
 - o La saisie d'une donnée provoque l'activation des boutons.
 - o Ajouter un bouton « SUIVANT », activable uniquement dans certains cas (Motif)

Le mode édition

Appliquer les principes du mode création sur les titres, les affichages, les encadrés,...

- **LE PROSPECT :**
 - o Il manque le titre « Détail du prospect », à l'identique des sections MOTIF, ORIENTATION...
- **Les actions commerciales :**
 - o Ajouter un menu « Actions commerciales » qui permet d'arriver directement sur la saisie des actions
- **MOTIF :**
 - o Appliquer les principes d'affichage : les sections, framboise écrasés, encadrés....
 - o Ajouter un bouton « SUIVANT » pour passer, lorsqu'on est en mode édition, au « bilan des ressources ».
- **BILAN DE COMPETENCE :**
 - o Appliquer les principes d'affichage : les sections, framboise écrasés, encadrés....
 - o Ajouter un bouton « SUIVANT » pour passer, lorsqu'on est en mode édition, au « bilan des ressources ».
- **Fonctionnement des boutons ANNULER / ENREGISTRER :**
 - o « griser les boutons » tant qu'il n'y a pas de saisie
 - o La saisie d'une donnée provoque l'activation des boutons.
 - o Ajouter un bouton « SUIVANT », activable uniquement dans certains cas (Motif)
- **ORIENTATIONS :**
 - o Appliquer les principes du mode création sur les titres, les affichages, les encadrés,...

La pop-up action commerciale

Nouvelle action

De Laureline JAFFRELOT (DIRECTEUR(TRIC) A

1. Action en cours

Action	Date action	jour/mois/année	
Mémo			

2. Prochaine action de relance

Action	Date action	jour/mois/année	09:00
Mémo			

- **Fonctionnement des dates**

Pour les dates d'action en cours et prochaine action de relance, il faudrait faire le même fonctionnement que les dates de l'admission, elles ne sont pas ergonomique lors de la saisie manuelle