

Université de Lille IUT de Lille
Département informatique

Découvrir les facettes du métier d'Intégrateur SOA webMethods

Rapport de stage

Soutenu le 24 juin
par
Camille Fourmaintraux

Encadrant entreprise : Alexandre Fretigny

Encadrant universitaire : Jean Carle

Remerciements

Je souhaite remercier mon tuteur de stage Monsieur Alexandre Fretigny, Chef de projet chez Emotheop, de m'avoir fait confiance tout au long de cet apprentissage.

Je remercie Monsieur Jean Carle, mon professeur de l'IUT de Lille et mon tuteur universitaire, qui m'a apporté de précieux conseils pour la rédaction de ce rapport de stage.

Je tiens à remercier aussi Monsieur Geoffrey Vanaerdewegh, Intégrateur SOA webMethods, qui m'a apporté son expertise lors de ma formation et m'a beaucoup appris.

Merci à Monsieur Thibaut Volpoet pour sa bienveillance.

Merci à Monsieur Alexandre Nuytens, Monsieur Maxime Bimont et Monsieur Flavien Carriere pour leur aide quand j'étais en difficulté.

Merci à Léonorine Tibère et Xavier Moyon, mes collègues stagiaires, pour leurs soutien et leur bonne humeur durant ce stage.

Un grand merci à l'ensemble de l'équipe d'Emotheop, pour leur accueil chaleureux, leur disponibilité et leur soutien.

Enfin, je remercie également ma famille et mes amis pour leur soutien pour ma recherche de stage et pour m'avoir relu pendant la rédaction de ce rapport de stage.

Table des matières

Remerciements	1
Résumés	3
Résumé	3
Abstract	3
Introduction	4
1 Présentation de l'entreprise et du Contexte de Stage	6
1.1 Présentation de l'entreprise	6
1.2 Contexte du stage	7
2 Missions et Méthodes Utilisées	10
2.1 Description des Missions	10
2.2 Méthodes et Outils Utilisés	12
2.2.1 AMAT - Développement de l'API	12
2.2.2 AMAT - Évolution des services de purges	14
2.2.3 Rétro documentation	15
2.2.4 Base de connaissances - Gestion des noms et des prénoms .	16
2.2.5 Base de connaissances - Gestion des erreurs	16
3 Analyse des Résultats et Compétences Acquis	17
3.1 Résultats Obtenus	17
3.1.1 Évolution d'AMAT	17
3.1.2 Services de Purges	17
3.1.3 Rétro-documentation	17
3.1.4 Base de Connaissance	18
3.2 Résultats Obtenus	18
3.2.1 Compétences Techniques	18
3.2.2 Compétences en Gestion de Projet	18
3.2.3 Compétences Relationnelles	19
Conclusion	20
4 Bibliographie	22
5 Glossaire	23
A Annexe	27

Résumés

Résumé

J'ai réalisé mon stage chez E-Mothep, une entreprise spécialisée dans les solutions logicielles pour la gestion des systèmes d'information. Sous la supervision d'Alexandre Fretigny, j'ai travaillé sur l'évolution d'AMAT, en intégrant de nouvelles fonctionnalités et en développant une API pour automatiser la mise à jour des référentiels, améliorant ainsi les processus internes. J'ai également rédigé de la rétro-documentation et contribué à une base de connaissances pour faciliter le travail futur. Ce stage m'a permis de développer des compétences en gestion de bases de données, développement d'API, documentation, et gestion de projet avec des outils comme DBeaver, Software AG Designer, Postman, et GitHub. Cette expérience a renforcé mes compétences techniques et confirmé mon projet professionnel dans le développement logiciel.

Abstract

I completed my internship at E-Mothep, a company specializing in software solutions for information systems management. Under the supervision of Alexandre Fretigny, I worked on the evolution of AMAT, integrating new features and developing an API to automate the updating of repositories, thereby improving internal processes. I also wrote retro-documentation and contributed to a knowledge base to facilitate future work. This internship allowed me to develop skills in database management, API development, documentation, and project management using tools like DBeaver, Software AG Designer, Postman, and GitHub. This experience strengthened my technical skills and confirmed my professional goal in software development.

Introduction

J'ai eu l'occasion d'effectuer un stage de 10 semaines du 15 avril au 21 juin au sein de l'entreprise E-Mothep, situé en métropole Lilloise, dans le cadre de mes études en seconde année de BUT informatique, anciennement DUT, à l'IUT de Lille.

Tous les sigles et acronymes dans ce rapport sont expliqués dans le Glossaire à la page 23 .

E-Mothep est une société à taille humaine fondée en 2004 de la volonté d'accompagner et de conseiller les projets informatiques de leurs clients. L'entreprise est spécialisée dans l'Architecture et l'Urbanisation, l'Intégration du système d'information, la Gestion de la relation client, le Pilotage de projet, l'Assistance à la Maîtrise d'Ouvrage et la formation.

E-Mothep est composée de plusieurs entités :

- E-Mothep Center : possède et gère les locaux de l'entreprise.
- E-Mothep Group : réunit les autres entités et s'occupent de l'administration.
- E-Mothep Consultants : intègre les solutions Software AG et conseille ses clients. Ses deux grands domaines d'intervention sont le conseil en architecture orienté services et l'intégration des solutions de Software AG.
- E-Mothep School : qui grâce à toutes les expériences accumulées dans la première, conçoit des formations en Conseil et Intégration sur les technologies webMethods, qui sont découpées en sessions de cours théoriques, en démonstrations et en travaux pratiques sur des cas métier. Leurs formateurs, experts dans le domaine du conseil et de l'intégration de solutions d'échanges, bénéficient d'une grande expérience du terrain, et proposent des formations sur mesure, selon le contexte métier ou les attentes du client.

Software AG est un éditeur de logiciel Allemand à portée multinational. J'explique plus en détail SoftwareAG et leur outil Software AG Designer dans l'annexe A.1 Exploration de Software AG Designer à la page 28.

E-Mothep est reconnue pour ses valeurs d'excellence, de solidarité, d'innovation, d'humanité et de proximité. Leur succès repose sur la collaboration de leurs experts, l'innovation de leurs manières de faire face aux évolutions humaines et technologiques et la progression constante de leurs collaborateurs. Ce stage a pour principal but de me faire découvrir la vie en entreprise, et de me permettre d'acquérir une expérience professionnelle pratique en lien avec mes études en découvrant de nouveaux outils informatique.

Il s'est déroulé sous la supervision de Monsieur Alexandre Fretigny, Chef de projet et avec la formation de Monsieur Geoffrey Vanaerdewegh, Intégrateur SOA webMethods. WebMethods est une suite de logiciels utilisés pour l'intégration et la gestion de processus métier, développé par Software AG.

Au cours de ce stage, j'ai été affecté à différentes missions très variées, chacune m'apportant et m'apprenant un nouveau lot de connaissances. Mon rôle en tant que stagiaire Intégrateur SOA webMethods consistait principalement à développer des flux et des webServices REST/SOAP, à analyser et corriger des flux ainsi qu'à

rédiger de la documentation. Plus spécifiquement, j'ai principalement été chargé de faire évoluer la solution AMAT d'E-Mothep. Les clients d'E-Mothep utilise la plateforme d'échanges webMethods depuis plusieurs années afin d'automatiser et de superviser ses chaînes de fabrication. Ils utilisent également la plateforme pour des échanges entre applications de ressources humaines. Avec de nombreux flux critiques, les clients disposent de la solution d'E-Mothep appelée AMAT qui permet de suivre l'état de leurs plateformes de production. AMAT permet de constituer des rapports d'analyses sur un environnement webMethods. Après analyses, un rapport est envoyé par mail aux personnes concernées. Si aucune déviation n'est trouvée, l'état « OK » est retourné. Dans le cas contraire, le détail est donné par l'identifiant unique de l'asset, permettant d'identifier rapidement le problème. Un asset est un élément surveillé par AMAT, chacun de ses attributs correspond à une entrée dans la base de données. Lorsqu'un client souhaite mettre à jour son référentiel, il faut qu'il envoie un mail aux collaborateurs d'E-Mothep pour le mettre à jour, sinon les rapports vont envoyer des détections d'erreurs qui n'ont pas lieu d'être. Évidemment, ça n'est pas pratique, ça ralentit grandement la fluidité des communications, ça peut créer des confusions et ça augmente le risque d'erreur. Un référentiel est un ensemble auquel doivent être référencé les assets AMAT afin de s'assurer qu'ils sont corrects. En les comparant au référentiel, cela permet d'être sûr qu'ils n'ont pas été modifiés d'une manière inattendue. L'évolution d'AMAT est donc cruciale, il a été décidé de mettre à disposition du client une Interface de programmation d'applications (API), permettant à plusieurs programmes ou composants informatiques de communiquer entre eux. Cette API devrait offrir la possibilité de mettre à jour automatiquement le référentiel E-Mothep et donc de permettre au client d'être autonome sur la mise à jour de ces référentiels. Mes autres missions secondaires ont été de faire de la rétro documentation sur les projets internes car ils manquaient de documentations techniques mais possédaient de la documentation utilisateur. Ensuite, j'ai été affecté sur le projet interne de base de connaissances. Le principe de ce projet consiste à rassembler à un même endroit toutes les solutions que les collaborateurs trouvent lorsqu'ils sont confrontés à un problème afin d'épargner aux futurs collaborateurs de devoir à nouveau chercher cette solution et ainsi de gagner du temps sur les futurs développements.

Dans un premier temps, je présenterai plus en détail l'entreprise et le contexte dans lequel s'est déroulé mon stage. Ensuite, je décrirai les missions qui m'ont été confiées et comment je les ai accomplies. Enfin, j'analyserai les résultats obtenus, les outils que j'ai appris et les compétences que j'ai acquises durant ce stage.

Présentation de l'entreprise et du Contexte de Stage

1.1 Présentation de l'entreprise

Fondée en 2004 par Monsieur Fabrice Vanaerdewegh, E-Mothep est une société par action simplifiée (SAS) innovante spécialisée dans le développement de solutions logicielles pour la gestion des systèmes d'information.

A l'origine, le siège social de l'entreprise se situait à Roubaix, mais afin de se rapprocher de sa clientèle, l'entreprise a déménagé en tout trois fois. Celle-ci est maintenant située depuis 2014 au parc scientifique de La haute Borne à Villeneuve d'Ascq, qui bénéficie d'une situation géographique privilégiée. En effet, celui-ci se situe à proximité du campus de l'Université de Lille, et accueille divers laboratoires de recherche et entreprises innovantes. Il est situé à moins de 350 kilomètres de cinq capitales européennes et au cœur d'un des complexes scientifiques majeurs du triangle Paris-Londres-Bruxelles. Ces locaux étant plus spacieux permettent d'embaucher un nombre plus important de salariés et participent directement au développement économique d'E-Mothep. La *figure n°1.1* à la page 6 montre l'évolution croissante de 2 personnes en 2005 à récemment en 2018 plus de 30 personnes en CDI et presque 40 avec les stagiaires. Cette figure montre également une accélération de la croissance faisant suite à l'achat en 2014 des nouveaux locaux. Aujourd'hui, E-Mothep compte une quarantaine d'employés et continue de recruter activement. De plus, E-Mothep possède une autre agence, située à Casablanca, au Maroc.



FIGURE 1.1 – Évolution du personnel d'E-Mothep de 2005 à 2018

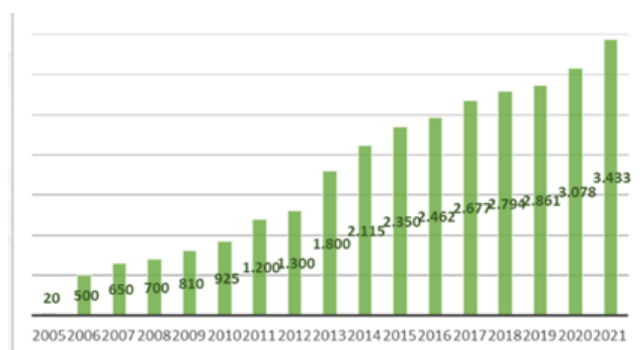


FIGURE 1.2 – Évolution du chiffre d'affaires en milliers d'euros

Comme l'indique la *figure 1.2* à la page 7, le chiffre d'affaires annuel d'E-Mothep s'élevait à moins d'un million en 2010 et à plus de deux millions en 2016. Le chiffre d'affaires augmente progressivement depuis la création de l'entreprise et plus récemment en 2021, il s'élevait à plus de 3.433.000 €.

Emothep connaît donc une croissance significative et régulière depuis sa création, tant sur le plan démographique, géographique et économique.

1.2 Contexte du stage

E-Mothep est composé de quatre entités, dont leurs rapports sont montrés dans la *figure 1.3* à la page 7 : E-Mothep Center : au sommet, E-Mothep Group : qui gère les deux autres, E-Mothep Consultants : l'activité principale, E-Mothep School : l'activité secondaire. E-Mothep Center est dirigé par Monsieur Fabrice Vanaerdewegh, le fondateur E-Mothep GROUP, dirigé par Monsieur Guillaume Deparis, englobe les deux autres entités. E-Mothep consultants est dirigé par Madame Élodie Boiteaux E-Mothep School est dirigé par Monsieur Thibaut Volpoet.



FIGURE 1.3 – Organigramme des entités d'E-Mothep

La *figure 1.4* à la page 8 nous explique l'organisation d'un département. Une entité est composée de plusieurs département, un encadrant fait référence au dirigeant d'une entité. Les encadrants dirigent les coordinateurs, qui eux-mêmes dirigent les collaborateurs. La structure d'E-Mothep est donc assez simple et facile à comprendre.



FIGURE 1.4 – Organigramme de l'organisation d'un département

Un département est composée d'au moins un collaborateur. Le poste de coordinateur est ouvert seulement à partir de 5 collaborateurs présents dans l'équipe. Un département équivaut à une équipe.

Une équipe peut travailler sur plusieurs projets et un projet peut concerner plusieurs équipes. Chaque projet nécessite ces rôles : chef de projet et un tech lead. Un chef de projet va être celui qui fait état de la progression aux dirigeants et va principalement être celui qui discute avec le client. Le tech-lead est celui qui choisit les technologies qui seront utilisés pour le projet et de quels manières elles seront utilisées.

Mon stage s'inscrit dans le cadre de mon cursus en BUT Informatique à l'IUT de Lille. Il se déroule du 15 avril au 21 juin au sein de l'équipe Nefertiti, sous la supervision de Monsieur Alexandre Fretigny, Chef de projet et Coordinateur.

Après plus de 15 ans d'existence et de clients satisfaits, E-Mothep GROUP créa en 2019 une nouvelle entité : E-Mothep School. Grâce à leurs années d'expériences, E-Mothep School conçoit des formations sur de nombreux sujets techniques mais la force d'Emothep School, c'est leurs formations sur mesures qui s'adaptent aux besoins selon le contexte métier ou les attentes des clients. Ces formations sont découpées en sessions de cours théoriques, en démonstrations et en travaux pratiques sur des cas métier. Grâce à ces TP qui mettent en scène des exercices représentant des situations réelles, l'apprentissage n'en est que plus efficace car moins abstrait.

E-Mothep Consultants, est l'entité principale de l'entreprise. Elle base son activité sur deux grands domaines d'intervention : le conseil et l'intégration. Elle accompagne ses clients, pour la plupart grands comptes, pour leur apporter Expertise et Conseil autour de la SOA, de l'urbanisation du SI ainsi que de leurs

évolutions au sens large. Aujourd'hui, une majorité de sociétés désirent sortir ou limiter les plans applicatifs, et entament ainsi une démarche « Best of Breed » : cela consiste à éviter de se fier à un seul éditeur pour choisir un seul logiciel d'entreprise multi-tâche, mais ils cherchent plutôt à trouver le meilleur logiciel pour chaque domaine d'application individuel et ainsi construire une architecture technologique sur mesure qui conviendrait parfaitement à leurs besoin. D'autres désirent mettre en place de nouvelles applications ou progiciels qui nécessiteront d'échanger de l'information avec les applications existantes. Dans ces conditions, comment faire pour rendre un SI agile, évolutif, et garantir les échanges d'information entre domaines métier ?

C'est la problématique à laquelle répond E-Mothep, en apportant aux entreprises tous les conseils et le savoir nécessaires pour parvenir à transformer leur SI.

E-Mothep apporte tous les conseils nécessaires à ses clients pour parvenir à transformer un SI, garantissant les échanges d'informations entre domaines métiers. L'entreprise apporte un accompagnement sur mesure à chacun de ses clients, les services proposés vont du cadrage du besoin, au conseil technique en passant par l'installation de la plateforme, à la réalisation dans le respect de l'architecture orientée service (SOA). L'architecture orientée service consiste à développer en utilisant des composants logiciels appelés services pour créer des applications métiers. Un service est un programme dans le cadre de Software AG Designer. C'est une unité fonctionnelle réutilisable exécutant une tâche précise allant de simplement transformer des données ou effectuer de simples opérations à l'orchestration de flux complexe ou l'orchestration d'applications multiples. Un flux correspond à un écoulement de données entre les services, c'est le cheminement des données et des opérations. Software AG Designer est utilisé pour concevoir et développer des services et des flows qui s'exécutent sur l' Enterprise Service Bus. Les développeurs peuvent créer, tester et déployer des services d'intégration, des services web, et des orchestrations de processus métiers dans un environnement graphique intuitif. L'Enterprise Service Bus (ESB) est une architecture de middleware¹ qui permet l'intégration des différentes applications et services au sein d'une entreprise. Il facilite la communication entre divers systèmes en utilisant un bus centralisé qui transmet les données et les messages de manière fiable et flexible.

1. Un logiciel qui se trouve entre un système d'exploitation et les applications qui s'exécutent sur celui-ci.

Missions et Méthodes Utilisées

2.1 Description des Missions

Durant ce stage, ma première et ma principale mission a été l'évolution d'AMAT. Ce projet visait à améliorer les performances des systèmes de surveillance existants en intégrant de nouvelles fonctionnalités et en corrigeant des bugs. Le système précédent était contraignant car si un client, dans ce cas Kermené¹, souhaitait faire évoluer son référentiel, il devait forcément passer par E-Mothep. Cela ralentissait le processus et devenait rapidement source d'erreurs et de confusions. Il a donc été décidé de créer une API qui permettrait de mettre automatiquement à jour les référentiels AMAT. Tout un système était donc déjà en place qui permettait d'analyser et de générer les rapports, celui-ci est schématisé dans la *figure 2.1* à la page 10.

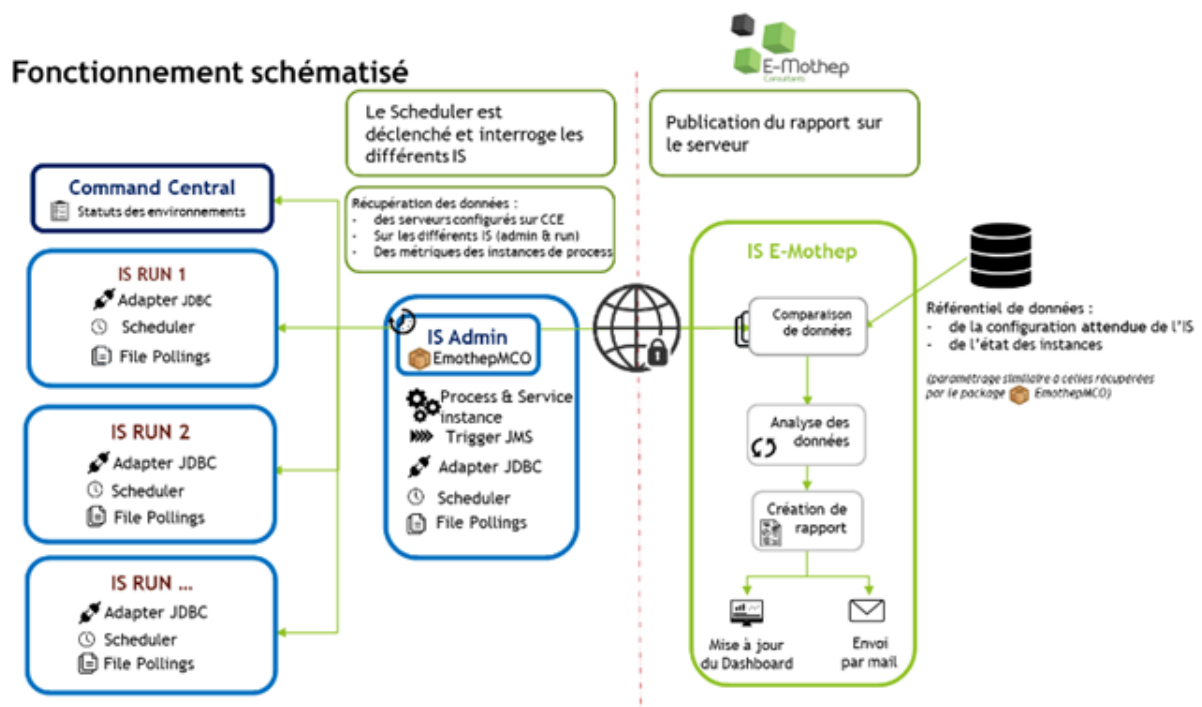


FIGURE 2.1 – Schéma explicatif du fonctionnement d'AMAT

Ce schéma explique que lorsque le scheduler se déclenche, les IS sont interrogés afin de connaître l'état de leurs assets. En cas de conflit entre les données collectées et la base de données servant de référentiel, un défaut est détecté et est signalé dans le rapport avec toutes les informations nécessaires pour comprendre la raison.

Ce système est opérationnel mais certains services ne fonctionnaient pas comme ils le devaient, il fallait donc aussi corriger ces problèmes en plus de développer

1. Acteur majeur de l'industrie agro-alimentaire française, filiale de Leclerc.

l'API qui automatise la mise à jour des référentiels.

Cela a été le cas par exemple avec les services de purges. Les serveurs génèrent chaque jour de nombreux fichiers de logs, la plupart sont purgés automatiquement par les serveurs eux-mêmes mais ce n'est pas le cas de tous. Des services ont été développés pour purger ces fichiers, mais ceux-ci ne renvoyaient aucun retour. Il était ainsi impossible de savoir combien de fichiers avaient été supprimés, ou si le service avait émis une erreur. Dans ce cas-là je n'ai pas dû créer mais faire évoluer ces services pour qu'ils permettent de mieux comprendre les actions qu'ils ont effectuées en renvoyant plus d'informations sur ce qui a été fait. Le développement des services s'effectuait avec Software AG Designer, un outil essentiel pour l'intégration que j'ai découvert grâce à cette première mission.

Cette première mission m'a permis de découvrir les outils principaux que l'entreprise utilise. Ma deuxième mission est moins concentrée sur la découverte de nouveaux outils et plus sur la découverte d'une facette du métier : la rédaction de documentation. Ma deuxième mission a été de rédiger de la rétro documentation pour les projets internes à l'entreprise. La rétro documentation consiste à faire une documentation pour un projet déjà existant, car soit la documentation n'a jamais été faite, soit qu'elle est devenue obsolète. Ces projets possédaient bien une documentation utilisateur, mais aucune documentation technique, ce qui rendait compliqué de savoir quelles méthodes servaient à quoi ou quelles méthodes étaient déclenchées par quelles actions utilisateur, et notamment quelles méthodes interagissaient avec la base de données. La documentation technique est la description du fonctionnement d'un système, d'un logiciel, ou d'un service, destiné à des informaticiens, alors que la documentation utilisateur est l'explication du fonctionnement de l'interface d'un produit destiné aux utilisateurs finaux. Cette deuxième mission m'a permis de découvrir plusieurs outils en documentant du code qui les utilisaient.

La dernière mission sur laquelle j'ai travaillé est l'un des projets internes : La base de connaissances. L'idée est que, dès qu'un collaborateur rencontre un problème, il note dans la base de connaissance la solution qu'il a trouvée, afin que les futurs collaborateurs n'aient pas à devoir rechercher et retrouver à nouveau une solution. C'est un investissement dans le temps, car la base deviendra de plus en plus utile au fil des années. Sur ce projet, ma première tâche a été de trouver une solution pour récupérer correctement le nom et le prénom d'un utilisateur. Pour ma deuxième tâche, je devais améliorer la gestion des erreurs dans la partie back en combinant le projet avec une gestion d'erreur particulièrement efficace et modulable qui avait été faite pour un autre projet. Pour cela je devais utiliser et comprendre les messages JMS et les topics car ce sont des outils essentiels dans l'utilisation du module de gestion d'erreur que je devais implémenter.

2.2 Méthodes et Outils Utilisés

2.2.1 AMAT - Développement de l'API

La mission d'évolution d'AMAT étant trop compliquée pour un simple stagiaire, mon responsable pour ce projet, Monsieur Geoffrey Vanaerdewegh a découpé ce qu'il fallait faire afin de l'adapter à mon niveau et m'a confié des tâches assez précises et concises. AMAT surveille des éléments appelé asset comme des adapters, des schedulers, des filePollings, des triggers. . . Chaque asset possède plusieurs attributs dont on veut surveiller et qui représente chacun une entrée dans la base de données. J'ai d'abord analysé celle-ci avec DBeaver pour comprendre comment étaient stockés les informations puis j'ai développé un service que j'ai appelé `synchronizeReferentiel` prenant en entrée trois informations : l'identifiant de l'asset, le type d'asset et le type de CRUD. CRUD étant l'un des type de modifications suivants : Créer, Lire, Mettre à jour, Supprimer. L'API utilise ce service unique qui gère chaque endpoints. Un endpoint étant un point de communication d'une API, soit une URL qui permet à un client (une application ou un utilisateur) de communiquer avec le service web en envoyant des requêtes HTTP à cette URL spécifique et en recevant des réponses structurées en retour. Chaque endpoint peut offrir différentes fonctionnalités ou ressources, définies par les développeurs de l'API. Ce service fait ensuite appel à de nombreux sous-service² en fonction des paramètres donnés en entrée car chaque asset étant très différents, ils ne se géraient pas du tout de la même manière.

id	asset_type	asset_fieldname	isimportant
1	filepolling	fileNameFilter	0
2	filepolling	completionDir	0
3	filepolling	enabled	1
4	filepolling	errorDir	0
5	filepolling	monitorDir	0
6	filepolling	filePollingInterval	0
7	filepolling	workDir	0
8	filepolling	processingService	0
9	filepolling	portAlias	1
10	filepolling	serverName	1
11	adapter	connectionState	0
12	adapter	packageName	0
13	adapter	connectionAlias	1
14	adapter	serverName	1
15	scheduler	execState	1
16	scheduler	interval	1
17	scheduler	service	1
18	scheduler	old	1
19	scheduler	serverName	1
20	trigger	status	1

id	id_host	id_asset_field	asset_name
222 307	376	11	KrnlDbcEsbpkgWm
222 308	376	12	KrnlDbcEsbpkgWm
222 309	376	45	KrnlDbcEsbpkgWm
222 310	376	11	KrnlDbcEsbpkgWm
222 311	376	12	KrnlDbcEsbpkgWm
222 312	376	45	KrnlDbcEsbpkgWm
222 313	376	11	KrnlDbcSupervision
222 314	376	12	KrnlDbcSupervision
222 315	376	45	KrnlDbcSupervision
222 316	376	11	KrnlDbcSupervision
222 317	376	12	KrnlDbcSupervision
222 318	376	45	KrnlDbcSupervision
222 319	376	11	KrnlDbcSupervision
222 320	376	12	KrnlDbcSupervision
222 321	376	45	KrnlDbcSupervision
222 322	376	11	KrnlDbcSupervision
222 323	376	12	KrnlDbcSupervision
222 324	376	45	KrnlDbcSupervision
222 325	377	11	KrnlDbcAchpkipBoa
222 326	377	12	KrnlDbcAchpkipBoa

FIGURE 2.2 – Les tables utilisées pour le projet d'évolution d'AMAT

Analyser la base de données m'a permis de comprendre ce qui était attendu en entrée et en sortie du flux du service et de chaque sous-service. Dans la base de données, il y a deux tables essentielles pour ce projet, que je montre en *figure 2.2* à la page 12 : Une table `ASSET_FIELD_CONFIGURATION`, à gauche dans l'image, avec chaque types d'asset et ses attributs qui doivent être enregistré dans

2. Ce que j'appelle sous-service est tout service que j'utilise dans un autre de mes services. Il n'y a pas de différence technique, tout service peut être appelé dans n'importe quel autre service.

la base de données et une autre table ASSET, à droite dans l'image, qui elle contient les données des assets.

Il me suffisait donc d'observer la première table pour savoir quels étaient les attributs à enregistrer puis de chercher parmi les services pré-faits pour en trouver un qui retourne les informations qui m'intéresse. Enfin, j'intégrais ce service à l'un de mes sous-service qui gérât ce type d'asset et qui sera appelé par le service synchronizeReferentiel. C'est l'un de ces sous-services qui enregistre les informations de l'asset dans la table ASSET.

DBeaver est un outil de gestion de base de données intuitif avec un éditeur SQL avancé et qui supporte de nombreuses bases de données tel que MySQL, PostgreSQL, SQLite, Oracle, SQL Server, et d'autres...

Le service utilise l'identifiant de l'asset pour récupérer les informations de l'asset sur l'Integration Server et nous devons cartographier³ ces informations sur le documentType, qui a une structure de données XML, récupéré du packaging E-Mothep du serveur AMAT. Puis en fonction du type de CRUD entré, le service ajoute, met à jour, ou supprime l'asset de la base de données. Les services utilisés pour récupérer les informations d'un asset à partir de son identifiant sont très différents les uns des autres, et chaque asset demande plus ou moins d'informations pour être ajouté dans la base de données, c'est pour cela qu'il est nécessaire de préciser le type d'asset en entrée du service. Chaque service a été développé avec l'interface graphique de Software AG Designer, bien que j'ai eu recours à coder directement en Java pour certains services plus complexes. Software Ag Designer est une surcouche d'Eclipse IDE qui permet de développer des services gérant des flux en low-code⁴.

Enfin on peut maintenant appeler en HTTP les services E-Mothep permettant d'insert, d'update ou de delete, selon le type de CRUD renseigné en entrée. J'ai réalisé des tests unitaires avec Postman sur chaque endpoint de l'API pour vérifier que tout fonctionnait correctement. J'ai documenté chacun de mes résultats dans un cahier de tests. Postman est un logiciel qui permet de créer des requêtes HTTP pour tester des API et vérifier leurs réponses. POSTMAN est largement adopté dans l'industrie car il permet aussi d'automatiser les tests, de créer des collections de requêtes et de partager des configurations avec l'équipe, facilitant le développement et la maintenance d'API.

3. Assigner des valeurs d'entrées aux valeurs d'un ensemble.

4. Désigne les outils de développement permettant de coder à partir d'une interface graphique sans avoir nécessairement recours à une programmation classique.

2.2.2 AMAT - Évolution des services de purges

Une fois l'API développée, ma mission a été de corriger les services existants qui étaient incomplets ou qui contenaient des erreurs. Pour savoir ce qu'il y avait à faire et qui s'en occupait, nous avons utilisé MANTIS.

MantisBT est un système de suivi de bugs open source. C'est un outil très pratique qui permet aux équipes de signaler, suivre et résoudre les bugs de manière organisée. Dans MANTIS, un ticket correspond à un bug ou à une tâche. On peut y indiquer sa sévérité, son avancement, la/les personne(s) affectée(s) à sa résolution, etc...

Un exemple est l'amélioration des services de purges. Ceux-ci ne purgeaient pas correctement les fichiers ni purgeaient tous les fichiers nécessaires. J'ai donc développé des nouveaux services de purges pour les remplacer et compléter ce qu'il faut purger. J'ai utilisé WinSCP pour accéder aux fichiers de logs sur le serveur et vérifier qu'ils ont bien été supprimés et tester mes services de purge.

WinSCP est un client SSH de transfert de fichiers open source pour Windows. C'est un outil qui permet de transférer des fichiers entre un ordinateur local et un serveur distant en utilisant plusieurs protocoles tels que FTP, SFTP et SCP⁵.

Un autre exemple : la plage de date était incorrecte dans les rapports d'AMAT généré. Celle-ci est supposé indiquer la plage de date entre le précédent rapport et le prochain rapport programmé par les schedulers. Cette plage de date était incorrecte, car elle affichait toujours la même chose. Je le montre dans la *figure 2.3* à la page 14. À gauche, un exemple de rapport incorrect. À droite, un exemple de rapport correct. On voit le problème de plage de date mentionné plus tôt dans le rapport de gauche. Malgré le fait que le rapport fut lancé vers 9h du matin, la plage de date indique 2024-04-29 16 :00 :00 au 2024-04-30 15 :59 :59 au lieu de 2024-04-29 9 :30 :00 au 2024-04-30 12 :30 :00.

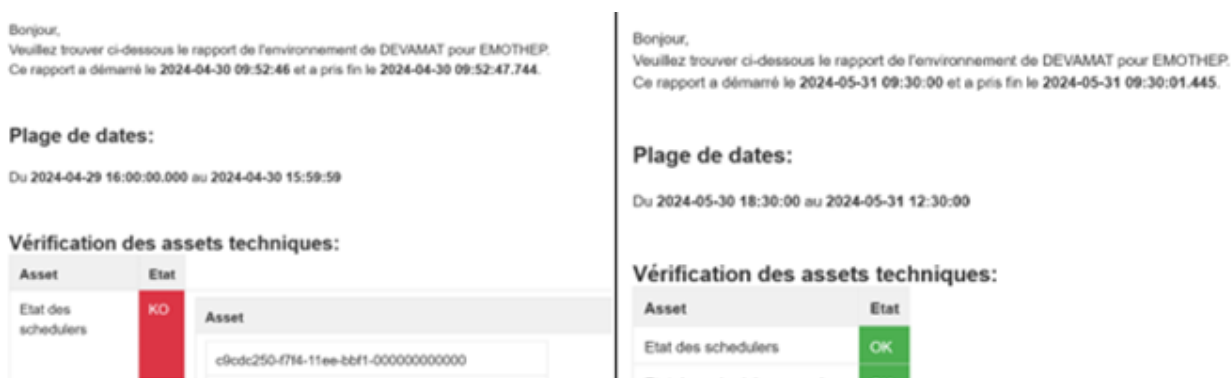


FIGURE 2.3 – Comparatif avant à gauche et après à droite de l'implémentation de ma solution pour le problème de plage de date

J'ai refactorisé les services qui génèrent ces rapports pour les rendre plus modulables et pour corriger ce bug. A Droite, on voit maintenant que la plage de date est correcte et correspond bien aux dates de lancement de rapport des scheduler.

5. Ce sont tous des protocoles pour la gestion de fichiers et la communication entre ordinateurs sur un réseau. Voir le Glossaire à la page 23 pour la signification des sigles et des acronymes.

2.2.3 Rétro documentation

Pour réaliser la mission concernant la rédaction de rétro-documentation sur les projets internes, j'ai scruté le code sur le GitHub et j'ai noté dans un document toutes les interactions avec la base de données Firebase.

GitHub est une plateforme de développement collaboratif basée sur Git, un système de contrôle de version. Elle permet aux développeurs de gérer et de stocker leurs codes sources pour y accéder de n'importe où, mais aussi de suivre les modifications, de collaborer avec d'autres développeurs et d'annuler des modifications accidentelles ou malveillantes en assurant une traçabilité complète grâce à un historique de chaque interaction. GitHub, GitLab ou d'autres plateformes de gestion de code sont des outils indispensables de développement, car elle rend possible le travail en équipe à grande échelle.

Firebase est une plateforme de développement d'applications mobiles et web « BackEnd as a service » (BaaS) proposée par Google. BaaS est un modèle de service où les fournisseurs cloud offrent une infrastructure BackEnd complète et des fonctionnalités prêtes à l'emploi pour les applications web et mobiles, permettant aux développeurs de se concentrer sur la partie FrontEnd. Le Backend est toute la partie que l'utilisateur ne voit pas, mais qui lui permet de réaliser des actions sur un site ou une application. Le Frontend désigne les éléments d'un site internet ou d'une application que les utilisateurs voient et interagissent avec. Firebase offre une base de données en temps réels, une base de données NoSql⁶, l'authentification utilisateur, l'hébergement de fichiers et de sites web, ainsi que des fonctionnalités d'analyses. Firebase est largement adoptée pour accélérer le développement d'applications et améliorer leur performance et leur fiabilité.

Une fois que j'ai complété ce document, j'ai parcouru les projets internes pour vérifier quelles actions de l'utilisateur déclenchaient quelles méthodes dans le code et donc quelles interactions avec la base de données, et j'ai rajouté ces observations sur mon document. J'ai ensuite divisé ce document pour chaque projet interne, afin qu'ils aient chacun leurs propres documentations. Ce projet utilise le framework Angular, le code source était donc en typescript, un langage que je n'avais pas eu l'occasion d'utiliser avant. Devoir analyser ce code source m'a donc permis de découvrir Angular et Typescript, deux outils importants pour un développeur front-end.

Angular est un framework open source de développement d'applications web front-end. Il utilise TypeScript pour permettre un développement structuré et typé. Angular offre une architecture basée sur les composants, ce qui facilite la réutilisation, la maintenance et la gestion du code.

TypeScript est un sur-ensemble de JavaScript⁷, conçu pour ajouter des types statiques au langage JavaScript. Il améliore la scalabilité, la maintenabilité et la robustesse des projets, ce qui le rend particulièrement apprécié des utilisateurs et des entreprises.

6. Catégorie de systèmes de gestion de bases de données non relationnelles.

7. Langage de programmation interprété non typé orienté objet principalement utilisé pour le développement web.

2.2.4 Base de connaissances - Gestion des noms et des prénoms

Dans ce projet, on affichait le nom de famille de l'utilisateur à partir de son identifiant LDAP : « prénom.nom ». C'était un bricolage maladroit, car les identifiants LDAP n'ont pas d'accents, et ils peuvent être incorrects dans les cas où le nom de famille trop long et a dû être raccourci ou s'il y a deux personnes avec le même nom dans l'entreprise car forcément il y aurait un identifiant LDAP avec un nom de famille modifié pour éviter les doublons. Il fallait donc trouver une solution pour récupérer le nom et le prénom ailleurs. Grâce à la rétro documentation que j'ai rédigé préalablement sur les projets internes, je savais que la base de données Firebase avait les informations que l'on souhaitait, mais pour obtenir ces informations, il fallait l'identifiant de l'utilisateur. Je me suis rendu compte, avec l'aide de mes collègues, que lorsque l'on se connecte avec l'identifiant LDAP et son mot de passe, on obtient un jeton (token) JMT qui sert à l'authentification. En décodant ce jeton, on obtient l'identifiant nécessaire pour la base de données, il n'y a plus qu'à faire une requête sur la table avec cet identifiant pour obtenir le bon nom et prénom de l'utilisateur.

2.2.5 Base de connaissances - Gestion des erreurs

Dans ce projet, la gestion des erreurs est incomplète. Une gestion des erreurs plus intéressante et modulable a été faite pour un autre projet. Je dois récupérer cette gestion d'erreur et l'implémenter dans ce projet-là. Je devrai ensuite utiliser POSTMAN pour accéder à l'URL de l'API et tester si cela renvoie les bons messages d'erreurs. Pour cette mission, j'ai dû refaire mon environnement de travail car il était mal configuré, notamment pour intégrer Git à mon Software AG Designer, ainsi que pour importer les packages nécessaires. Pour cela j'ai dû les exporter en fichier ZIP du serveur de développement de l'entreprise et les importer dans mon serveur local. Je devais ensuite suivre la documentation du module de gestion d'erreurs pour modifier les services appropriés et pour configurer les serveurs afin d'intégrer le module. Par exemple, je devais vérifier dans la base de données quelles valeurs l'objet que l'on souhaite ajouter sont obligatoires pour les rendre aussi obligatoires à la saisie dans le code. Je devais aussi m'assurer que certains services renvoient une exception si certaines valeurs étaient manquantes et gérer ces exceptions. Pour ce projet, j'ai dû envoyer les exceptions détectées par message JMS. Java Message Service est une API pour permettre un dialogue standard entre des applications ou des composants grâce à des composants de manière asynchrone. Elle permet donc d'utiliser des services de messagerie dans des applications java en s'abonnant à un émetteur, dont les messages se font transporter grâce à des topics et des queues. Une queue envoie le message à un seul abonné, alors qu'un topic envoie le message à chaque abonné. Cette mission n'est pas encore terminée au moment où je rédige ce rapport de stage.

Pour comprendre le code d'un projet, pour résoudre les problèmes auxquels j'ai été confrontés et pour apporter ma pierre à l'édifice dans les projets de l'entreprise, j'ai dû découvrir et apprendre de nombreux nouveaux outils que je ne connaissais pas ou peu avant ce stage.

Analyse des Résultats et Compétences Acquis

3.1 Résultats Obtenus

3.1.1 Évolution d'AMAT

La mission principale de l'évolution d'AMAT a conduit à des améliorations significatives dans ses performances et ses fonctionnalités. L'API créée permet désormais aux clients de mettre à jour leurs référentiels sans intervention manuelle, réduisant ainsi les erreurs et accélérant le processus. Les erreurs sont aussi plus facilement détectables et donc plus facilement corrigeables.

3.1.2 Services de Purges

L'amélioration des services de purges a résolu plusieurs problèmes critiques. En effet, les nouveaux services de purge suppriment désormais correctement tous les fichiers nécessaires, libérant ainsi de l'espace de stockage et optimisant les performances des serveurs. De plus les services développés fournissent maintenant des retours détaillés, indiquant le nombre de fichiers supprimés et signalant les erreurs éventuelles, ce qui améliore le suivi et la maintenance.

3.1.3 Rétro-documentation

La rétro-documentation des projets internes a apporté des bénéfices notables. Les documents techniques produits permettent au développeur de comprendre rapidement les interactions entre les méthodes et la base de données, facilitant ainsi la maintenance et le développement futurs. La documentation complète pour chaque projet interne améliore la qualité des développements et réduit les temps d'apprentissage pour les nouveaux arrivants. Je n'ai pas été particulièrement satisfait de la documentation que j'ai rendue, car j'ai rarement fait de documentation avant cela, et j'ai eu beaucoup de mal à comprendre ce que faisaient et comment fonctionnaient les projets internes à partir de leur code source. De ce fait, bien que j'ai fait du mieux que j'ai pu, je pense que la documentation que j'aie rédigée reste très imparfaite et peut être améliorée.

3.1.4 Base de Connaissance

Mes apports sur le projet de base de connaissance sont légers, mais importants. L'implémentation de la solution pour récupérer les noms et prénoms des utilisateurs via FireBase a amélioré la précision des informations affichées, et c'est tout de même plus agréable de ne pas avoir son nom mal écrit. L'intégration d'un système de gestion des erreurs plus modulable et plus performant a renforcé la robustesse du projet, même si cette mission est toujours en cours.

3.2 Résultats Obtenus

3.2.1 Compétences Techniques

Durant ce stage, j'ai découvert de nombreux nouveaux logiciels que je ne connaissais pas ou peu. Évidemment je ne suis pas devenu un expert de ces outils en seulement dix semaines, mais cela m'a permis de connaître leur fonction, d'apprendre à les utiliser ; ainsi que leur place dans l'entreprise. J'ai appris à utiliser Software AG Designer grâce à la formation de Monsieur Geoffrey Vanaerdewegh et j'ai pu appliquer ces connaissances lors du développement de l'API pour l'évolution d'AMAT. En utilisant ces connaissances sur un projet concret, j'ai pu bien comprendre comment fonctionner cet IDE. Mon expérience avec DBeaver et FireBase m'a permis de comprendre et de manipuler des bases de données complexes, améliorant ainsi mes compétences en SQL et NoSQL. Utiliser Postman pour les tests unitaires et GitHub pour la gestion du code source m'a permis de me familiariser avec des outils de développement couramment utilisés dans l'industrie.

3.2.2 Compétences en Gestion de Projet

Travailler dans cet environnement m'a apporter des connaissances de base sur le déroulement d'un projet. J'ai appris à communiquer, à travailler en équipe, à m'adapter, à collaborer et à répartir et prioriser les tâches importantes. L'utilisation de MANTISBT pour le suivi des bugs m'a permis de comprendre l'importance de la gestion des problèmes et de leur résolution systématique. De même pour Git, car la gestion des versions¹ d'un projet est très importante surtout quand plusieurs équipes travaillent sur un même projet qui dure dans le temps. En effet, pouvoir revenir en arrière en cas d'accident ou d'erreur dans les choix de conception est essentiel. Le projet de rétro documentation m'a fait me rendre compte de l'importance de documenter lors du développement d'un projet. Revenir en arrière pour relire son code ou analyser le code d'un autre, des mois voire des années après sa rédaction rend la création d'une documentation beaucoup plus éprouvant.

1. Création et gestion de plusieurs versions de produits, qui ont toutes la même fonction générale, mais sont améliorées, mises à niveau ou personnalisées.

3.2.3 Compétences Relationnelles

Collaborer avec les différentes équipes et avec ses collègues afin de trouver des solutions à des problèmes ou de présenter des solutions techniques m'a aidé à développer des compétences en communication, essentielles pour le travail en équipe et la gestion de projet. Grâce aux projet de rétro documentation, j'ai pu apporter un support technique aux futurs collaborateurs d'E-Mothep et rédiger de la documentation technique m'a appris à transmettre des informations techniques de manière claire et efficace.

Conclusion

Mon stage chez E-Mothep a été une expérience formatrice et enrichissante à plusieurs égards. Il m'a permis de participer activement à des projets concrets et d'apporter des contributions significatives à l'entreprise. Le projet principal sur lequel j'ai travaillé, l'évolution d'AMAT, visait à améliorer les performances des systèmes de surveillance en intégrant de nouvelles fonctionnalités et en corrigeant des bugs. La mise en place de l'API pour la mise à jour automatique des référentiels a permis d'optimiser les processus internes, réduisant les erreurs et accélérant les délais de mise à jour pour les clients. Cette automatisation a non seulement augmenté l'efficacité opérationnelle mais a également améliorée la satisfaction des clients en leur offrant une plus grande autonomie.

J'ai apporté une documentation sur les projets internes qui en manquait cruellement. La documentation complète pour chaque projet interne devrait améliorer la qualité des développements et réduira les temps d'apprentissage pour les nouveaux arrivants. Rédiger cette documentation a été pour moi la partie la plus complexe de ce stage. J'avais peu d'expérience en rédaction de documentation et en analyse de code, et je ne connaissais pas beaucoup Angular et typescript. Devoir donc rédiger de la documentation technique qui demande une compréhension approfondie du code du projet n'a vraiment pas été simple. C'est en persistant d'analyser le code et la base de données et en cherchant à comprendre comment ce code s'engrenait dans le reste du projet que j'ai fini par réussir à rédiger cette documentation.

Ces projets internes représentent une partie importante du développement de l'entreprise. A terme, ces projets internes bénéficieront grandement à l'entreprise en améliorant et fluidifiant certains processus qui sont aujourd'hui compliqués, voire ils pourront être vendus à d'autres entreprises et ainsi générer des revenus non négligeables pour l'entreprise.

Le projet interne que j'ai trouvé le plus intéressant est la base de connaissances, j'aime beaucoup cette idée de regrouper les solutions aux problèmes communs que les collaborateurs pourront de nouveau rencontrer dans le futur, je suis content d'avoir pu participer à ce projet et d'y avoir apporté ma pierre à l'édifice. Trouver comment afficher le nom et le prénom d'un utilisateur s'est révélé être plus complexe que prévu, mais grâce à mes analyses sur les projets internes pour la rétro documentation j'ai pu trouver une piste pertinente, et même si j'étais toujours bloqué, je n'ai eu plus qu'à oser demander de l'aide pour atteindre mon objectif qui n'était alors plus qu'à portée de main.

Sur le plan humain et personnel, ce stage m'a permis de développer non seulement des savoir-faire mais aussi des savoir-être. En savoir-faire, j'ai approfondi mes connaissances en développement d'API avec Software AG Designer et Post-Man, en gestion de base de données avec DBeaver et FireBase, et en gestion de projets avec Mantis et GitHub. En Savoir-être, j'ai principalement travaillé en équipe, ce qui m'a permis d'améliorer mes compétences en communication et en collaboration. J'ai appris à exprimer clairement mes idées, à écouter et à intégrer

les feedbacks de mes collègues mais aussi demander de l'aide quand j'étais bloqué sans pour autant devenir dépendant de leurs aides.

L'expérience que j'ai acquise durant ce stage m'a aidé à confirmer mon projet professionnel en renforçant mon intérêt pour le développement de logiciels. J'espère pouvoir continuer mes études en alternances dans ce cadre. Je me sens maintenant plus confiant dans mes compétences et motivé pour poursuivre une carrière dans ce domaine.

Ce stage chez E-Mothep a été une étape cruciale dans mon parcours professionnel. J'ai appris énormément de choses en un court laps de temps et ces compétences acquises me seront précieuses dans ma future carrière. Je suis reconnaissant pour l'opportunité qui m'a été donnée, je tiens à remercier tous les collaborateurs d'E-Mothep pour leur soutien, leurs conseils et pour cette précieuse expérience qu'ils m'ont apportés.

Bibliographie

Site d'E-Mothep : <https://e-mothep.fr/>

Site de gestion des stages de l'IUT : <https://stages.iut-info.univ-lille.fr>

Software AG :

— Designer : <https://documentation.softwareag.com/webmethods/designer/sdf10-5/index.html>

— Documentation : <https://documentation.softwareag.com>

— WebMethods : https://www.softwareag.com/en_corporate/platform/integration-apis/webmethods-integration.html

— WebMethods Wiki : <https://en.wikipedia.org/wiki/WebMethods>

— Integration Server : <https://www.techtarget.com/searchcio/definition/integration-server>

— Universal Messaging Server : https://documentation.softwareag.com/universal_messaging/num10-7/10-7_UM_webhelp/index.html

Angular : <https://angular.io>

DBeaver : <https://dbeaver.io>

FireBase : <https://firebase.google.com>

GIT : <https://www.git-scm.com>

GitHub : <https://github.com>

PostMan : <https://www.postman.com>

MantisBT : <https://mantisbt.org>

WinSCP : <https://winscp.net>

Glossaire

A

- AMAT : Automated Monitoring and Analysis Tools : Outils automatisés de surveillance et d'analyse
- AMOA : Assistance à la maîtrise d'ouvrage
- API : Application Programming interface : Interface de programmation d'applications, Une API est un moyen permettant à deux ou plusieurs programmes ou composants informatiques de communiquer entre eux. C'est un type d'interface logicielle, offrant un service à d'autres logiciels
- Asset : Un élément surveillé par AMAT, dont chacun de ses attributs représente une entrée dans la base de données.

B

- BaaS : BackEnd as a service : Back-end en tant que service, le service proposé s'occupe entièrement du back-end, en mettant à disposition une API pour s'y connecter.
- Back-end : Toute la partie que l'utilisateur ne voit pas, mais qui lui permet de réaliser des actions sur un site ou une application.
- Best of Breed : Plutôt que de se fier à un seul éditeur pour choisir un seul logiciel d'entreprise multi-tâche, trouver le meilleur logiciel pour chaque domaine d'application individuel.
- BUT : Bachelor Universitaire de Technologie

C

- Cartographier : Dans ce cas-là, signifie faire correspondre les objets d'un ensemble avec les objets d'un autre ensemble.
- CRUD : Create, Read, Update, Delete, signifiant Créer, Lire, Mettre à jour, Supprimer

D

- Documentation technique : Description du fonctionnement d'un système, d'un logiciel, ou d'un service, destiné à des informaticiens.
- Documentation utilisateur : Explication du fonctionnement de l'interface d'un produit destiné aux utilisateurs finaux.
- DUT : Diplôme universitaire de technologie

E

- Eclipse : IDE open source utilisé principalement pour le développement java.
- Endpoint : Point de communication d'une API.
- ESB : Signifie Enterprise Service Bus, c'est une architecture de middleware qui permet l'intégration des différentes applications et services au sein d'une entreprise. Il facilite la communication entre divers systèmes en utilisant un bus centralisé qui transmet les données et les messages de manière fiable et flexible.

F

- Flux : Écoulement de données entre les services, c'est le cheminement des données et des opérations.
- FTP : File Transfer Protocol : protocole qui assure le transfert de fichiers d'un utilisateur à un autre
- Front-end : Désigne les éléments d'un site internet ou d'une application que les utilisateurs voient et interagissent avec.

G

- Git : Système de contrôle de version open source très populaire.

H

- Hardware : éléments matériels d'un système informatique.
- HTTP : Hypertext Transfer Protocol : Protocole de transfert hypertexte.

I

- IDE : Integrated Development Environment : Application logicielle aidant les programmeurs à développer en offrant outils facilitant le développement.
- IUT : Institut Universitaire de technologie.

J

- JavaScript : Langage de programmation interprété non typé orienté objet principalement utilisé pour le développement web.
- JMT : JSON Web Token, utilisé pour l'authentification.
- JSON : JavaScript Object Notation, format de fichier qui utilise un texte lisible par l'homme pour stocker et transmettre des objets de données constitués de paires attribut-valeur. Concurrent de YAML.
- JMS : Java Message Service est une API Java qui permet aux applications de communiquer entre elles de manière asynchrone via des messages. JMS est utilisé pour la communication entre les composants d'une application distribuée et permet la dé-couplage des composants, ce qui facilite la scalabilité et la flexibilité.

K

- Kermené : Client d'E-Mothep. Acteur majeur de l'industrie agro-alimentaire française, filiale de Leclerc.

L

- Low-Code : Désigne les outils de développement permettant de principalement coder à partir d'une interface graphique sans nécessairement devoir écrire des lignes de codes. A ne pas confondre avec le no-code.
- LDAP : Lightweight Directory Access Protocol est un langage ouvert et multi plateforme utilisé entre un client et un serveur lors d'une connexion permanente. Principalement utilisé durant ce stage pour l'authentification.

M

- MantisBT : Mantis Bug Tracker, système de suivi de bugs open source populaire.
- MiddleWare : Interlogiciel, c'est un logiciel qui se trouve entre un système d'exploitation et les applications qui s'exécutent sur celui-ci.

N

- NoSql : catégorie de systèmes de gestion de bases de données non relation-

nelles

- No-Code : Désigne les outils de développement permettant de coder à partir d'une interface graphique sans écrire de code informatique.

O

- Outil : En informatique, cela désigne un logiciel ou un type de logiciel qui donne à l'utilisateur la possibilité d'effectuer une tâche précise.

P

- Projet : Mise en œuvre des ressources humaines et matérielles particulières, sur une période déterminée.

Q

- Query : Une demande d'information souvent extraite d'une base de données.
- Queue : File d'attente, les messages JMS sont consommés par un seul receveur. Une fois qu'un message est lu, il est retiré de la queue.

R

- Référentiel : Ensemble auquel doivent être référencé les assets AMAT afin de s'assurer qu'ils sont corrects.
- REST : REpresentational State Transfer : Transfert d'État représentatif.
- Rétro Documentation : Faire une documentation pour un projet déjà existant, car soit la documentation n'a jamais été faite, soit qu'elle est devenue obsolète.

S

- SAS : Société par actions simplifiées.
- SCP : Secure Copy Protocol : outil pour copier de manière chiffrée des fichiers entre postes distants.
- Services : Un service est un programme dans le cadre de Software AG Designer. C'est une unité fonctionnelle réutilisable exécutant une tâche précise allant de simplement transformer des données ou effectuer de simples opérations à l'orchestration de flux complexe ou l'orchestration d'applications multiples.
- Sous services : Ce que j'appelle sous-service est tout service que j'appelle dans l'un de mes services et ceux afin de les différencier. Il n'y a donc aucune différence technique, tout service peut être appelé dans n'importe quel autre service.
- SFTP : SSH/Secure FTP : protocole sécurisé qui assure le transfert de fichiers d'un utilisateur à un autre.
- SI : Système d'information.
- SOA : Service oriented architecture : Architecture orientée service.
- SOAP : Simple Object Access Protocol : Protocole d'accès aux objets simples.
- Software AG : (Software : Logiciel informatique) Société multinationale allemande de logiciels qui développe des logiciels d'entreprise pour la gestion des processus métier, l'intégration et l'analyse du Big Data.
- SQL : Langage populaire de requêtes aux bases de données relationnelles.
- SSH : Secure SHell : protocole de communication réseau qui permet à deux ordinateurs de communiquer.

- Surcouche : Logiciel installé par-dessus un autre, s'en servant comme base.

T

- Tech-lead : Responsable technique, choisit les technologies qui seront utilisées pour le projet.
- Token : Object d'identification, représentant le droit d'effectuer certaines opérations.
- Topic : Sujets, les messages sont publiés sur un topic et peuvent être consommés par plusieurs abonnés. Tous les abonnés actifs au moment de la publication recevront une copie du message.

U

- URL : Uniform Resource Locator : Localisateur de ressources uniformes, identifiant unique utilisé pour localiser une ressource sur Internet. On l'appelle également adresse Web.

V

- Versions : création et gestion de plusieurs versions de produit, qui ont toutes la même fonction générale, mais sont améliorées, mises à niveau ou personnalisées.

W

- Webmethods : Entreprises axées sur l'intégration d'applications et l'intégration de processus métiers, rachetées par Software AG.

X

- XML : Extensible Markup Language : Langage de balisage extensible.

Y

- YAML : un langage de sérialisation de données lisibles par l'homme qui est souvent utilisé pour écrire des fichiers de configuration. Concurrent de JSON.

Z

- ZIP : un format de fichier courant utilisé pour compresser un ou plusieurs fichiers ensemble en un seul emplacement.

Annexe

Exploration de Software AG Designer

Software AG est une entreprise allemande spécialisée dans les logiciels d'entreprise, offrant des solutions pour l'intégration de systèmes, la gestion des processus métiers, l'Internet des objets et la gestion des données. Elle collabore avec un réseau mondial de plus de 400 partenaires pour fournir des solutions technologiques avancées et soutenir la transformation numérique de ses clients. En France, Software AG compte plusieurs partenaires stratégiques, tel que Accenture, Cap Gemini, Atos, CGI et E-Mothep. L'entreprise fait partie de ses partenaires depuis plusieurs années. En effet, l'un des outils principaux qu'utilise E-Mothep pour l'intégration est Software AG Designer.

Designer est un outil intégré de développement d'applications et de gestion des processus métiers, faisant partie de la suite webMethods de Software AG. Il permet de créer des processus métier, des services d'intégration, des applications web et des API à l'aide d'interfaces graphiques intuitives. Au début de ma formation, j'ai trouvé perturbant de pouvoir glisser-déposer des composants pour construire des applications sans avoir à écrire beaucoup de code.

On peut voir à gauche de la *figure A.1* la liste des paquetages. Ce sont les mêmes paquetages qu'en Java, soit un dossier qui regroupe des fichiers source liés, facilitant ainsi la modularité, la réutilisabilité et la gestion des dépendances dans les projets de développement logiciel. On voit à droite de l'image la partie où l'on code les services. Chaque service peut être appelé dans un autre service, ils sont représentés par les icônes de flèches bleues. Les autres icônes représentent des structures de programmation classique.

Sont présents sur l'image dans l'ordre : la boucle (loop), l'embranchement (branch) et la séquence (sequence), qui est simplement une suite de ligne de code qui doivent être exécutés à la suite. On ne les voit pas dans la *figure A.1*, mais d'autres structures existent, tel que la carte (Map), pour gérer les valeurs, le répéteur (repeat), qui est un autre type de boucle et le « essai-capture » (try-catch) pour la gestion d'erreur. Designer supporte l'intégration avec divers systèmes d'en-

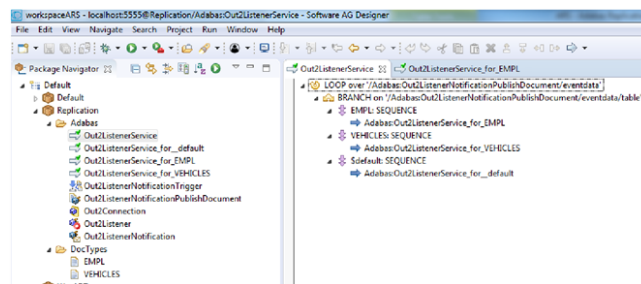


FIGURE A.1 – Interface de Software AG Designer

treprise, bases de données et services web, facilitant la connexion et l'automatisation des flux de travail au sein des organismes. L'outil permet de concevoir, tester et déployer des API REST et SOAP, ce qui aide les entreprises à créer des services réutilisables et à exposer leurs fonctionnalités à d'autres applications. Cela a été mon activité principale lorsque j'utilisais Designer durant mon stage. Designer permet de gérer le cycle de vie complet des services, de la conception à la mise en œuvre, en passant par le déploiement et la surveillance. Les fonctionnalités qui m'ont été les plus utiles durant le stage sont celles de simulation et de débogage qui permettent de tester les applications et les processus. Designer a des outils de débogage très puissants qui rends le repérage et la correction de bugs beaucoup moins pénible. Par exemple il est possible d'afficher des logs, d'exécuter un programme étape par étape, ou de stocker les variables d'entrées si le service ne marche plus afin de pouvoir reproduire la situation qui a causé l'erreur.

Pour résumer, Software AG Designer est un outil puissant pour le développement d'applications et la gestion des processus métier, offrant une interface intuitive et des fonctionnalités complètes pour répondre aux besoins complexes des entreprises modernes. Il facilite la création, l'intégration et l'optimisation des processus et des services, tout en augmentant la productivité des développeurs et en réduisant les risques d'erreurs.