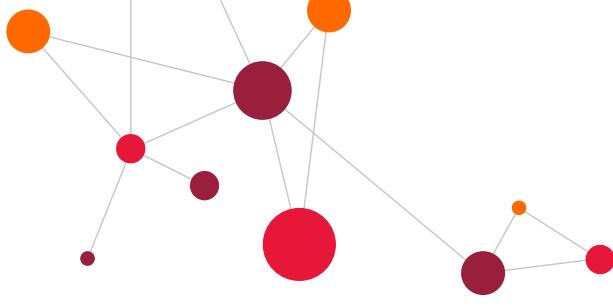




DOSSIER DE VALIDATION

Concepteur-développeur d'applications numériques

Nom Prénom	Laurin Maxime
Nom(s) Prénom(s) du ou des tuteurs	Palard Isabelle
Acronyme de la certification IPI visée	CDAN
Niveau visé	Niveau II
Date de la soutenance	Mercredi 27 Novembre 2019
Lieu de la soutenance	EPSI/WIS BORDEAUX – 114 rue Lucien Faure – 33000 BORDEAUX



Remerciements

Je souhaite tout d'abord remercier Isabelle Palard, ma tutrice, de m'avoir recruté dans le projet et aidé dans mon intégration à l'équipe.

Pour des raisons de confidentialité sur mon projet, ne pouvant les citer personnellement, je souhaite également remercier tous les membres de l'équipe avec qui j'ai collaboré et qui se reconnaîtront, pour m'avoir aidé, formé et soutenu pendant toute l'année au sein de ce projet.

Je tiens à remercier également tous mes formateurs, David Gayerie, David Jacquot, Michel Gillet, Sylvain Labasse, Thomas Verdois et Bastien Gallay, pour leurs enseignements tant sur les méthodes que sur les technologies.

Mes remerciements s'étendent également à Isabelle Eugene, ma conseillère Pôle Emploi, sans qui je n'aurai jamais eu connaissance de cette formation et sans qui cette aventure n'aurait jamais eu lieu.

Enfin, j'exprime tous mes remerciements à toute ma famille, tout particulièrement à mes parents, pour m'avoir soutenu moralement et financièrement dans ma transition vers cette nouvelle vie.

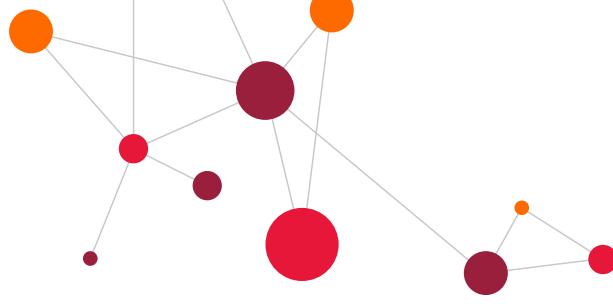
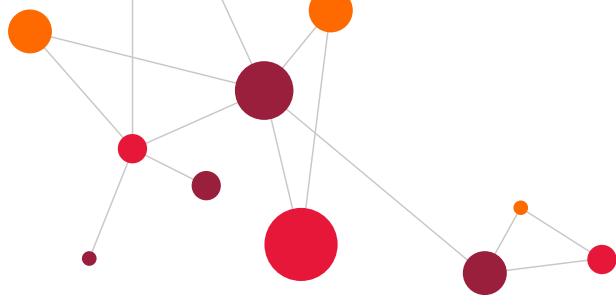
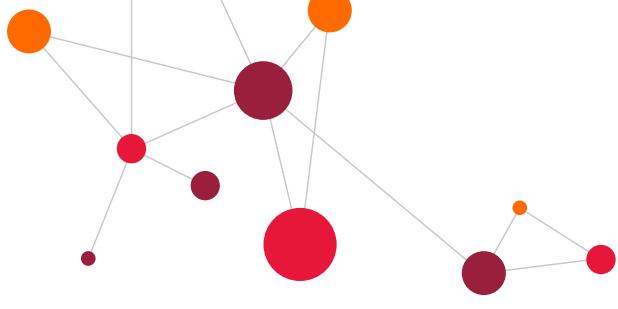


Table des matières

1	Introduction	3
2	Présentation de l'entreprise CGI	4
2.1	L'HISTOIRE DE CGI	4
2.1.1	ORGANISATION	4
2.1.2	CHRONOLOGIE DES POINTS CLÉS DE CGI	5
2.2	CGI EN CHIFFRES	5
2.3	LES CLIENTS DE CGI	6
2.4	CGI EN EUROPE ET EN FRANCE	7
2.5	BORDEAUX	8
2.5.1	FGDC	9
3	L'entreprise et moi	10
3.1	POSITIONNEMENT	10
3.1.1	LA BI	10
3.1.2	MON PROJET	11
3.2	LES OUTILS UTILISÉS	12
3.2.1	INFORMATICA	13
3.2.2	BUSINESS OBJECT (SAP BI 4.1)	15
3.2.3	NOTEBOOK++	18
3.2.4	SUPERPUTTY	18
3.2.5	GIT	18
3.3	FONCTIONNEMENT	19
3.3.1	LE CYCLE EN V	19
3.3.2	SCRUM	20
3.3.3	MON PROJET	21
4	Exemples de missions réalisées	24
4.1	INTÉGRATION D'UN NOUVEAU FLUX DANS LE DWH	24
4.1.1	CRÉATION DES TABLES DU DATAWAREHOUSE	24
4.1.2	MODIFICATION DES DIFFÉRENTS SCRIPTS AFIN DE PRENDRE EN COMPTE LES FLUX	25
4.1.3	CRÉATION DU MAPPING INFORMATICA ET SON WORKFLOW ASSOCIÉ	26
4.1.3.1	Le fichier domaine_XXXX.csv	31
4.1.3.2	Le fichier alim_XXXX.cfg	31
4.1.3.3	Le fichier Pre_trt_unitaire.ksh	32
4.1.4	TEST D'ALIMENTATION	33
4.1.5	CRÉATION DES INDICATEURS	34
5	Formation	36
5.1	L'ARCHITECTURE MVC	36



5.2	PROJET DE FORMATION GESTIONMATERIEL	37
5.2.1	API	38
5.2.2	ANDROID	38
5.2.3	CLIENT LÉGER	42
6	Conclusion	45
Annexe 1 : Referentiel		1
Annexe 2 : Extrait d'une spécification fonctionnelle de création de vue		1
Annexe 3 : Script de création d'indicateurs avec conventions de nommage		2
Annexe 4 : fiche de test unitaires avec tests de non régrésion		3
Annexe 5 : MODOP		6
Annexe 6 : Bon de livraison		9
Annexe 7 : Modification et versionning d'un indicateur		11
Annexe 8 : Outil GOJIRA		12
Annexe 9 : Script de conception de base de donnée		13
Annexe 10 : Schéma de Base de Donnée		16
Annexe 11 : Maquette Projet Android		17



1 Introduction

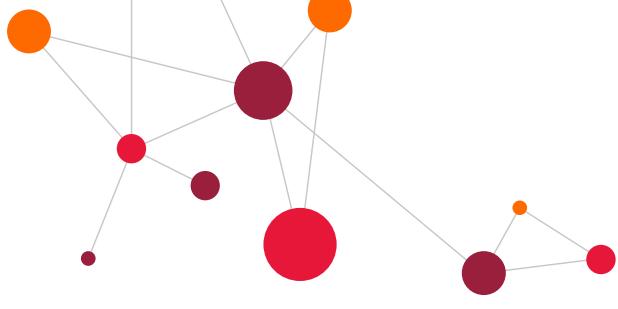
Un concepteur/développeur d'application numérique conçoit et développe des services numériques à destination des utilisateurs en respectant des normes et des standards professionnels tout en assurant la sécurité informatique à toutes les étapes. Il doit aussi s'adapter en continu sur les évolutions technologiques.

Il peut être autonome sur les projets de petite taille ou, comme dans mon cas, dans des projets plus importants, travailler au sein d'une équipe.

Le concepteur/développeur d'application numérique est amené à travailler sur différents langages informatiques. Dans mon entreprise, certains collègues peuvent travailler sur du COBOL, d'autres sur du JAVA, C++, C#, quant à moi, je travaille dans la BI, où je crée principalement des scripts SQL, manipule les ETL (Extract Transform Load) et modifie des scripts KSH (Korn Shell).

Ce mémoire est divisé en deux parties, la première partie présentera l'entreprise, mon projet et des tâches réalisées, puis la seconde partie se concentrera sur les compétences acquises durant ma formation à l'EPSI avec un projet d'étude.

Enfin, nous exposerons ma vision du métier de Concepteur/développeur d'applications numérique.



2 Présentation de l'entreprise CGI

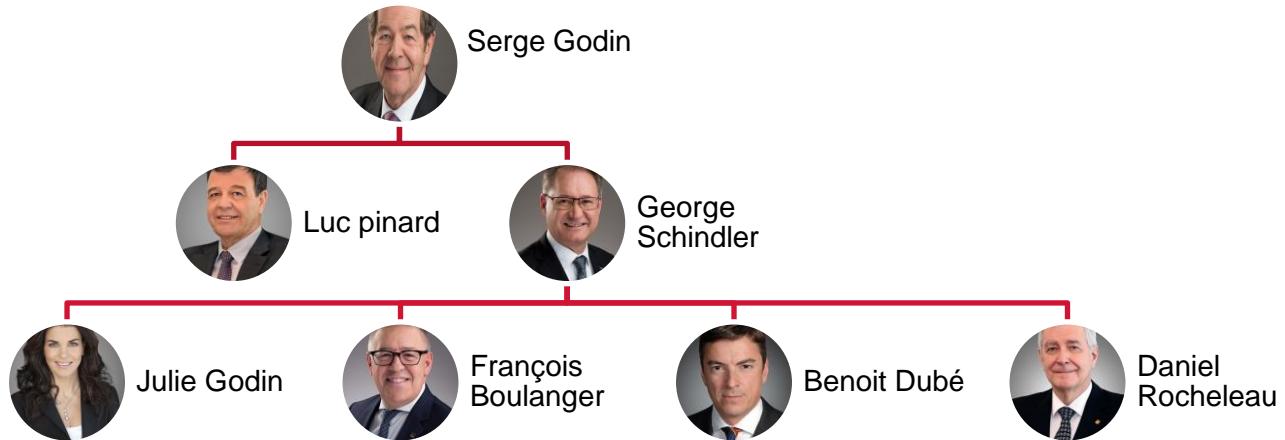
2.1 L'histoire de CGI

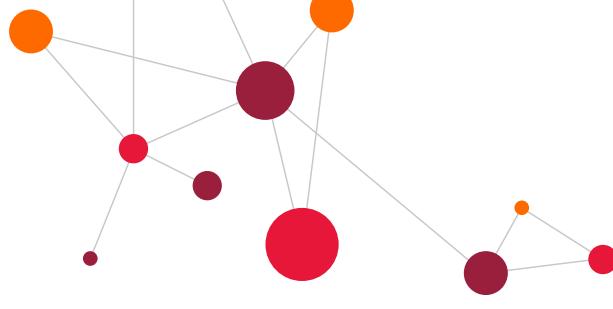
L'entreprise CGI, « Conseillers en Gestion et Informatique » ou « Consultants to Government and Industry » en anglais, est une ESN (Entreprise de Services du Numérique) canadienne fondée en 1976 par Serge Godin et André Imbeau.

Cette entreprise est spécialisée dans :

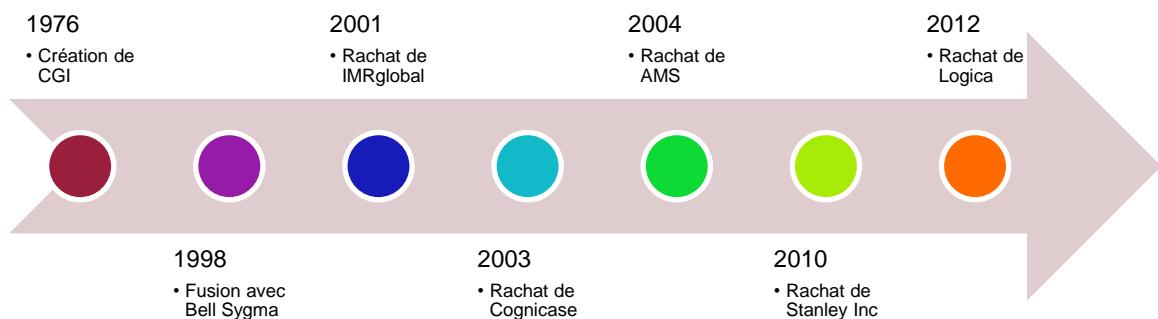
- Les services, conseils en technologie de l'information.
- L'intégration de systèmes
- Recherche de solutions pour les entreprises.

2.1.1 Organisation





2.1.2 Chronologie des Points clés de CGI



2.2 CGI en Chiffres

CGI est en constante évolution depuis sa création :

- Évolution du nombre de salariés depuis la création



- Chiffres clé

Fondée en 1976

43 ans de croissance rentable

11,5 milliards \$CA
de revenus

77 500
consultants

40 pays
400 emplacements

5 000 clients à l'échelle mondiale
font appel à nos services complets

Plus de **175** solutions de propriété
intellectuelle desservant 30 000 clients



2.3 Les clients de CGI

CGI a des accords avec plus de 5000 clients à travers le monde dans différents secteurs d'activité tels que décrits ci-dessous :

SECTEUR BANCAIRE	COMMUNICATIONS	GOUVERNEMENTS	SANTÉ ET SCIENCES DE LA VIE	ASSURANCE

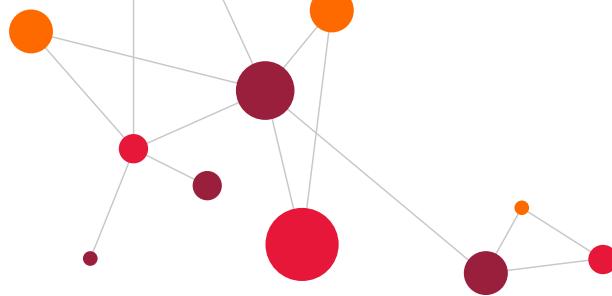
➤ Collaboration avec **15 des 20 principales banques** à l'échelle mondiale
 ➤ **Secteurs** : banques de détail, marchés des capitaux, services aux entreprises et transactions bancaires

➤ Collaboration avec **6 des plus importants fournisseurs de services de communication au monde**
 ➤ **Gamme de services** : transformation de la facturation, de l'orchestration des commandes, de l'assurance des revenus et du service à la clientèle

➤ Collaboration avec plus de **2 000 clients gouvernementaux** dans **15 pays**
 ➤ **Secteurs** : gouvernements provinciaux, d'États, locaux, centraux et fédéraux ainsi que défense, renseignement et industrie spatiale

➤ **Plus de 1 000 établissements de santé**; régimes d'assurance-maladie offrant des services à 195 millions de bénéficiaires; **plus de 50 sociétés** des sciences de la vie
 ➤ **Secteurs** : assureurs, fournisseurs, entreprises pharmaceutiques et biotechnologiques, organismes de réglementation

➤ Collaboration avec **7 des 10 principales compagnies d'assurance** mondiales
 ➤ **Secteurs** : assurance de dommages et assurance vie

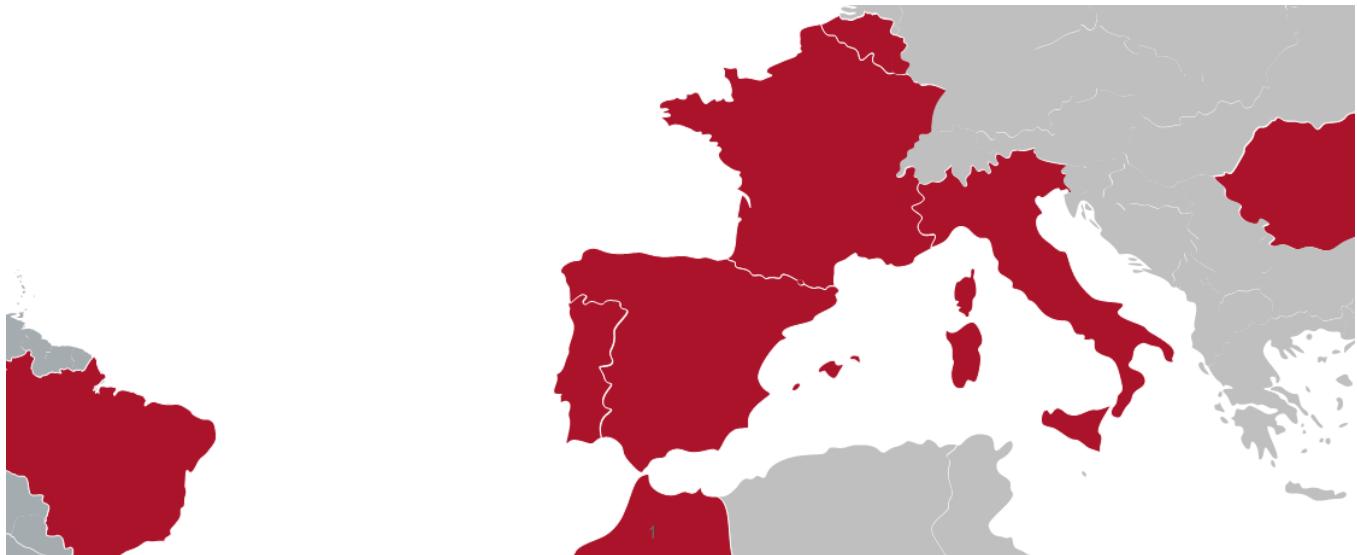


SECTEUR MANUFACTURIER	PÉTROLE ET GAZ	COMMERCE DE DÉTAIL ET SERVICES AUX CONSOMMATEURS	TRANSPORT ET LOGISTIQUE	SERVICES PUBLICS			
MICHELIN AIRBUS CAE TOYOTA BMW Air Liquide 	RioTinto Chevron PETRO-CANADA RENAULT DAIMLER nordseefonden	ExxonMobil encana equinor TOTAL REPSOL NESTE	L'ORÉAL SNC-LAVALIN Taylor Wimpey Couche-Tard GROUPE CLARINS metro EMPIRE COMPANY LIMITED	Carrefour Walmart Canada Auchan VR GROUP ProRail Rijkswaterstaat Ministry of Infrastructure and Water Management	DHL Lufthansa VIA Rail Canada Queensland Rail DB TRAFIKVERKET	EDSN ENGIE ENERGI NEST NORTHERN POWERGRID ELEXON FINGRID allianz	Welsh Water ENEDIS NORTHERN POWERGRID

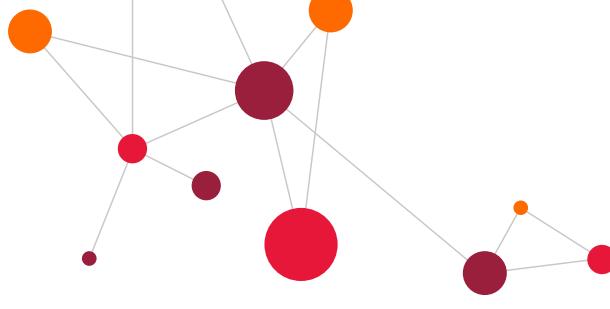
> Plus de 700 clients du secteur manufacturier
 > Secteurs : ressources naturelles, biens commerciaux et industriels, industrie automobile
 > Collaboration avec les principales entreprises pétrolières et gazières à l'échelle mondiale
 > Services en production, raffinage, approvisionnement et distribution; commerce électronique interentreprises (B2B) et de l'entreprise au consommateur (B2C)
 > Plus de 800 clients à l'échelle mondiale
 > Secteurs : commerce de détail, vente en gros, biens de consommation courante et services aux consommateurs
 > Plus de 200 clients à l'échelle mondiale
 > Secteurs : services postaux et logistique, transport aérien, ferroviaire, maritime et routier ainsi que transit régional
 > Plus de 450 clients des secteurs de l'électricité, de l'eau et du gaz
 > Secteurs : vente au détail, exploitants de réseaux et autres

2.4

CGI en Europe et en France



La SBU (L'unité d'affaire stratégique) Ouest et sud de l'Europe est composée de la Belgique, du Brésil, de l'Espagne, de la France, de l'Italie, du Luxembourg, du Maroc, du Portugal et de la Roumanie



Les 15000 membres de la SBU interviennent dans les principaux secteurs d'activités, à savoir le conseil, l'intégration et le outsourcing.

En France, CGI collabore avec toutes les entreprises du CAC40 qui ne sont pas des ESN, avec un score de satisfaction client de 9,10 /10.

CGI France possède une multitude d'entités.

Nous allons nous intéresser directement à notre agence de Bordeaux :

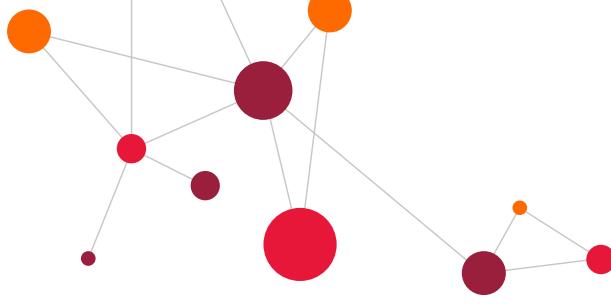
2.5 **Bordeaux**



CGI possède 3 entités à Bordeaux :

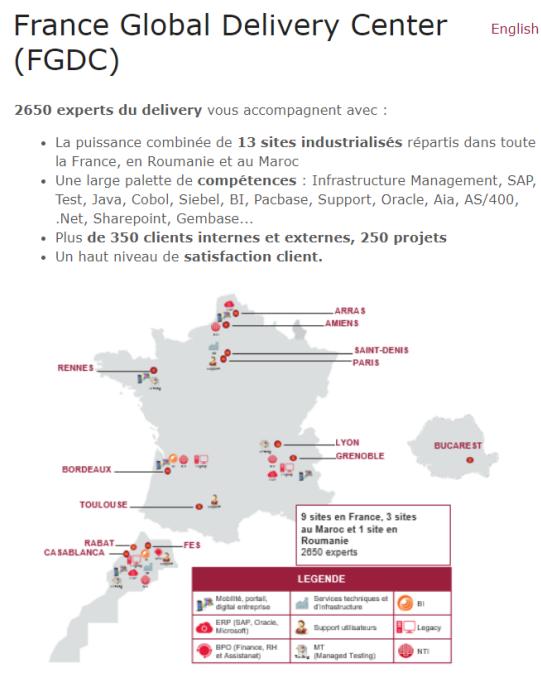
- Grand Ouest
- TPS HR
- FGDC.

Nous allons nous intéresser à mon entité, FGDC Bordeaux



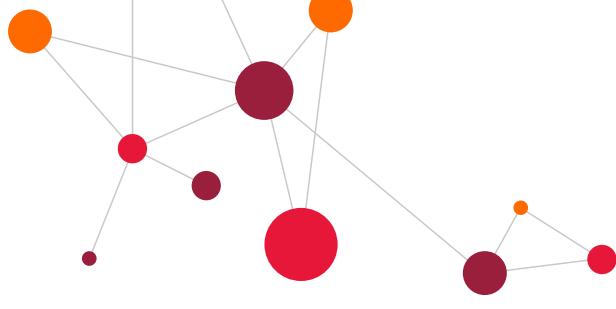
2.5.1 FGDC

FGDC (France Global Delivery Center), Gestion de projet global de service, concerne une multitude de secteurs. Il possède un service industriel répondant aux spécificités de chaque clients grâce à des équipes spécialisées dans chaque centres :



Organisation GDC Bordeaux





3 L'entreprise et moi

3.1 Positionnement

3.1.1 La BI

L'informatique décisionnelle (Business intelligence en anglais) est un ensemble d'outils et de méthodes ayant pour but de transmettre les informations pertinentes aux managers d'entreprises.

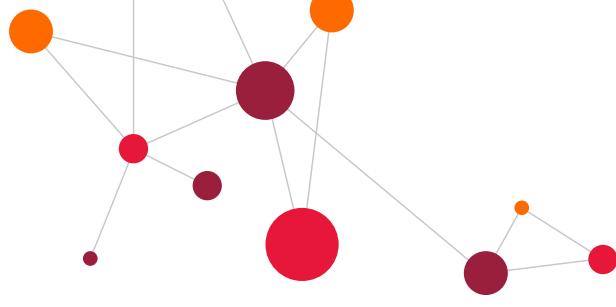
Elle permet en outre d'améliorer et d'optimiser les décisions et les performances. Son objectif est de les aider à comprendre leur environnement ainsi que de les aider à prendre les bonnes décisions stratégiques. Comprendre également les besoins de l'utilisateur est donc primordial pour une bonne entente.

Pour cela, il faut créer notre système décisionnel :

Principe général simple d'un système décisionnel :

Collecter Intégrer Organiser Restituer

- La collecte : elle permet d'alimenter notre système décisionnel (SID). La technique employée consiste à extraire les données et filtrer les informations pertinentes pour les analyses.
- L'intégration permet de préserver la cohérence des données et de les stocker dans un entrepôt. Pour cela il est nécessaire de nettoyer ces données, de les convertir si besoin (changer le type) et de les rassembler.
- L'organisation, comme son nom l'indique, est le fait de réorganiser les données pour les adapter aux applications décisionnelles. Il faut donc agréger et structurer les données afin d'améliorer les temps de réponses et optimiser le SID.
- La restitution : elle permet aux décideurs de prendre les bonnes décisions. Il faut pour cela se servir d'un outil de reporting comme SAP BI 4 par exemple.



Les étapes de la conception d'un SID :

1. Identifier les besoins
2. Analyser la source
3. Identifier les données nécessaires
4. Définir le DTM et le DWH
5. Alimenter le DWH
6. Alimenter le DTM

3.1.2 Mon projet

Je suis Analyste développeur Junior chez CGI dans un projet où je suis prestataire pour un grand acteur dans le domaine de l'Energie.

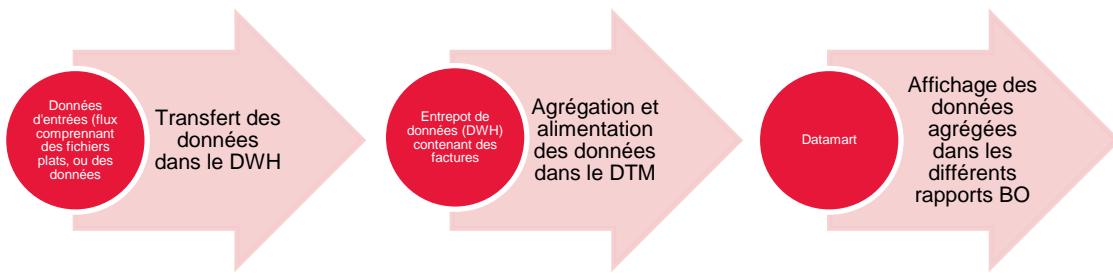
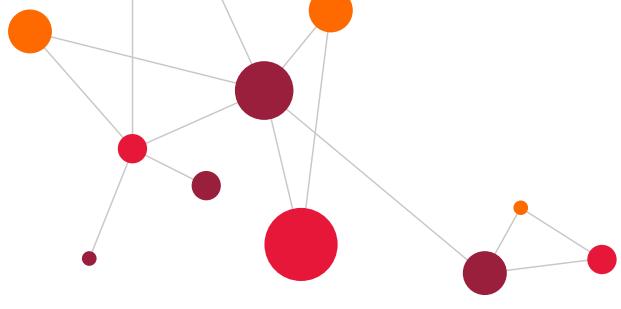
Par soucis de confidentialité, je ne suis pas en mesure de divulguer les informations sensibles tels que les données, le nom des collègues, des clients ou des applications.

Un projet pouvant avoir plusieurs applications, mon équipe et moi collaborons sur une de ces applications. Celle-ci est un outil décisionnel permettant d'industrialiser la production des publications, de garantir la qualité des données, et également d'alimenter d'autres applications.

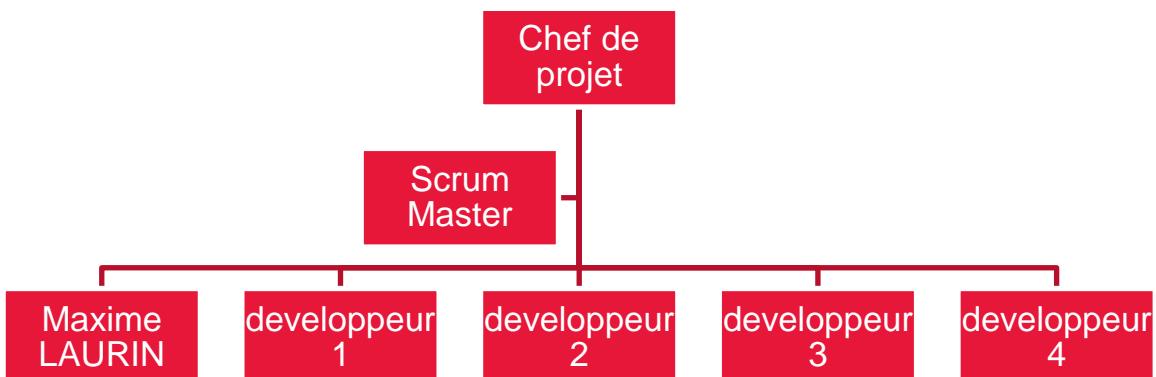
Les principales publications de cette application sont :

- Le marché de masse
- Le marché d'affaires
- Moteur de recherche PEC
- Reporting finance
- Divers reporting

Voici un schéma explicatif simplifié :

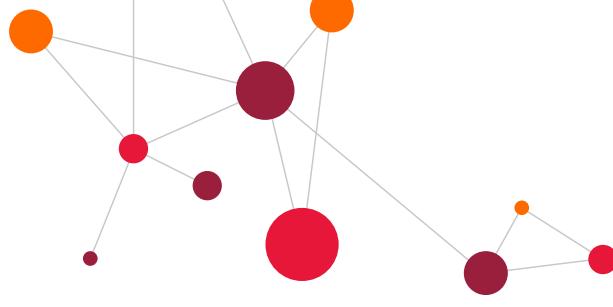


Organigramme de ma partie de projet



3.2

Les outils utilisés

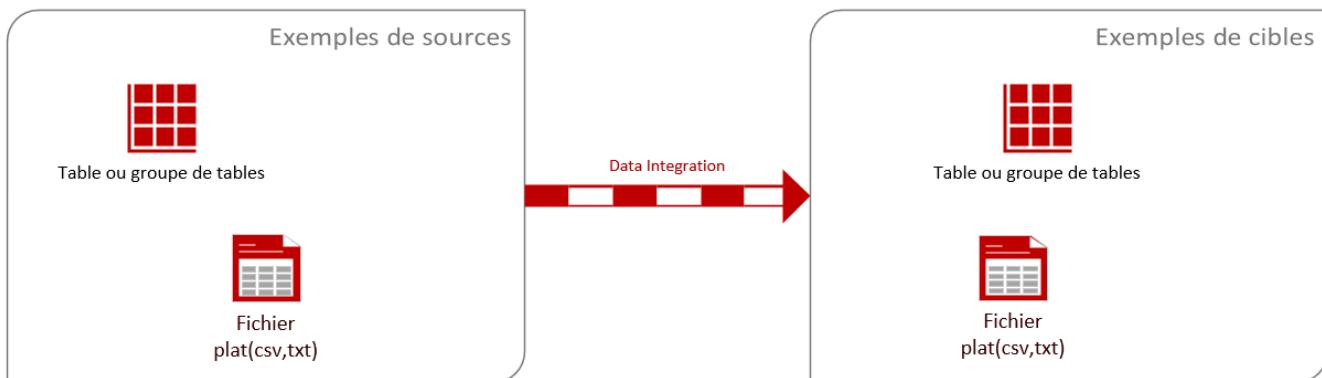


Pour mener à bien mes missions, j'ai à disposition plusieurs logiciels :

3.2.1 **Informatica**

Informatica est un ETL (Extract-Transform-Load) permettant, comme son nom l'indique, d'extraire des données, de les transformer, puis de les charger.

Grace à cet outil, nous pouvons par exemple extraire les données d'un fichier ou d'une table, et les charger dans une autre table ou fichier.



Son point fort est sa performance, en effet, il est capable de charger des millions de données avec rapidité.

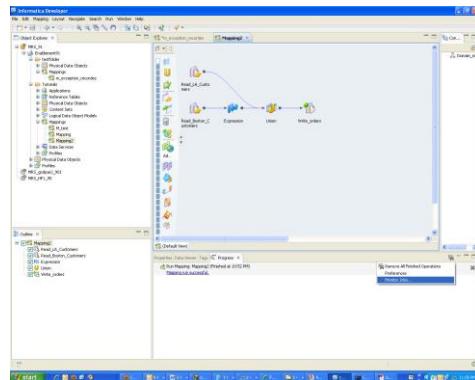
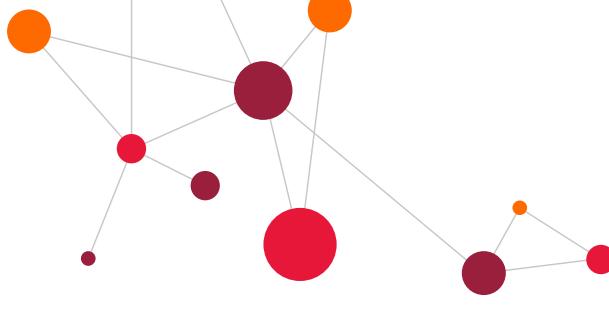
Cet ETL est principalement utilisé pour :

- Intégrer des données dans un système d'informations (Data Warehouse, Datamart)
- Interfacer des systèmes hétérogènes
- Migrer des données

La suite Informatica est composée de plusieurs outils :

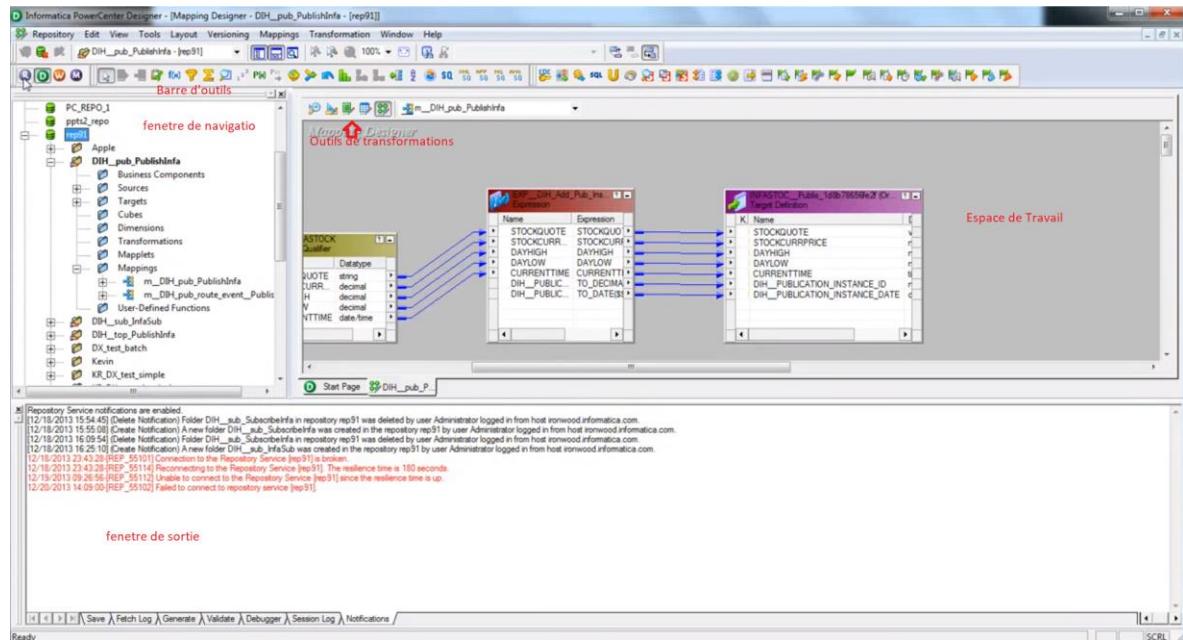
- **Informatica Développer**

L'outil développer Tool est une application permettant d'implémenter des solutions Big Data, d'intégrer des données comme des Web Services, des qualités, profilages et services de données.



- **PowerCenter Designer**

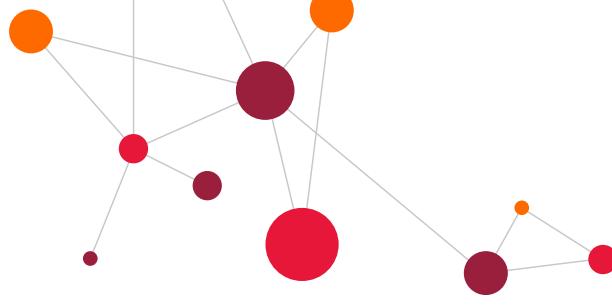
PowerCenter designer fournit des outils pour créer et modifier des mapping, sources, cibles et des transformations



- **PowerCenter Repository Manager**

PowerCenter repository Manager est un outil pour ajouter de nouveaux domaines et est utilisé pour organiser des métadonnées stockées dans le repository. On peut notamment :

- Gérer les autorisations des utilisateurs et des groupes



- Éditer, copier, créer ou supprimer des dossiers
- Afficher les métadonnées.
- PowerCenter Workflow Manager

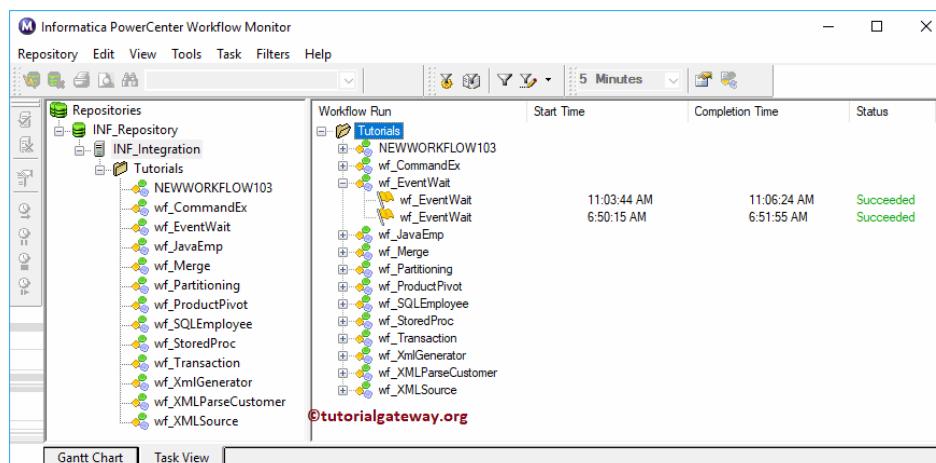
PowerCenter Workflow Manager est un outil utilisé pour créer un workflow.

Un workflow est un ensemble d'instructions pour exécuter les mappings précédemment conçus dans le PowerCenter Designer.

On peut par exemple choisir l'ordre des mappings, donner des instructions complémentaires comme modifier la source d'un mapping ou encore surcharger des requêtes SQL.

- PowerCenter Workflow monitor

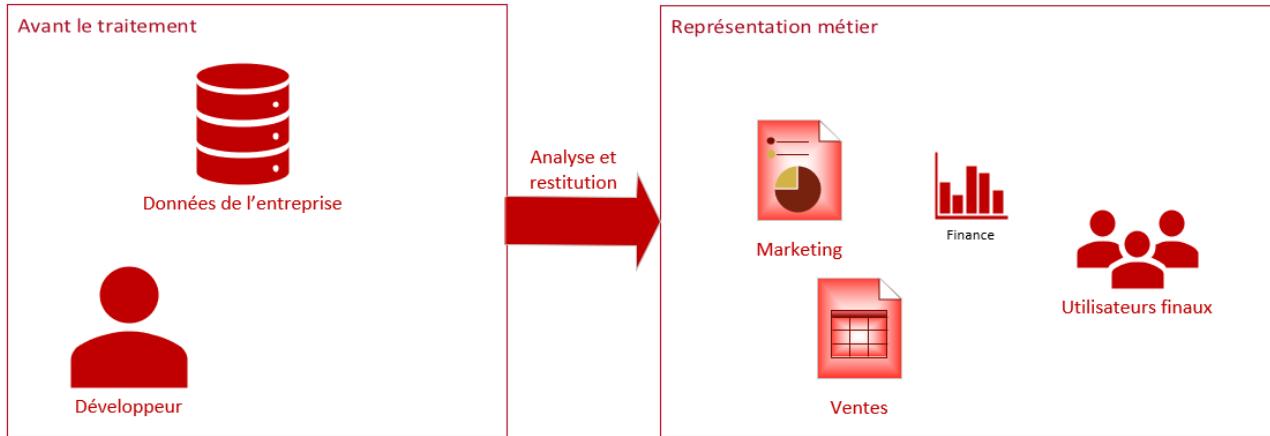
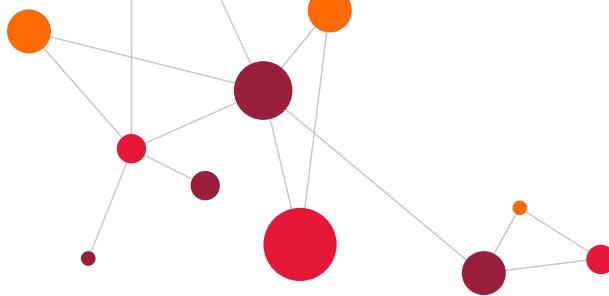
PowerCenter Workflow Monitor est l'outil indispensable du développeur. Il sert à afficher les logs d'une session, permet de connaître la source d'un plantage, voir le nombre de lignes insérées ou encore de voir l'ordre dans lequel les mappings se sont exécutés en cas de multiples mappings dans un workflow.



3.2.2 Business Object (SAP BI 4.1)

La suite SAP BI 4 est un outil de reporting très utilisé dans la BI.

Elle permet d'offrir aux utilisateurs un accès aux informations décisionnelles partout sous forme de tableaux, graphiques etc...



Cette suite est composée de plusieurs outils :

- Outil de conception d'univers :

Cet outil nous permet de créer des univers afin de pouvoir restituer des données dans le Web Intelligence Rich Client. Il est notamment utilisé afin de fournir une interface intuitive et compréhensible permettant à des utilisateurs non spécialistes de Web intelligence d'exécuter des requêtes afin de créer des rapports, et d'effectuer de l'analyse de données.

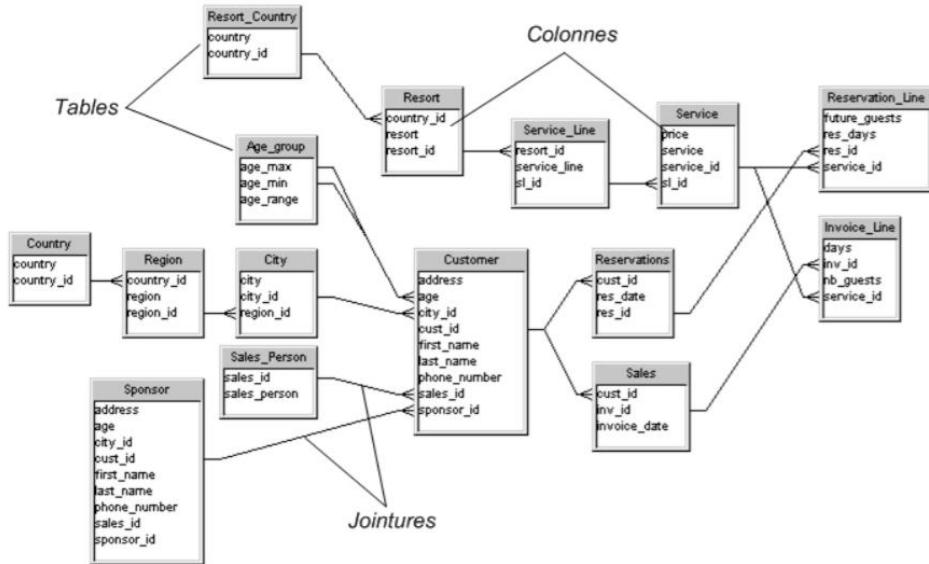
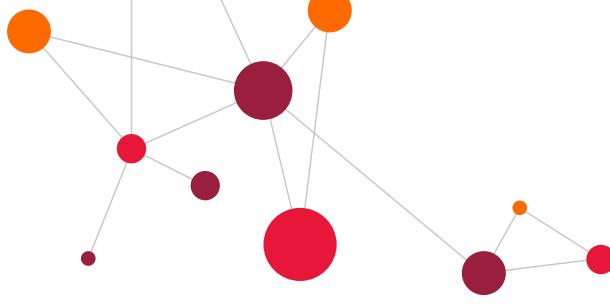
Le rapport doit correspondre à l'environnement professionnel des utilisateurs finaux :

Il faut donc utiliser un vocabulaire fonctionnel et éviter le vocabulaire technique afin de présenter une interface professionnelle aux structures SQL de base de données.

Un univers est un simple fichier, qui contient toutes sortes de paramètres et d'éléments :

- Des paramètres de connexion pour un ou plusieurs middlewares de base de données
- Des structures SQL (Objets) qui correspondent à de vraies structures SQL de la base de données (Colonnes, tables, fonctions) et sont regroupées en classe. Les utilisateurs pourront voir ces objets.
- Un Schéma des tables et des jointures utilisées dans la base de données.

Voici un exemple d'univers :



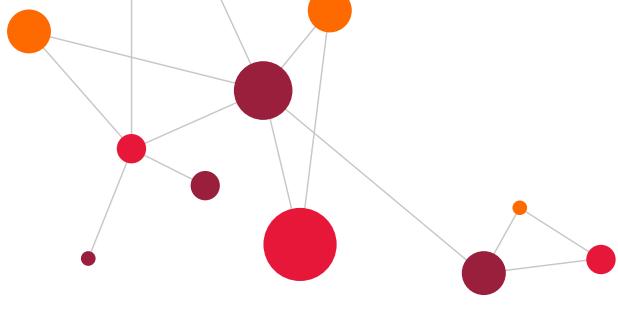
- Web Intelligence Rich Client :

Web Intelligence permet de créer des rapports reposant sur les données à analyser. Il se base sur l'univers expliqué précédemment et permet de créer des graphiques, tableaux afin de restituer les données dont aura besoin l'utilisateur.

Aperçu de l'interface de Web Intelligence Rich Client

Account	Amount
Accounts Payable	46,255,420.57
Additional Paid In Capital	182,652,497.27
Allowance for Bad Debt	3,229,223.89
Amortization of Goodwill	135,909.43
Building Leasehold	727,156.74
Buildings & Improvements	10,661,199.24
Cash	146,693,601.71
Commissions	3,976,603.27
Conferences	68,133.89
Construction In Progress	1,184,020.07
Curr Xchg Gain/(Loss)	86,484

- Webi



Webi est le client léger de Web Intelligence Rich Client et permet de créer les rapports directement depuis notre navigateur.

3.2.3 Notepad++

Notepad++ est l'outil indispensable pour éditer des fichiers de configurations ou créer des scripts SQL sans avoir besoin de passer sur Oracle SQL developer, sauf pour exécuter les requêtes.

3.2.4 SuperPuTTY

SuperPuTTY est un client SSH. Contrairement à PuTTY, celui-ci propose une interface pertinente pour gagner en rapidité en proposant l'utilisation de plusieurs fenêtres afin d'améliorer l'utilisation des multiples cessions.

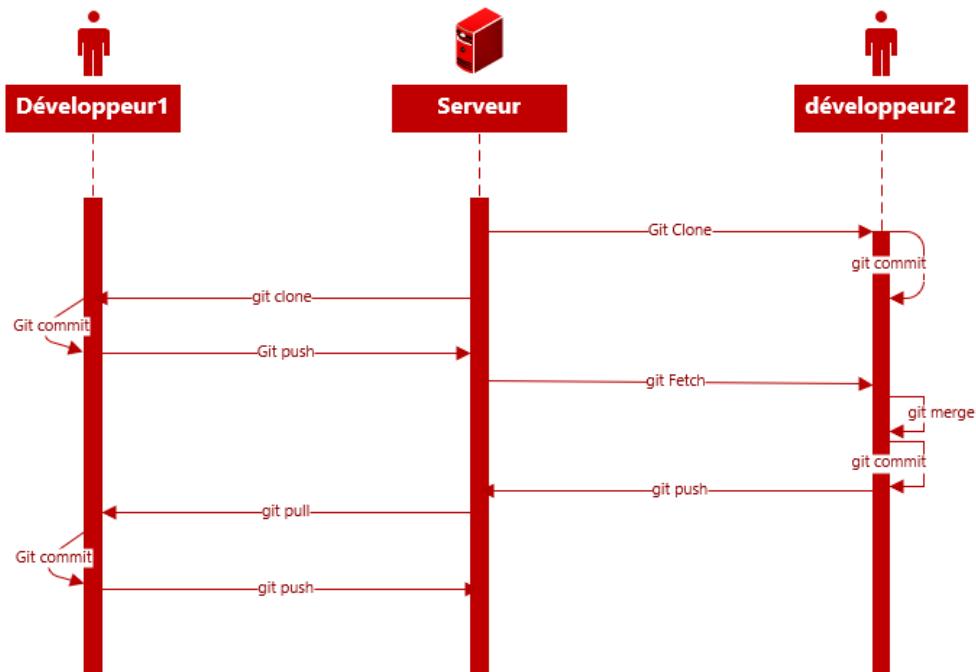
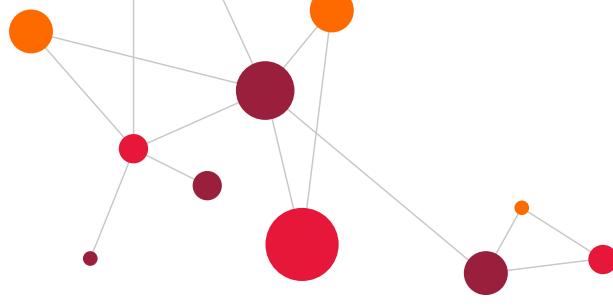
Un système de fenêtrage est aussi intégré afin de pouvoir mettre jusqu'à 4 fenêtres simultanées sur un écran.

3.2.5 Git

Git est un logiciel de gestion de version décentralisé.

Il permet de travailler tout en gardant une trace des modifications apportées successivement, et peut retrouver l'état antérieur des données en cas d'erreurs sur une version.

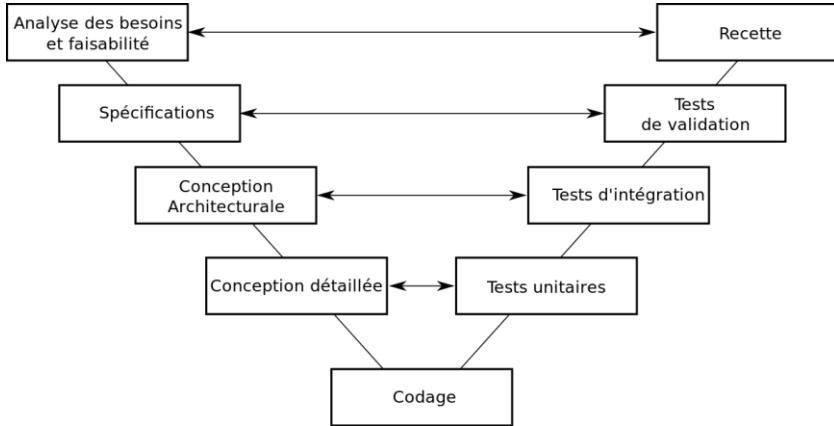
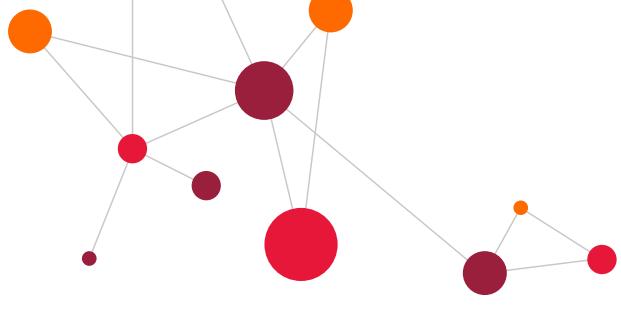
Diagramme de séquence git



3.3 Fonctionnement

3.3.1 Le cycle en V

Le cycle en V était une méthode de gestion de Projet très populaire dans les années 80/90.



Le principal but du « cycle en V » est de réaliser les différentes étapes les unes après les autres tout en limitant au maximum les retours arrière. Si un retour arrière demande à recommencer le cycle, les modifications sont souvent réalisées en cours du cycle malgré le manque d'encadrement dû à cette méthodologie.

Les points forts de cette méthode sont :

- La préparation en amont des tâches de spécifications
- Les rôles bien définis
- Les tests adaptés aux besoins initiaux

Les points faibles, en revanche, sont

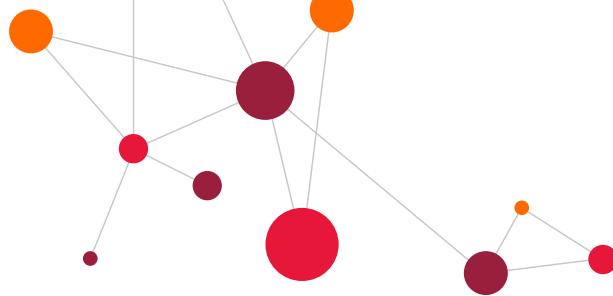
- Le manque de souplesse
- Le manque de flexibilité pour l'entreprise.

3.3.2 Scrum

Scrum, ou Mêlée, est une méthode de développement permettant de concevoir des logiciels en impliquant au maximum leurs clients.

Elle est articulée autour de trois acteurs :

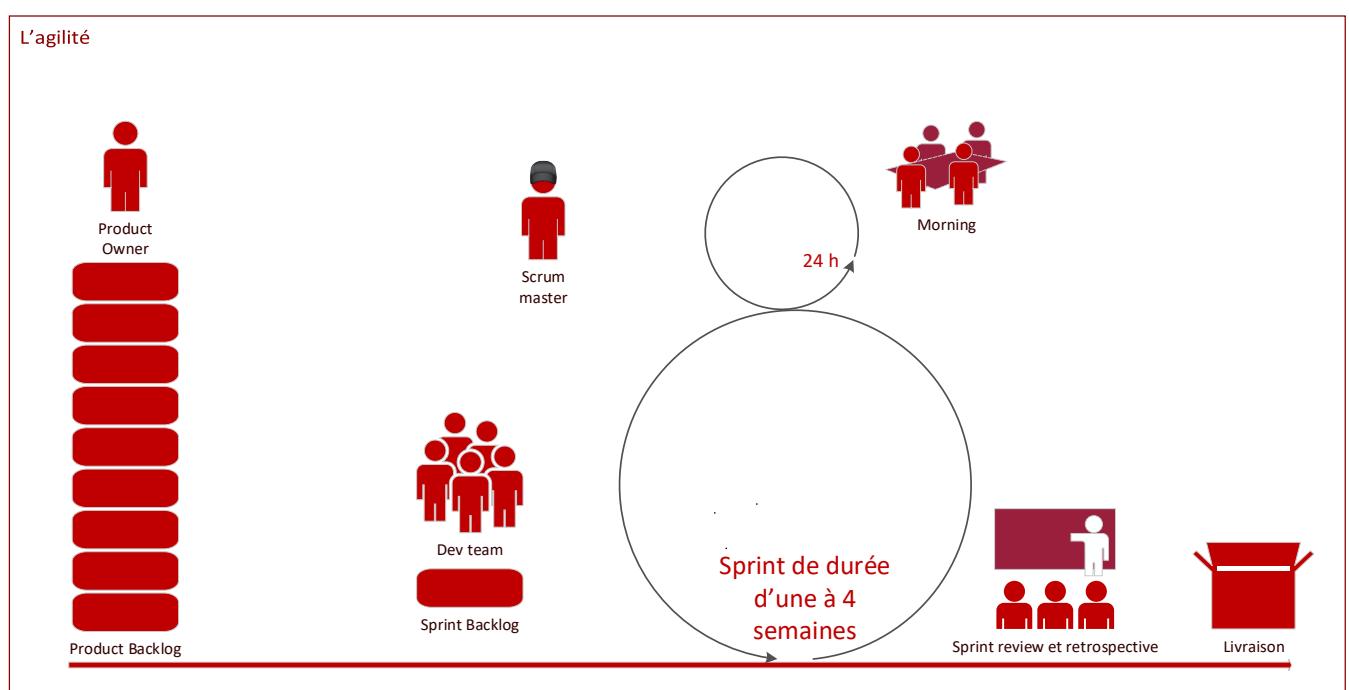
- Le Product Owner, qui a une vision complète du produit
- Le Scrum master qui agit comme un coach, s'assure du bon déroulement du projet
- L'équipe scrum, dont je fais partie, qui développe l'application



Le Product Owner liste des tâches à réaliser qui a pour vocation d'évoluer au fur et à mesure que le produit se construit. Lors du commencement du « Sprint », l'équipe doit se documenter et noter les différentes US (User story) afin de déterminer le niveau de charge de celles-ci. Vient ensuite la répartition des tâches où l'équipe agile peut choisir ses US en fonction de leurs compétences.

Chaque jour à lieu un « Morning ». Le Morning est une petite réunion où chaque membre de l'équipe prend 5 minutes pour expliquer sans entrer dans le détail ce qu'il a fait, ce qu'il va faire et s'il y a des points bloquants.

A la fin du « Sprint », le client fait une présentation au maximum fonctionnelle afin que le Product Owner valide ou non les modifications réalisées afin que l'US soit livré.

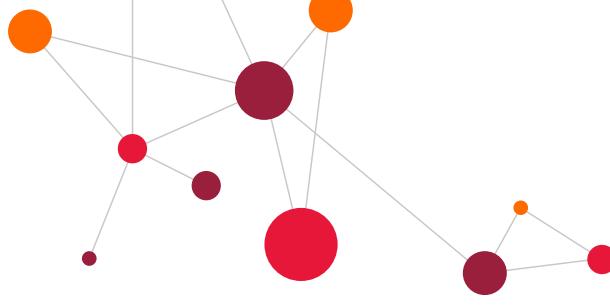


On peut aussi noter que cette méthode fait appel à des outils visuels comme des tableaux, et des cartes pour noter les US.

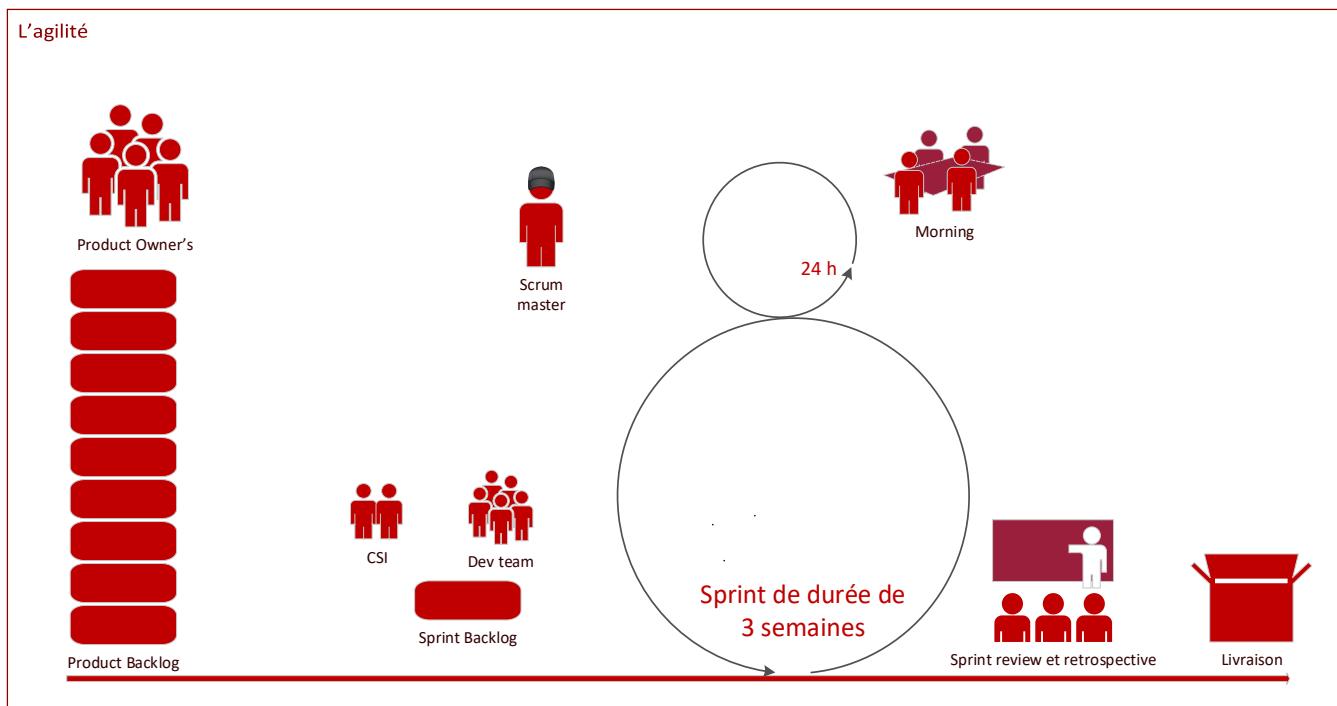
3.3.3 Mon projet

Mon projet a choisi d'utiliser la méthode Scrum après des années en « cycle en V ».

Il est intéressant d'en parler, car il est en pleine transition entre le « cycle en V » et le « scrum ».



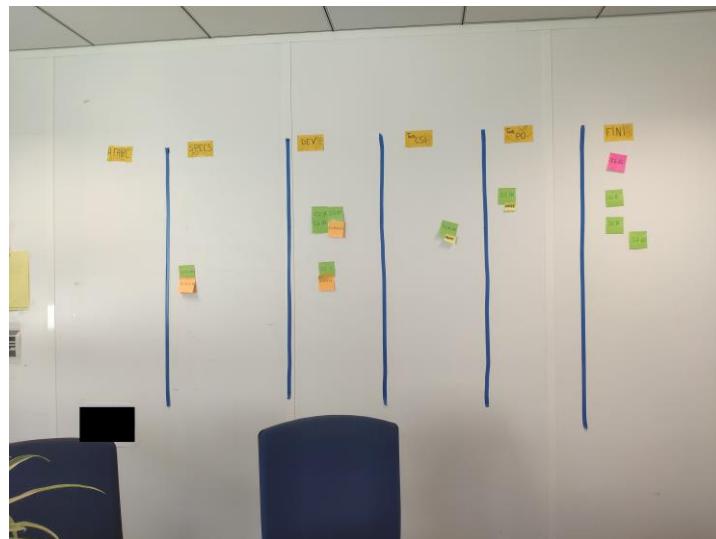
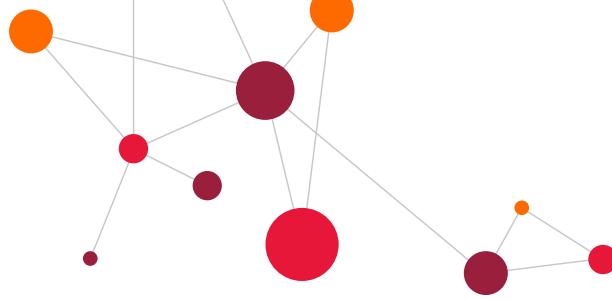
L'agilité



Comme l'illustre le schéma ci-dessus, et comme je l'ai vu durant mon année à CGI, j'ai pu observer l'évolution de cette méthode jusqu'à aujourd'hui :

- Plusieurs Product Owner, dont chacun a une vision spécifique à un domaine (restitution, intégration etc.)
- La présence de fonctionnels, comme dans le « cycle en V », s'occupent de rédiger les spécifications et de recetter nos US.
- Il est arrivé dans les débuts que le chef de projet assigne des tâches à un développeur, or, aujourd'hui, le chef de projet évoque les US et les développeurs peuvent maintenant choisir qui peut les faire .

Voici le ScrumBoard de notre « dev team », ce ScrumBoard nous sert à suivre l'avancement de nos US :



Les cartes du « planning poker » :

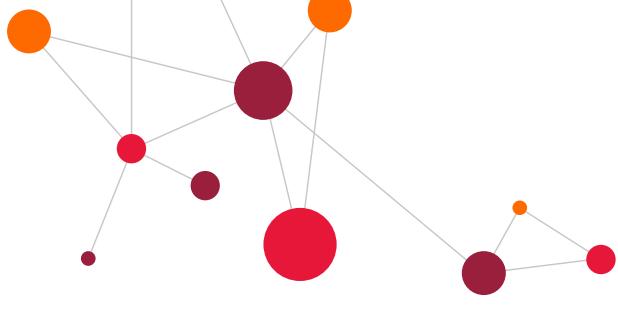


Chaque semaine, je participe à une réunion appelée planning Poker, elle a pour but de noter les différentes US du backlog pour le prochain sprint.

Les cartes se basent sur la suite de Fibonacci afin d'illustrer la complexité de la tâche et l'incertitude de l'estimer. Plus la complexité est grande et plus l'incertitude l'est aussi.

Chaque membre de la dev team montre la carte, s'il y a des estimations différentes, chacun expose son point de vue afin de clarifier des choses mal comprises. Une fois les avis exprimés, les membres reprennent une carte. Si au bout de plusieurs fois les membres ne se mettent pas d'accord, la majorité l'emporte.

L'outil GOJIRA est utilisé pour afficher les US, s'imputer sur les tâches et informer l'équipe en temps réel des avancées.



4 Exemples de missions réalisées

4.1 Intégration d'un nouveau flux dans le DWH

J'ai choisi cette mission car c'est l'une des plus polyvalentes que j'ai pu faire parmi toutes mes missions réalisées.

Ma mission consiste à intégrer un nouveau flux dans le Data Warehouse :

Les outils utilisés :

- Informatica
- SuperPuTTy
- Oracle SQL Developper
- Notepad ++

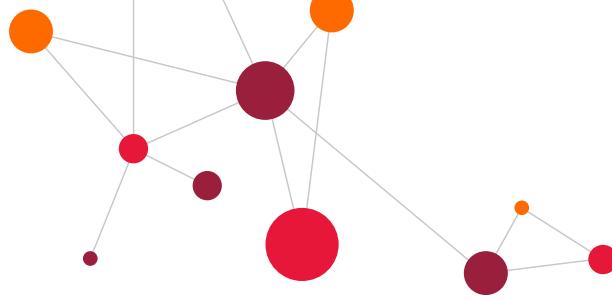
Il faut au préalable commencer la FTU (Fiche de Tests Unitaire) afin de déterminer les tests et les impacts que notre travail va engendrer. Elle nous sert également à garder le cap en cas d'erreur d'interprétation.

Nous allons suivre pas à pas le déroulement de l'US.

4.1.1 Crédation des tables du DataWareHouse

Pour la création des tables du DWH, il y a plusieurs paramètres à prendre en compte :

- Il doit être ré exécutable :
Le script va être réutilisé plusieurs fois en production, il faut donc penser à tout faire pour qu'il ne plante pas. Dans notre cas, il faut faire une commande de suppression de table avant la création pour être sûr que celle-ci ne plante pas.
- Elle doit être partitionnée par période.
- Elle doit être dans un tablespace afin d'avoir du stockage alloué.
- Les commandes ne doivent pas comporter d'espace (le script plantera à l'exécution dans l'environnement de production)



- Le script doit comporter un en-tête afin de savoir ce que fait le script, qui l'a fait, pour quelle US, quand elle a été réalisée.

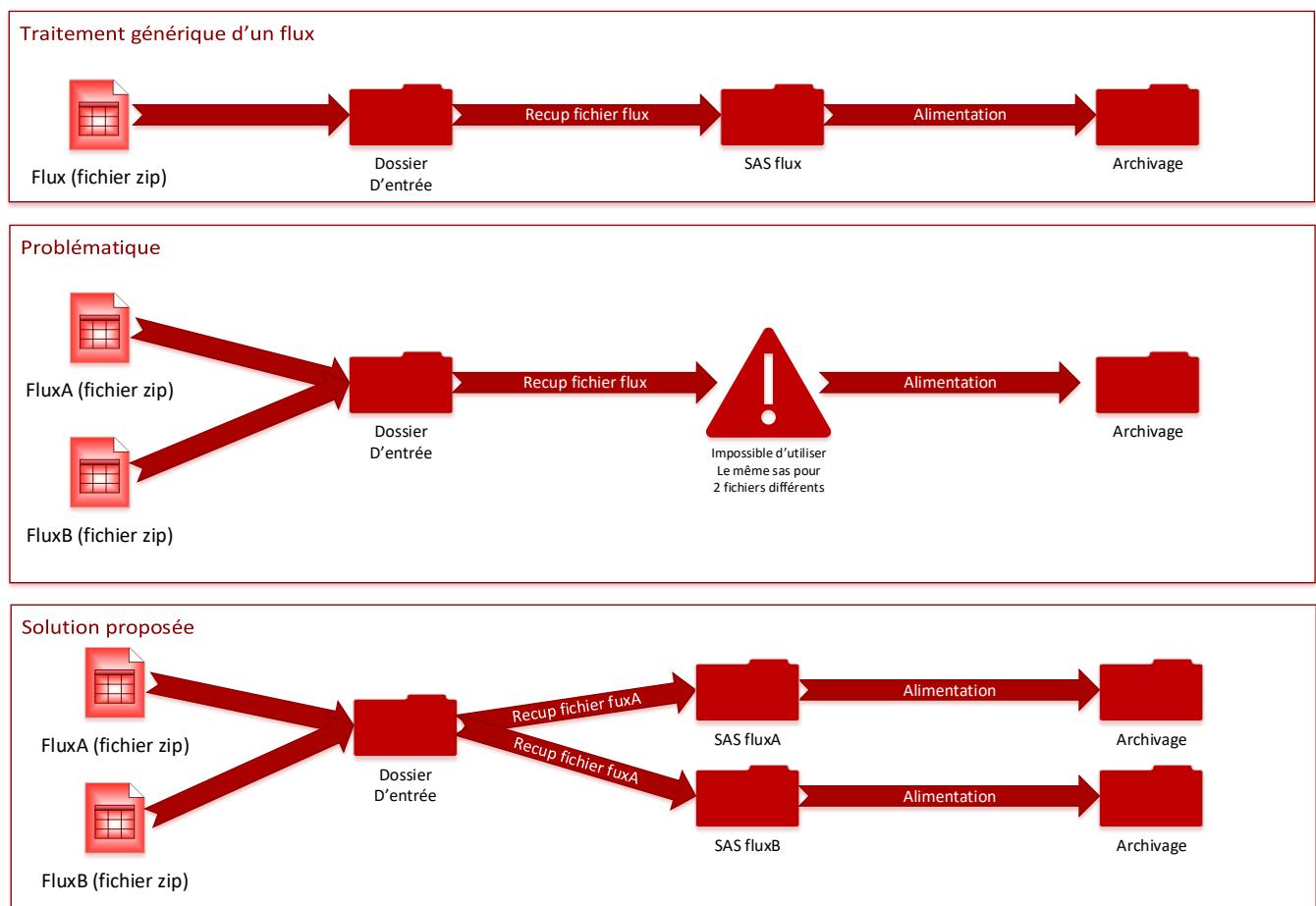
4.1.2 Modification des différents scripts afin de prendre en compte les flux

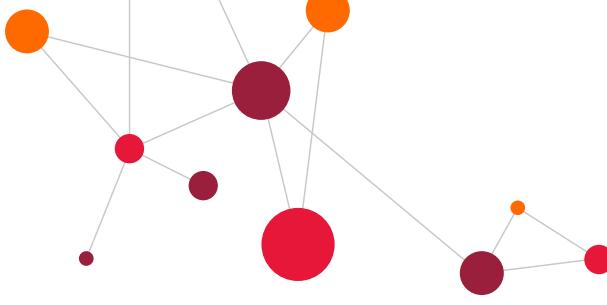
Pour intégrer un flux, avant toute chose, il faut créer les différents répertoires où les dossiers seront déposés.

- Problématique :

Les scripts sont conçus pour intégrer un seul type de fichier par dossier, soit un seul par flux,

Voici un schéma explicatif :





La solution proposée est donc d'adapter le script du « récup fichier » séparant les fichiers dans un sas, pour cela, il faut :

- Analyser le nom du fichier
- Dézipper le fichier et le copier dans le bon répertoire suivant le nom.
- Supprimer le fichier du répertoire d'entrée

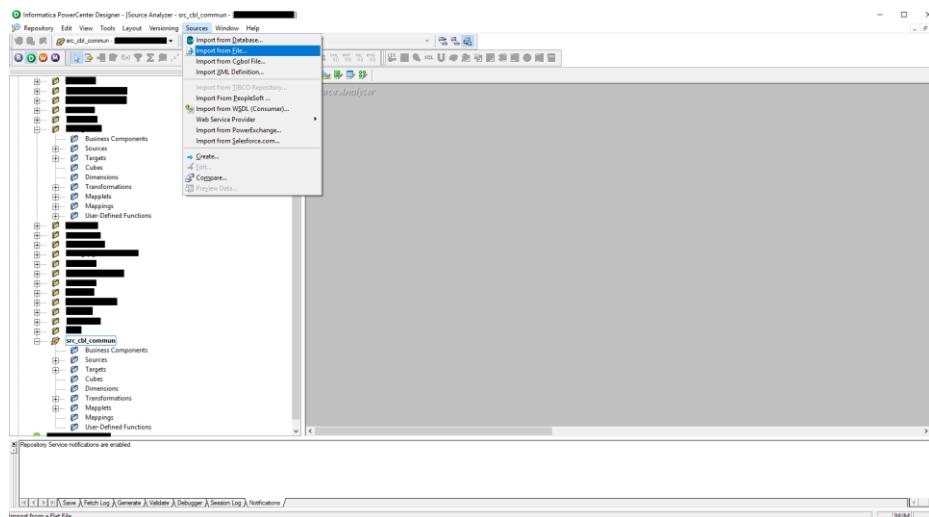
4.1.3 Crédit du mapping Informatica et son Workflow associé

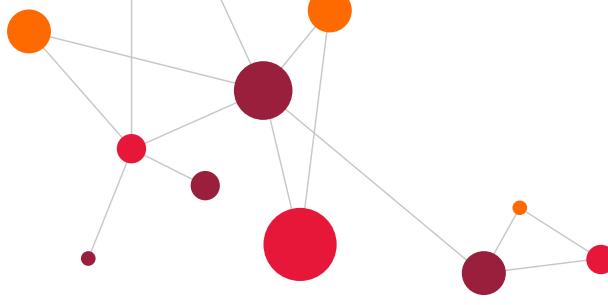
Pour créer :

Tout d'abord, par soucis de réutilisation du code, je dois impérativement créer les sources et cibles dans le répertoire commun.

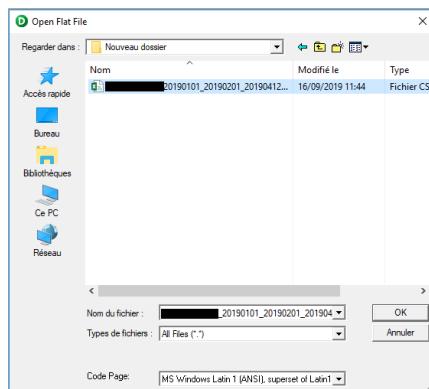
Voici les étapes de création d'une nouvelle cible :

- Tout d'abord, dans Informatica PowerCenter designer, sélectionner l'option « Import from File »

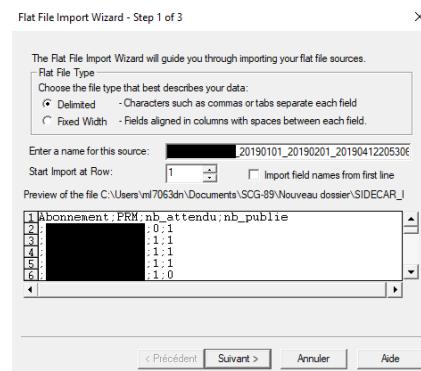




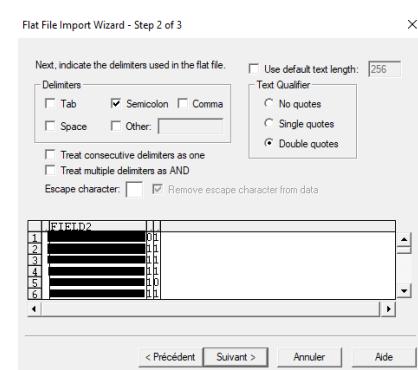
- Sélectionner le fichier qui sera le modèle de notre source

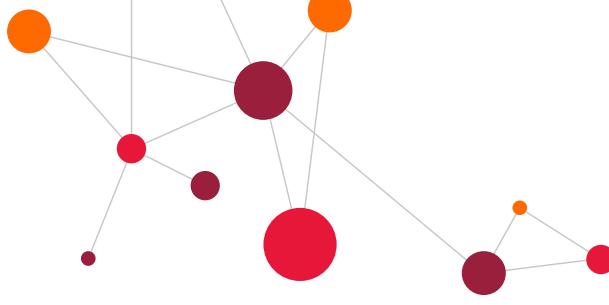


- Penser à cocher « import Field names for first line » pour prendre en compte la première ligne en tant que colonne

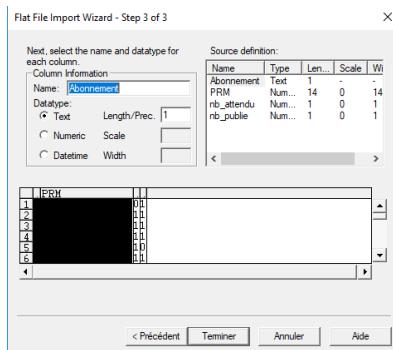


- Cocher Semicolon afin de reconnaître le point-virgule en tant que séparateur pour les champs

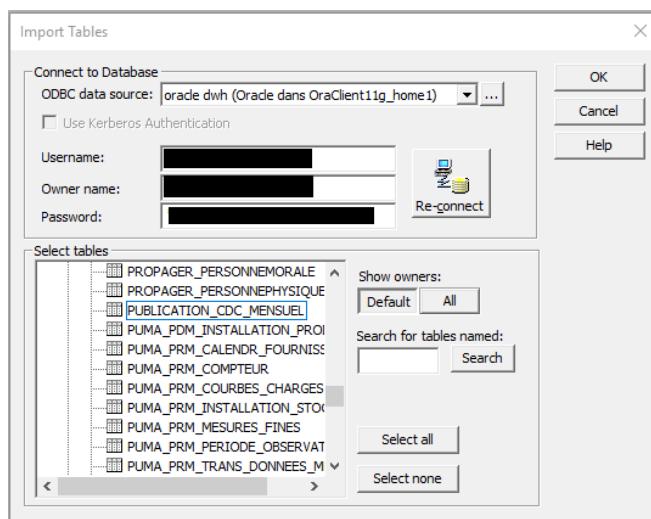




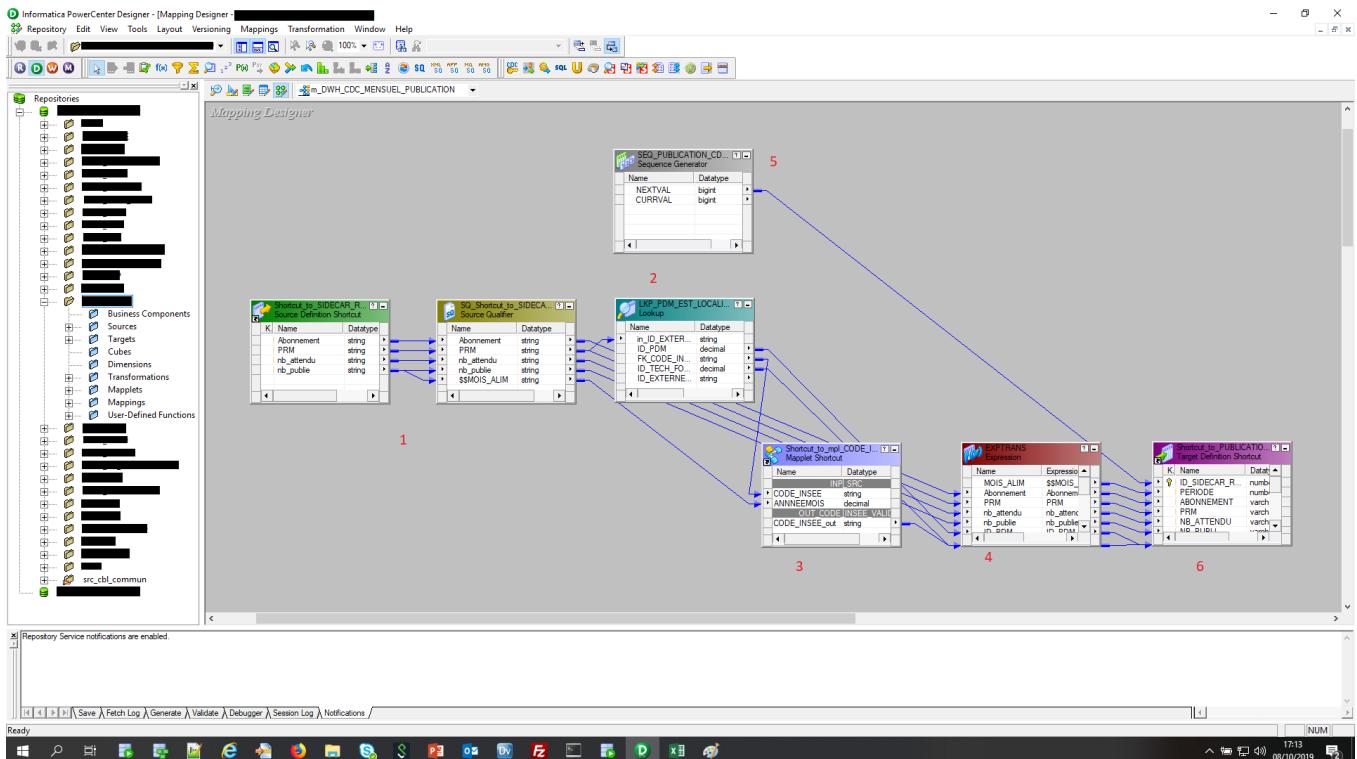
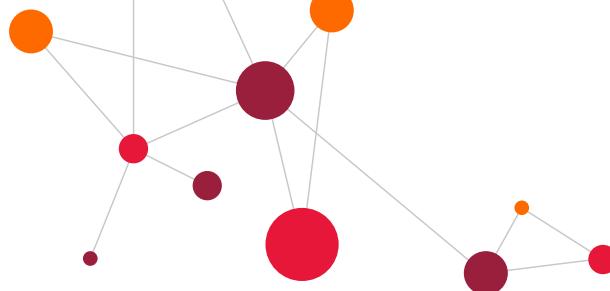
- Enfin, modifier ci-besoin les informations comme le type du champ et sa longueur.



Une fois la source créée, j'ai pu ajouter une nouvelle cible, qui, celle-ci, s'importe tout simplement depuis la base de données :



Maintenant que la source et la cible sont créés, il nous reste à « mapper » le tout .



1. Placer la source :

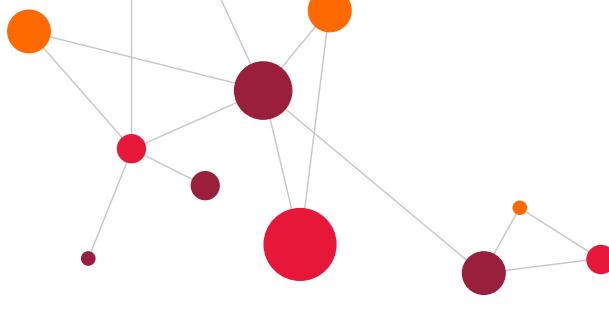
Le « Source Qualifier » se place automatiquement. Il est obligatoire car c'est lui qui contient les instructions SQL, il est possible d'overider la requête de celui-ci.

2. Placer un « Lookup » plutôt qu'un « Source Qualifier » pour des soucis de performance:

En effet, le « Source Qualifier » va analyser toute la table à vérifier alors que le « Lookup » ne va prendre que la valeur des champs de notre source afin de faire un contrôle d'existence.

3. Placer un maplet :

Un maplet est comme un mini mapping réutilisable qui contient des transformations. Ce maplet ci-dessus sert à vérifier les codes Insee et les mettre à jours s'ils sont obsolètes.



4. Placer une expression :

Je mets une expression afin de pouvoir ajouter et transformer des champs qui seront dans la table cible →

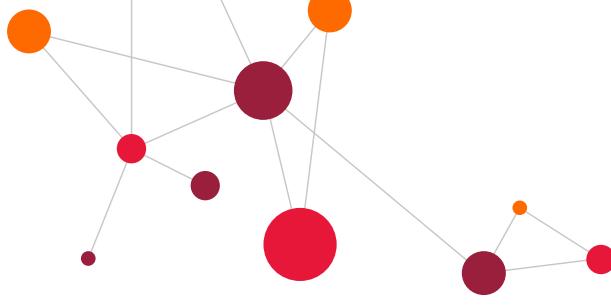
	Port Name	Datatype	Prec	Scale	I	O	V	Expression
1	MOIS_AJLM	string	10	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	\$\$MOIS_AJLM
2	Abonnement	string	255	0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Abonnement
3	PRM	string	14	0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	PRM
4	nb_attendu	string	11	0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	nb_attendu
5	nb_publie	string	3	0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	nb_publie
6	ID_PDM	decimal	9	0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	ID_PDM
7	FK_CODE_INSEE	string	5	0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	FK_CODE_INSEE
8	ID_TECH_FOURNISSEUR1	decimal	19	0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	ID_TECH_FOURNISSEUR1
9	TECH_ID_TRT_UNITAIRE	decimal	10	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	\$\$ID_TRT_UNITAIRE
10	TECH_CREA	date/time	29	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	SYSDATE
11	TECH_MODIF	date/time	29	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	SYSDATE
12	TECH_USER	string	29	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	'DWH_PUBLICATION_CDC_MENSUEL'

5. Placer un « Sequence Generator » :

J'ajoute un « Sequence Generator » afin de créer les id uniques générés automatiquement par Informatica, ils sont incrémentés de 1 à chaque nouveau champ inséré.

6. J'ajoute enfin la cible créée précédemment.

Il me reste maintenant à générer le workflow.



Il y a plusieurs modifications imposées afin de bien se connecter au bon serveur etc...

Même principe pour l'autre mapping.

Maintenant que les 2 mappings sont créés, je vais modifier les fichiers de configuration pour faire le lien entre les entrants et mon WorkFlow.

4.1.3.1 Le fichier domaine_XXXX.csv

Ce fichier consiste à faire la jointure entre le fichier en entrée (flux) et le Workflow.

Nous devons y renseigner :

1. Le nom du flux,
2. Le nom du fichier (qui ne changera pas),
3. Le type fichier entrant,
4. Le numero unique incrémenté de 1 à chaque nouvelle jointure,
5. Le Workflow associé.

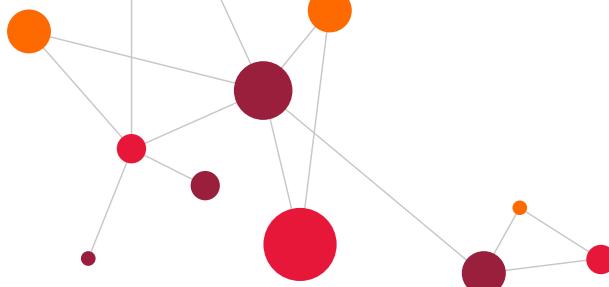
```

7 [REDACTED]_COLLE[REDACTED]:Export[REDACTED]_Tx_Colle[REDACTED].zip;237;w_DWH[REDACTED]DC_MENSUEL_COLLE[REDACTED]
8 [REDACTED]_PUBLIC[REDACTED];SIDE[REDACTED][REDACTED].zip;238;w_DWH[REDACTED]DC_MENSUEL_PUBLIC[REDACTED]
9
  1   2   3   4   5

```

4.1.3.2 Le fichier alim_XXXX.cfg

Ce fichier sert à récupérer des paramètres pour le workflow



CGI

```

GNU nano 2.0.9          File: alim.cfg

## v9.0 ##
#####
[DWH_MGP.s_m_DWH_SUIVI_SMO_HEBDO]
$InputFile 1=<ENTREE/>FICHIER
$PMsessionLogFile=<PM_LOG/s_m_DWH_.log
$BadFile 1=<PM REJECT/s_m_DWH_.bad

## v9.4 ##
#####
[DWH_MGP.s_m_DWH_DC_MENSUEL_PUBLIC]
$InputFile 1=<ENTREE/>FICHIER
$PMsessionLogFile=<PM_LOG/s_m_DWH_DC_MENSUEL_PUBLIC.log
$BadFile 1=<PM REJECT/s_m_DWH_DC_MENSUEL_PUBLIC.bad

[DWH_MGP.s_m_DWH_DC_MENSUEL_COLLE]
$InputFile 1=<ENTREE/>FICHIER
$PMsessionLogFile=<PM_LOG/s_m_DWH_DC_MENSUEL_COLLE.log
$BadFile 1=<PM REJECT/s_m_DWH_DC_MENSUEL_COLLE.bad

## v9.8 ##
#####

Get Help      WriteOut      Read File      Prev Page      Cut Text      Cur Pos
Exit          Justify       Where Is       Next Page      Uncut Text     To spell

```

Le \$inputFile correspond au fichier en entrée

Le \$PMsessionLogFile est impératif pour les logs, idem pour le \$BadFile.

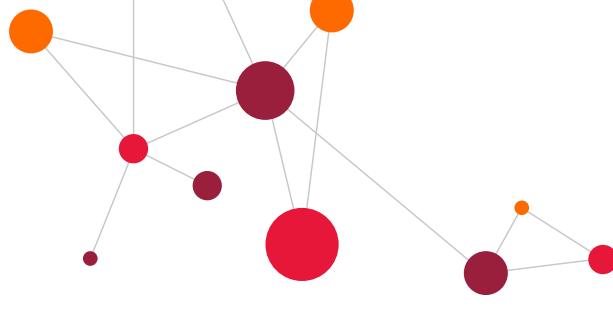
4.1.3.3 Le fichier Pre_trt_unitaire.ksh

```

# cas particulier pour les 3 [REDACTED]
case $NOM_FLUX in
    add_months ${MOIS_ALIM} 1 || exit 1
    MOIS_NOM_FICHIER=${FONC_RETOUR}; unset FONC_RETOUR
;;
    cd ${SAS}
    Mois=`expr substr ${MOIS_ALIM} 5 2`
    Année=`expr substr ${MOIS_ALIM} 1 4`
    echo ${Mois} ${Année}

    Jour=`ls suivie* [0-9][0-9][0-9]-${Mois}-[0-9][0-9].csv |head -1| cut -c43-44`
    echo Jour :${Jour}
    if [[ ${Jour} -gt 0 && ${Jour} -lt 16 ]]
    then
        MOIS_NOM_FICHIER=${Année}-${Mois}-${Jour}
        echo "Première quinzaine"
        QUINZAINE=Q1
    else
        echo "Seconde quinzaine"
        MOIS_NOM_FICHIER=${Année}-${Mois}-${Jour}
        QUINZAINE=Q2
    fi
;;
    PUBLIC [REDACTED]
    cd ${SAS}
    JOUR=`ls [REDACTED]_${MOIS_ALIM}* |head -1| cut -c19-20`
    RESTE=`ls [REDACTED]_${MOIS_ALIM}* |head -1| cut -c21-44`
    MOIS_NOM_FICHIER=${MOIS_ALIM}${JOUR}${RESTE}
;;
*) MOIS_NOM_FICHIER=${MOIS_ALIM}
;;

```



Ce fichier est essentiel pour l'alimentation.

Un flux classique se base sur la période du fichier pour déterminer le « mois alim » de la table. Or notre fichier est sous une forme différente.

Un flux classique est sous cette forme :

nomflux_201901.csv.

Dans ce cas, le script récupère automatiquement la période en se basant sur le fichier précédemment modifié (domaine_XXXX.csv). Or, dans notre cas, le fichier prendrait pour période **20190101_20190201_20190412205306** (ce qui n'existe pas).

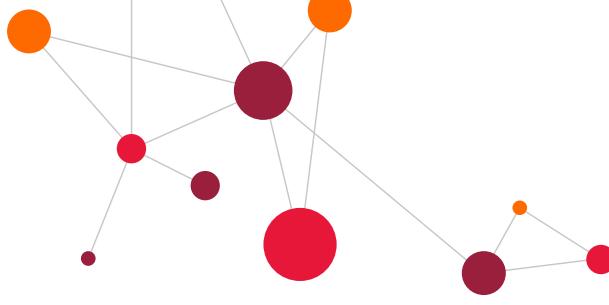
nomflux_20190101_20190201_20190412205306.csv

La seule solution est alors d'ajouter un cas spécifique pour ce flux. J'ai procédé pour cela à un « cut » dans le nom du fichier afin de séparer la période et le reste du nom du fichier. Ainsi, le fichier a pu être reconnu tout en ayant une bonne période.

4.1.4 **Test d'alimentation**

Maintenant que la partie Unix est terminée, je peux faire les tests, qui sont la partie décisive pour l'US. Plusieurs paramètres sont à prendre en compte :

- Le flux est-il reconnu?
- La récupération se fait elle bien?
- Le traitement Informatica se lance-t-il bien lors de l'alimentation?
- La période est-elle la bonne?
- Les données sont-elles correctement insérées? Au bon format?
- Les données correspondent elles à la demande ?
- L'incrémentation de l'Id se fait-elle?
- La partition est-elle supprimée lorsqu'on retraite le même fichier?



Je peux utiliser mes propres jeux de données pour les tests, ou utiliser ceux fournis par les entrants comme ci-après :

l1 RAW_20190201_20190301_20190412205410.csv	
1	abonnement;PRM;nb_attendu;nb_publie
1	;30001481273069;1;1
1	;30001480125029;1;1
1	;30001480413367;1;1
1	;30001481273069;1;2
1	;30001421236320;1;1
1	;50015340032331;1;1
1	;30001421699897;1;1
1	;30001480390169;1;1
1	;30001481273069;1;1
1	;30001481202865;1;1
1	;30001481308026;1;1
1	;30001420497309;1;1
1	;30001480390169;1;4
1	;300007941695;1;1
1	;30001481085620;1;1
1	;30001960380628;1;1
2	;30001480275180;1;1
2	;30002154121399;1;1
2	;300007941695;1;1
2	;30001420077678;1;1
2	;30001422255998;1;1
2	;30001480084496;1;1
2	;30001481273069;1;1
2	;30001421479413;1;1
2	;50075507359323;1;1
2	;30001420645577;1;1
3	;30001422121049;1;1
3	;30001480390169;1;1
3	;30009520411269;1;1
3	;30000510442512;1;1
3	;30000220299595;1;1
3	;50098220442390;1;1
3	;30001420645577;1;1
3	;30001424226566;1;1
3	;30001480255051;1;1
3	;3000142221049;1;1
4	;30000510442512;1;1
4	;30000440261609;1;1
4	;30001421529378;1;1
4	;30000410312570;1;1
4	;50098416612799;1;1
4	;30001426130228;1;1
4	;30001421529378;1;1
4	;30001960380841;1;1
4	;30001481263606;1;2
4	;30000520326538;1;1
5	;30001420474903;1;1
5	;30001424197192;1;1

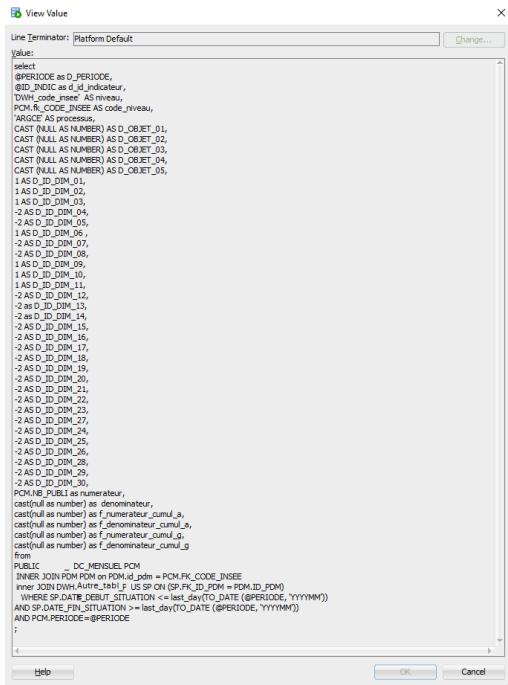
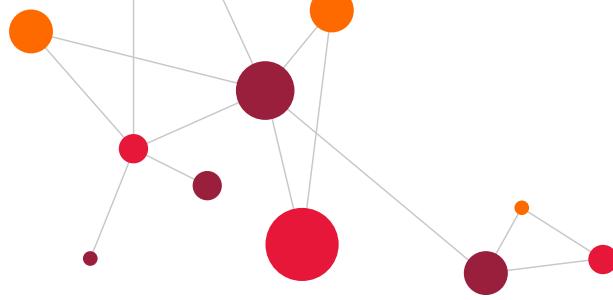
Une fois que tout est cohérent, il faut maintenant créer les indicateurs.

4.1.5 Crédation des indicateurs

Qu'est-ce qu'un indicateur ? Un indicateur est soit un taux, soit un quotient, soit une moyenne ou un pourcentage. Un indicateur est conçu pour synthétiser une information qui a fait l'objet de comparaisons pertinentes. Dans mon cas, je vais créer un indicateur de taux de relevé.

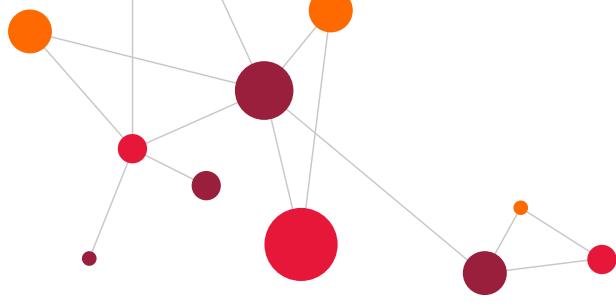
Tout d'abord, je dois me renseigner sur le groupe de l'indicateur, ce qui est crucial pour savoir quelle requête je vais faire pour le calcul. Ensuite, je me base sur une requête existante pour ce groupe et l'adapter (Voir l'Annexe 3 pour le script des indicateurs).

Voici un exemple de requête d'indicateur sur la page suivante : on peut observer le nombre de publications pour la colonne « NUMERATEUR » :



Pour tester les indicateurs, il ne reste plus qu'à faire la commande d'alimentation du datamart (DTM) et laisser le Workflow du groupe d'indicateur tourner. Les résultats seront présents dans la table Agrégée dans le DTM.

Maintenant que tout est prêt, un membre de l'équipe, le CSI (chargé du système d'information), peut maintenant tester, comparer les résultats avec ceux attendus et les communiquer au Product Owner pour que celui-ci puisse faire ses tests complémentaires et valider l'User Story.



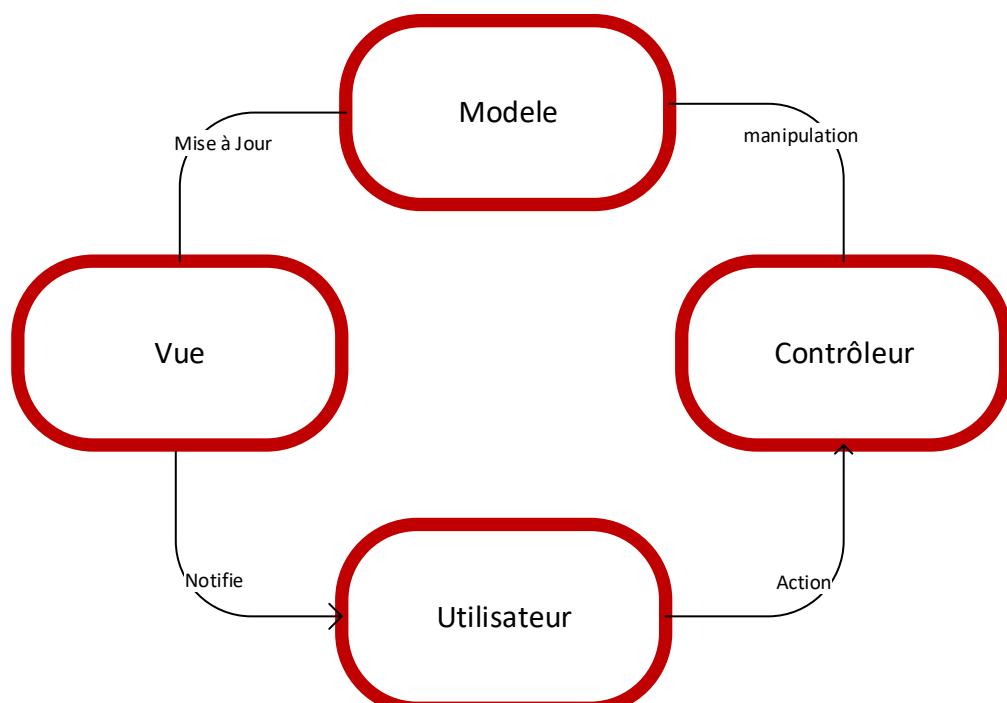
5 Formation

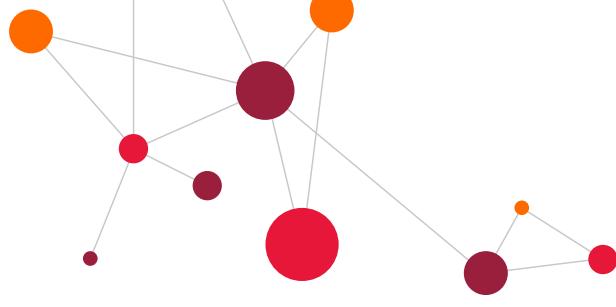
En formation, j'ai eu l'occasion de pratiquer la programmation objet, le développement avec des Framework ou du web.

5.1 L'architecture MVC

L'architecture MVC (Modèle Vue Contrôleur) est un nom donné à une manière d'organiser son code sous trois parties :

- Le modèle : Il gère les données de l'application. Son rôle est de récupérer des informations dans la base de données, de les organiser et de les assembler pour les rendre utilisables par le contrôleur.
- La vue : Elle se concentre sur les interactions avec l'utilisateur. Elle se contente de récupérer des variables pour savoir quoi afficher.
- Le contrôleur : Il gère la dynamique de l'application, c'est la partie qui prend les décisions. Elle fait le lien entre la vue et le modèle en analysant les données de la vue pour les basculer dans le modèle.





5.2 Projet de formation GestionMateriel

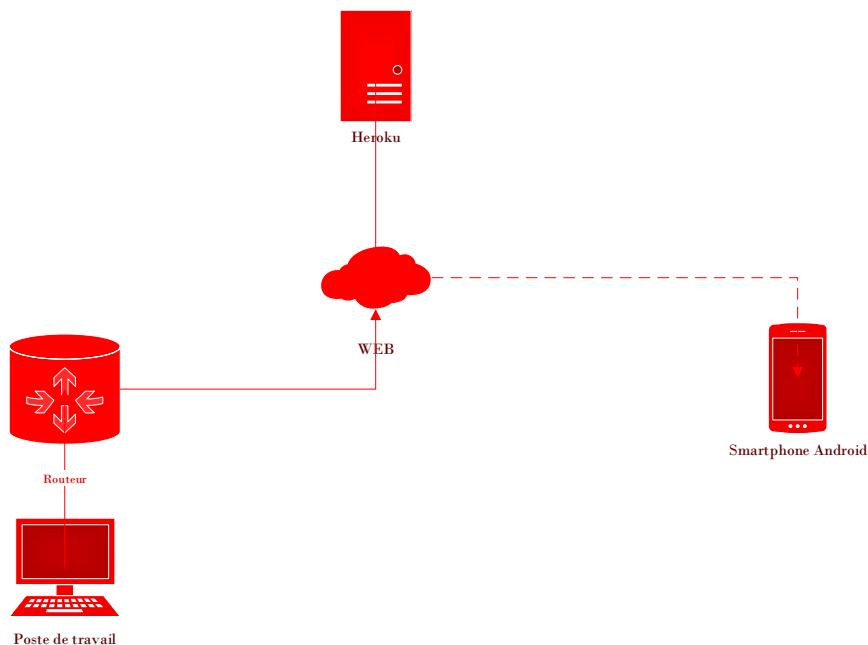
GestionMateriel est un projet évolutif proposé par nos formateurs.

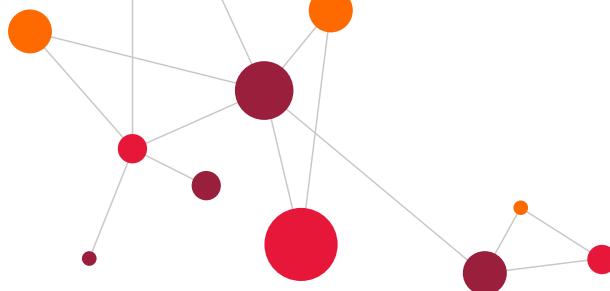
Le projet consistait tout d'abord à créer une application android avec l'IDE android Studio.

Il fallait tout d'abord afficher les clients, les contacts de ces clients et leurs materiels, le tout en lecture à partir d'un fichier JSON contenu dans un site hébergé.

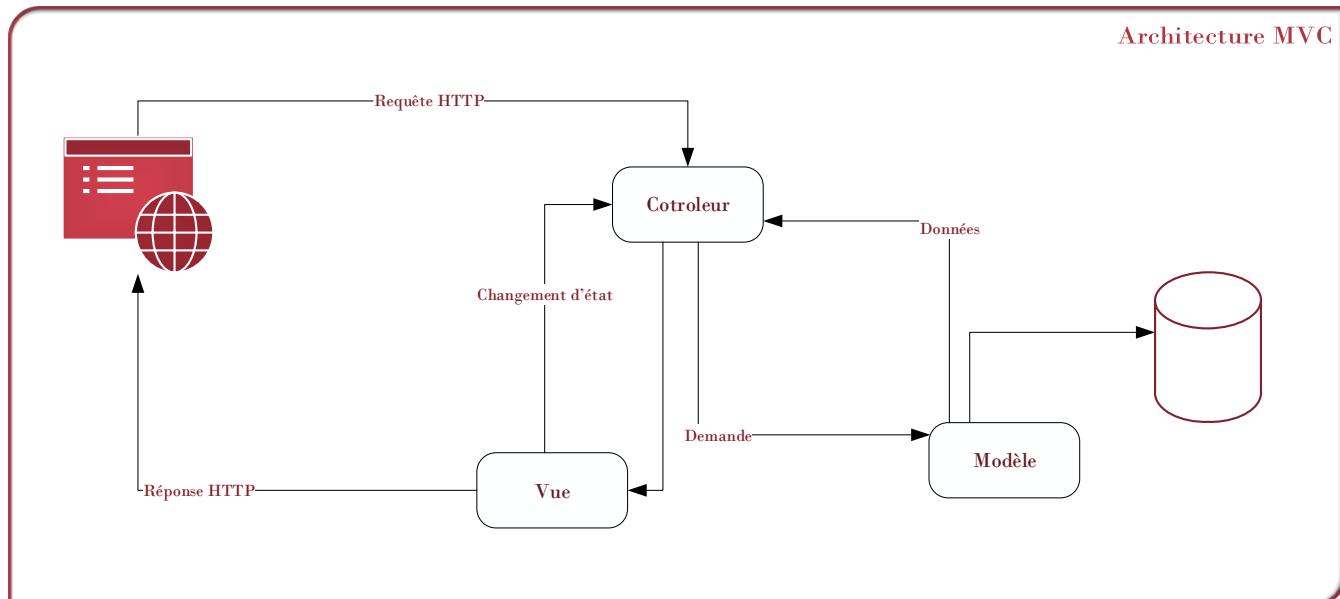
La seconde phase de ce projet était de créer nous-même une API en java.

La troisième phase consistait enfin à créer une application légère afin de pouvoir l'utiliser également sur des périphériques tel que des ordinateurs à partir d'un navigateur internet.





5.2.1 API



L'API est le cœur du projet, elle est indispensable pour les échanges avec la base de données. L'API est développée en JAVA avec Maven pour ses avantages d'extensibilité, de facilité d'installation des plugins et de son respect des conventions.

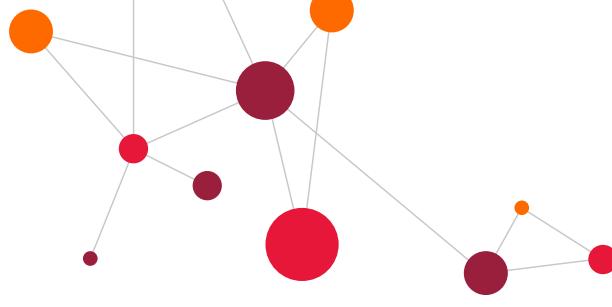
Les méthodes `@GetMapping` est le point d'entrée de l'API, elle reçoit les Requêtes http de notre application android ou du client léger et y répond.

Lors de la création d'un nouveau client, le contrôleur reçoit la requête, qui est transférée dans le service puis dispatchée dans l'entité client afin de vérifier la conformité de la requête. Le client sera alors créé.

5.2.2 Android

L'application Android communique directement avec l'API externe expliquée précédemment. Elle permet notamment :

- D'afficher une liste de clients avec leurs informations,
- D'afficher les contacts d'un client,
- D'afficher les matériels d'un client ainsi que les ports,
- De scanner le QR Code d'un matériel et d'afficher ses informations dans l'application.



Elle permettra aussi (factures non implémentées) d'ajouter un client, des contacts et/ou des matériels

La première étape est de récupérer les clients et de les afficher dans une liste :

Pour récupérer les clients, la librairie Volley est utilisée. Elle peut recourir aux méthodes GET/POST/PUT/DELETE. Volley envoie alors une requête http, puis récupère le tout sous format JSON, transformé en objet grâce à la librairie Jackson.

Une fois les utilisateurs récupérés, je les affiche dans une liste. Pour cela, j'utilise le widget RecyclerView .

Lorsqu'on clique sur un client, une activité composée de fragments est alors affichée.

Une activité peut être alors découpée en un ou plusieurs fragments afin d'être flexible et modulable visuellement. Elles peuvent être considérées comme des sous activités.

Dans notre cas, sous une même interface, 3 onglets (fragments) apparaissent lors de la sélection du client :

- Les informations du client
- Les contacts
- Le matériel.

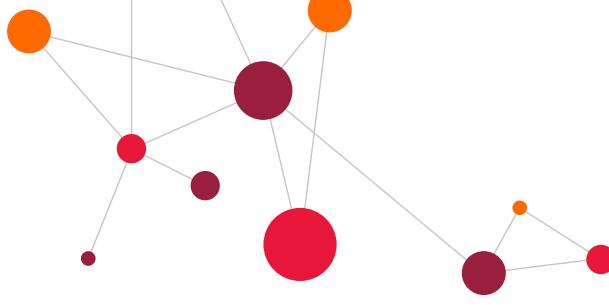
L'onglet contact est intéressant car j'utilise un widget cardView pour un design carte de visite.

Pour créer cette liste, il faut tout d'abord Ajouter le widget dans le Gradle.

Le Gradle est un gestionnaire de dépendance puissant, qui permet de consolider un certain nombre de ressources externes pour le projet.

```

dependencies {
    implementation fileTree(dir: 'libs', include: ['*.jar'])
    implementation 'androidx.appcompat:appcompat:1.0.2'
    implementation 'com.google.android.material:material:1.0.0'
    implementation 'androidx.constraintlayout:constraintlayout:1.1.3'
    implementation 'androidx.lifecycle:lifecycle-extensions:2.0.0'
    testImplementation 'junit:junit:4.12'
    androidTestImplementation 'androidx.test:runner:1.2.0'
    androidTestImplementation 'androidx.test.espresso:espresso-core:3.2.0'
    implementation 'com.fasterxml.jackson.core:jackson-core:2.9.8'
    implementation 'com.fasterxml.jackson.core:jackson-annotations:2.9.8'
    implementation 'com.squareup.picasso:picasso:2.5.2'
    implementation 'com.fasterxml.jackson.core:jackson-databind:2.9.8'
    implementation 'androidx.recyclerview:recyclerview:1.1.0-beta01'
    implementation 'com.android.volley:volley:1.1.1'
    implementation 'com.android.support:cardview-v7:22.1.1'
```



Il faut ensuite préciser dans le XML que ce sera un style CardView :

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.cardview.widget.CardView xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_margin="5dp"

    app:cardBackgroundColor="@android:color/white"
    app:cardCornerRadius="2dp"
    android:includeFontPadding="false"
    app:cardElevation="2dp">

    <!-- Les CardView possèdent des attributs supplémentaires dont
        - cardBackgroundColor
        - cardElevation pour l'élévation (donc aussi l'ombre)
        - cardCornerRadius pour arrondir les angles
    -->

    <androidx.constraintlayout.widget.ConstraintLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content">

        <LinearLayout

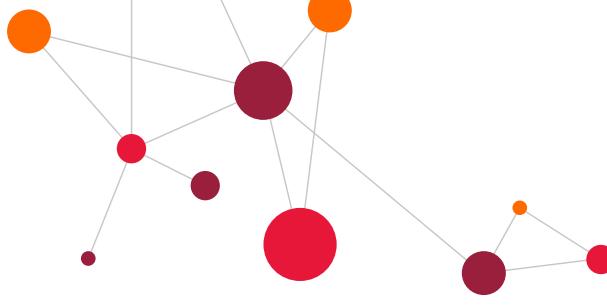
            android:layout_width="0dp"
            android:layout_height="wrap_content"
            android:background="#f2flef"
            android:orientation="vertical"
            app:layout_constraintBottom_toBottomOf="parent"
            app:layout_constraintEnd_toEndOf="parent"
            app:layout_constraintHorizontal_bias="0.037"
            app:layout_constraintStart_toStartOf="parent"
            app:layout_constraintTop_toTopOf="parent"
            app:layout_constraintVertical_bias="0.0">

            <!-- Les CardView agissent comme des FrameLayout,
                pour avoir une organisation verticale nous devons
                donc rajouter un LinearLayout -->

        </LinearLayout>
    </androidx.constraintlayout.widget.ConstraintLayout>
</androidx.cardview.widget.CardView>
```

Ensuite, je dois modifier le fichier du fragment :

Pour afficher les cartes, il faut tout d'abord déclarer une recyclerView dans le layout des contacts, utiliser ensuite un adapter (bindContact) afin de faire la liaison entre le recyclerView et les données de la liste puis, enfin, créer une viewHolder afin de représenter visuellement les éléments de la liste de données dans le recycler view pour compléter cette création.



```

package com.example.gestionmatieriel.presentation.detailclient;
import ...

public class ContactsFragment extends Fragment {
    private RecyclerView recyclerViewContact;

    @Override
    public View onCreateView(@NonNull LayoutInflater inflater, ViewGroup container,
                           Bundle savedInstanceState) {
        View root = inflater.inflate(R.layout.fragment_contacts_layout, container, false);
        DetailClientActivity activity = (DetailClientActivity) getActivity();
        Client c = activity.getClient();

        recyclerViewContact = (RecyclerView) root.findViewById(R.id.recycleViewContacts);
        recyclerViewContact.setLayoutManager(new LinearLayoutManager(activity));
        recyclerViewContact.setAdapter(new ListeContactsAdapter(c.getContacts()) {});

        FloatingActionButton ajoutContact = root.findViewById(R.id.ajoutContact);
        ajoutContact.setOnClickListener((View) -> {
            Snackbar.make(view, "Replace with your own action", Snackbar.LENGTH_LONG)
                    .setAction("Action", new Intent(null).setClass(AjoutContactActivity.class))
                    .startActivity(i);
        });
    }

    return root;
}

private class ContactViewHolder extends RecyclerView.ViewHolder {
    private Contact contact;
    private TextView nomContact;
    private TextView fonctionContact;
    private TextView prenomContact;
    private TextView mailContact;
    private TextView telContact;

    //itemView est la vue correspondante à 1 cellule
    public ContactViewHolder(@NonNull View itemView) {
        super(itemView);
        //c'est ici que l'on fait nos findView
        nomContact = (TextView) itemView.findViewById(R.id.nomContact);
        prenomContact = (TextView) itemView.findViewById(R.id.prenomContact);
        fonctionContact = (TextView) itemView.findViewById(R.id.fonctionContact);
        telContact = (TextView) itemView.findViewById(R.id.telContact);
        mailContact = (TextView) itemView.findViewById(R.id.mailContact);
    }
}

public void bindContact(Contact contact) {
    nomContact.setText(contact.getPersonne().getNom());
    prenomContact.setText(contact.getPersonne().getPrenom());
    fonctionContact.setText(contact.getFonction().getLibelle());
    mailContact.setText(contact.getPersonne().getMail());
    telContact.setText(contact.getPersonne().getTelephone());
    this.contact = contact;

    mailContact.setOnClickListener((v) -> {
        Intent intent = new Intent(Intent.ACTION_SEND);
        intent.setType("plain/text");
        intent.putExtra(Intent.EXTRA_EMAIL, new String[] { mailContact.getText().toString() });
        intent.putExtra(Intent.EXTRA_SUBJECT, value + "SessionClient");
        intent.putExtra(Intent.EXTRA_TEXT, "AjoutActivity").createChooser(intent, "Envoyer");
    });
    telContact.setOnClickListener((v) -> {
        Intent smsIntent = new Intent(Intent.ACTION_VIEW);
        smsIntent.setType("vnd.android-dir/mms-sms");
        smsIntent.putExtra("address", telContact.getText().toString());
        startActivity(smsIntent);
    });
}
}

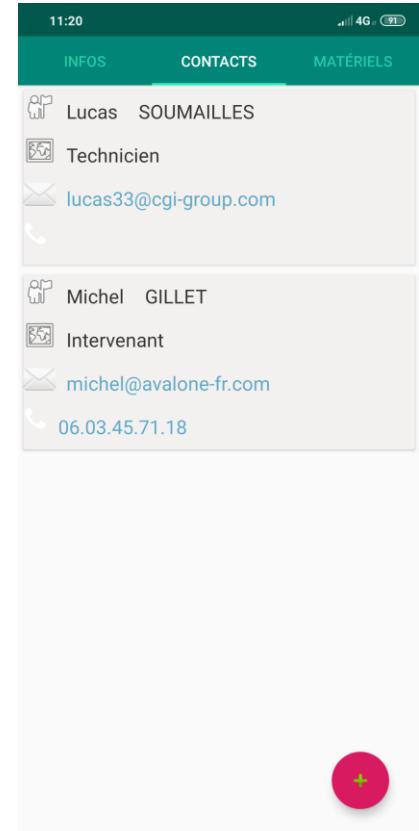
private class ListeContactsAdapter extends RecyclerView.Adapter<ContactViewHolder> {
    List<Contact> listContact;
    public ListeContactsAdapter(List<Contact> list) {
        this.listContact = (list == null ? new ArrayList<Contact>() : list);
    }

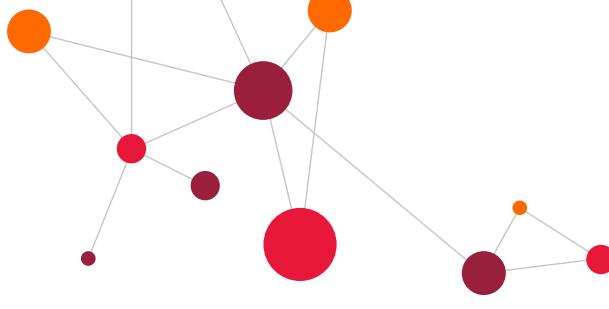
    @Override
    public ContactViewHolder onCreateViewHolder(ViewGroup viewGroup, int itemType) {
        View view = LayoutInflater.from(viewGroup.getContext()).inflate(R.layout.liste_contacts_layout, viewGroup, false);
        return new ContactViewHolder(view);
    }

    @Override
    public void onBindViewHolder(ContactViewHolder myViewHolder, int position) {
        Contact contact = listContact.get(position);
        myViewHolder.bindContact(contact);
    }

    @Override
    public int getItemCount() { return listContact.size(); }
}
}

```





5.2.3 Client léger

Après Android, il est également important de pouvoir y accéder avec un navigateur web.

Le framework utilisé est Angular. Pour le design, j'ai choisi Boostrap.

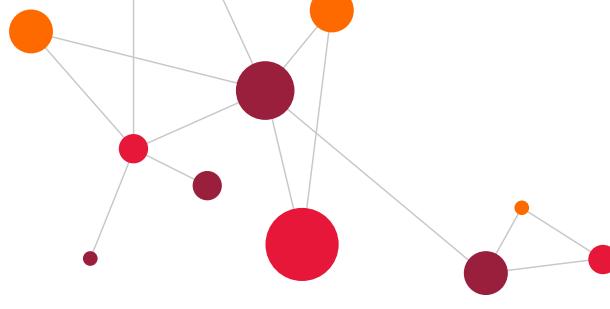
Contrairement à l'application Android, l'ajout du client est implémenté.

Il faut au préalable avoir créé l'entité client ainsi que les entités dépendantes,

```
EtiennePS, a month ago | 2 authors (maxime and others)
1 import { Ville } from './ville';           maxime, 2 months ago • Ajout des entités
2 import { Contact } from './contact';
3 import { Materiel } from './materiel';
4
5 EtiennePS, a month ago | 2 authors (maxime and others)
6 export class Client {
7     public constructor (
8         public nom: string,
9         public adresse1: string,
10        public adresse2: string,
11        public ville : Ville) {}
12
13    public id: number;
14    public contacts: Array<Contact>;
15    public materiels: Array<Materiel>;
16 }
```

ainsi que GenericService.ts afin de fournir à l'application l'url de l'API :

```
1 import { HttpClient, HttpHeaders } from '@angular/common/http';
2
3 EtiennePS, a month ago | 1 author (EtiennePS)
4 export abstract class GenericService {
5     protected baseUrl = "https://api-gestion-materiel.herokuapp.com/api/v1";
6     protected httpOptions = {
7         headers : new HttpHeaders({
8             'Accept': 'application/json',
9             'Content-Type': 'application/json'
10        })};
11    constructor(protected http: HttpClient) { }
12
13 }
```



Le formulaire d'ajout du client est ajouté dans une modale, ou fenêtre flottante :

```
***** MODALES *****
displayAjoutClient() {
  this.isModalCreationClient = true;
  this.formClient.reset();
  this.formClient.patchValue({
    idVille: 0
  });
  this.showModalClient = true;
}

displayEditClient(idClient: number) {
  this.isModalCreationClient = false;
  this.formClient.reset();
  this.clientService.getClientById(idClient).subscribe(c => {
    this.formClient.patchValue({
      id: c.id,
      nom: c.nom,
      adresse1: c.adresse1,
      adresse2: c.adresse2,
      idVille: c.ville.id
    });
  });
  this.showModalClient = true;
}

onSubmitClient() {
  let v: Ville = new Ville();

  if(this.formClient.value.idVille < 1) {
    v.nom = this.formClient.value.nomVille;
    v.codePostal = this.formClient.value.cpVille;
  }
  else {
    v.id = this.formClient.value.idVille;
  }

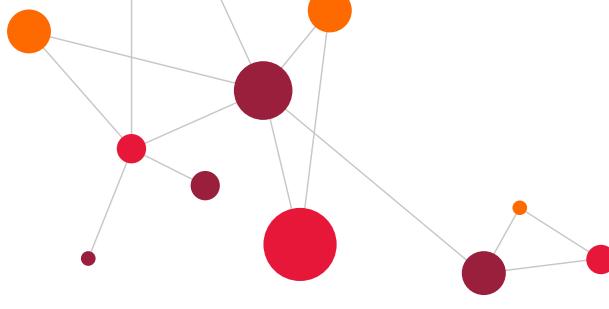
  let c: Client = new Client(this.formClient.value.nom, this.formClient.value.adresse1, this.formClient.value.adresse2, v);

  if(this.isModalCreationClient) {
    this.clientService.createClient(c).subscribe(data => {
      this.update();
      this.showModalClient = false;
    });
  }
  else {
    c.id = this.formClient.value.id;
    this.clientService.editClient(c).subscribe(data => this.update());
  }
}
}
```

```
<div class="w3-container">
  <button (click)="displayAjoutClient()" class="w3-button w3-blue">Créer</button>

  <div class="w3-modal" *ngIf="showModalClient">
    <div class="w3-modal-content">
      <header class="w3-container w3-teal">
        <span (click)="showModalClient=false" class="w3-button w3-display-topright">&times;</span>
        <h2>{{ isModalCreationClient ? "Créer" : "Modifier" }} client</h2>
      </header>
      <div class="w3-container">
        <form [FormGroup]="formClient">
          <label for="nom">Nom</label>
          <input type="text" formControlName="nom" class="w3-input w3-border" placeholder="Nom du client"/>
          <label for="adresse1">Adresse</label>
          <input type="text" formControlName="adresse1" class="w3-input w3-border" placeholder="1 rue de la liberté"/>
          <label for="adresse2">Complément d'adresse</label>
          <input type="text" formControlName="adresse2" class="w3-input w3-border" placeholder="ZI"/>
          <label for="idVille">Ville</label>
          <select name="idVille" formControlName="idVille" class="w3-select">
            <option value="0">Nouvelle ville</option>
            <option *ngFor="let ville of villes" value="{{ ville.id }}">{{ ville.nom + " (" + ville.codePostal + ")" }}</option>
          </select>

          <div *ngIf="+formClient.value.idVille < 1">
            <label for="nomVille">Nom de la ville</label>
            <input type="text" formControlName="nomVille" class="w3-input w3-border" placeholder="Bordeaux"/>
            <label for="cpVille">Code Postal</label>
            <input type="text" formControlName="cpVille" class="w3-input w3-border" placeholder="33000"/>
          </div>
        </form>
      </div>
    </div>
  </div>
```



Pour pouvoir appeler cette fonction, il faut créer dans le service les fonctions pour envoyer les différentes requêtes directement vers l'API :

```
op > service > TS client.service.ts > ...
EtiennePS, a month ago | 2 authors (EtiennePS and others)
import { Injectable } from '@angular/core';
import { Observable } from 'rxjs';
import { Client } from './entity/client';
import { GenericService } from './generic.service';

EtiennePS, a month ago | 2 authors (EtiennePS and others)
@Injectable({
  providedIn: 'root'
})
export class ClientService extends GenericService {

  getClientById(idClient: number): Observable<Client> {
    return this.http.get<Client>(this.baseUrl + '/clients/' + idClient, this.httpOptions);
  }

  getAllClients(): Observable<Client[]> {
    return this.http.get<Client[]>(this.baseUrl + '/clients', this.httpOptions);
  }

  createClient(client: Client): Observable<Client> {
    return this.http.post<Client>(this.baseUrl + '/clients', client, this.httpOptions);
  }

  editClient(c: Client): Observable<Client> {
    return this.http.put<Client>(this.baseUrl + '/clients/' + c.id, c, this.httpOptions);
  }

  deleteClient(idClient: number): Observable<Client> {
    return this.http.delete<Client>(this.baseUrl + '/clients/' + idClient, this.httpOptions);
  }
}
```

Voici le résultat visible sur le site:

Créer client

Nom

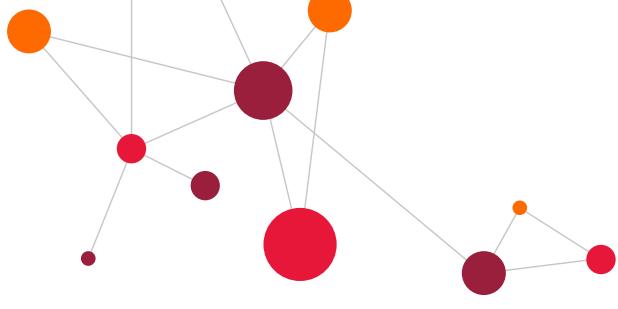
Adresse

Complément d'adresse

Ville

Code Postal

Créer



6 Conclusion

Cette année d'alternance professionnelle m'a permis de parfaire mes connaissances et m'a conforté dans mon choix de reconversion dans le domaine de conception et de développement d'applications numérique.

Elle m'a aussi permis de comprendre que ce métier ne se limite pas au développement, c'est avant tout un métier de contact, d'analyse et de rigueur, ce que j'apprécie.

Ma vision du développeur a donc complètement évolué entre mon entrée dans la formation et en cette fin d'année. En effet, la collaboration en équipe est essentielle pour le bon déroulement d'un projet et nous sommes amenés à beaucoup dialoguer pour, ensemble, résoudre des problèmes, partager des connaissances ou tout simplement améliorer la qualité de vie d'un projet.



ANNEXES

Annexe 1 : Referentiel

Bloc de compétences : Qualité et sécurisation du code réalisé

Compétences ou capacités qui seront évaluées	Critères d'évaluation	Exemples d'activités et tâches	Activités pratiquées	Origine de l'acquisition	Preuves apportées & ref. annexe
- Formaliser, identifier les résultats attendus.	La liste de contrôle des attendus fonctionnels est paraphée.	Étude de l'existant. Rédaction du cahier des Spécifications fonctionnelles.	Rédaction Spécifications fonctionnelles détaillées (SFD) pour la création d'une vue matérialisée	E(1 S)	Annexe 2
Respecter des contraintes. Respecter les recommandations qualité de la norme en vigueur pour l'architecture des logiciels.	Un plan d'assurance qualité est observé. L'application est organisée en couches indépendantes. Les règles métier sont encapsulées dans des services logiciels. L'accès aux données est réalisé par des services logiciels indépendants du mode de stockage.	Conception/architecture d'applications logicielles. Conception de services métiers. Conception de services d'accès aux données.	Conception d'une API en Java Modèle MVC (Modèle vue contrôleur) API sous Spring	E(1 M) F(1 M) F (2 M)	Page 38 Page 36 Page 38
Anticiper les évolutions. Qualifier les risques	L'exécution de l'application est répartie entre un nombre d'ordinateurs adapté au contexte.	Détermination du nombre de tiers de l'application.	Application légère sous angular avec communication API	F(1M)	Page 42

<p>Respecter une norme de présentation des écrans et documents de sortie.</p>	<p>Un formulaire d'estimation des risques est rempli.</p> <p>Une norme de présentation des données est respectée. Les interfaces Homme/Machine sont validées.</p>	<p>Réalisation d'une interface homme/machine (IHM)</p> <p>Estimation, qualification des risques sécurité.</p> <p>Réalisation des maquettes de sorties interactives.</p> <p>Réalisation des maquettes de sortie imprimée.</p>	<p>Application légère sous angular avec communication API.</p> <p>Remontée d'Alerte sur l agrandissement d'un champ non prévu.</p> <p>Création de rapports BO</p> <p>Création maquette Projet Android</p>	<p>F(1M)</p> <p>E(1H)</p> <p>E(3s)</p>	<p>Page 42</p> <p>Analyse en cours à ce moment</p> <p>Page 15 Annexe 11</p>
<p>Concevoir des programmes avec une orientation objets.</p> <p>Garantir un accès sécurisé aux données.</p>	<p>Une programmation orientée objets est utilisée.</p> <p>Le taux de réutilisation du code utile est > 80 %. Des gabarits sont utilisés. Une charte de nommage est utilisée.</p>	<p>Programmation de logiciels.</p>	<p>API</p> <p>Création de scripts SQL avec des conventions</p>	<p>F(1M)</p> <p>E(3j)</p>	<p>Page 38</p> <p>Annexe 3</p>

Livrer le logiciel déverminé.	Le taux de documentation interne du code est > 8 % et < 15 %. Les anomalies d'accès aux données ne génèrent pas d'interruption de l'exécution et sont répertoriées.	Programmation de l'accès aux données de l'entreprise. Tests unitaires. Préparation des jeux de tests.	Gabarits, fonctions des scripts SQL TU/FTU/TNR pour éviter les régressions	E(1h)	Annexe 3
Livrer le logiciel conforme aux attentes.	Des outils de contrôle automatique du code sont utilisés. Aucun défaut visible ne persiste. Les contraintes spécifiques au projet sont respectées. Un manuel d'assurance qualité est respecté. Une méthode de recettage est utilisée. L'étape du projet est validée.	Contrôles de l'existence d'anomalies. Recettage du logiciel. Validation d'une étape du projet.	Utilisation Outil intégré de contrôle des scripts Des documentations internes et des modops sont créés Une fois le développement terminé, recettage par le fonctionnel puis recettage et validation par le Product Owner	E(3j) E(1h) E(1j) E(1S)	Annexe 4 Outil interne au projet non divulgué Annexe 5 Page 35
Clôturer une mission.	Le PV de réception du logiciel est validé.	Mise en exploitation.	Validation des US par le PO et mise en Production	E(4H)	Annexe 6

Bloc de compétences : Audit, conception, méthode de projet

Compétences ou capacités qui seront évaluées	Critères d'évaluation	Exemples d'activités et tâches	Activités pratiquées	Origine de l'acquisition	Preuves apportées & ref. annexe
Formaliser des processus	La procédure du service utilisateur est formalisée et validée.	Étude de l'existant. Identification des procédures en place.	Modification et versioning d'un indicateur	E(1j)	Annexe 7
	La procédure du service utilisateur est conforme aux règles du système de management des services de l'entreprise.	Contrôle de la conformité des procédures utilisées avec la gouvernance de l'entreprise.	Echanges avec les décideurs, remontée par mail, validation puis mise à jour des documents.	E(1S)	
Formaliser les règles de gestion et d'organisation des données de l'entreprise.	La circulation du document résultat du traitement prévu est matérialisée dans un diagramme de workflow.	Recensement des documents utilisés, identification de leur circulation et des acteurs concernés.	Mise a jour des modops et capitalisation des méthodes utilisées	E(1A)	Annexe 5
	La proposition de reconstruction de la procédure est validée.	Reconfiguration de procédure.			
	La base de données est modélisée.	Conception d'une base de données.	Création d'une base de donnée pour projet de POEI	E(1M)	Annexe 9,10

Une méthode de conception par objets est utilisée.	Concevoir des éléments logiciels réutilisables.	Conception de l'architecture applicative.	Utilisation d'un Maplet Informatica	E(1j)	Page 29
Une méthode AGILE est utilisée. Absence de signaux d'alertes au point de contrôle du projet.	Produire du logiciel en équipe. Remonter les alertes au(x) décideur(s).	Programmation en équipe. Écriture de code. Coordination de l'avancement.	Agilité du projet, programmation en équipe Morning (agilité) avec remontée aux décideurs	E(1A) E(2H)	Page 20 à 23 Page 20 à 23
Les étapes du projet sont planifiées.	Estimer des délais.	Planification des tâches du projet.	Utilisation de GOJIRA et estimation des tâches et des délais	E(1A)	Annexe 8
Le projet est conforme au schéma directeur de l'entreprise et respecte les principes d'urbanisation du S.I. Les spécifications fonctionnelles produites respectent le cahier des charges fourni. L'impact de modification est acceptable.	Concevoir une solution logicielle. Anticiper des répercussions.	Conception de la solution logicielle.	Respect des normes et des contraintes (processus livraison, Agilité, Programmation de scripts) Analyse d'impacts / TNR et éventuellement remonter les alertes	E(1A) E(2J)	Annexe 4

Bloc de compétences : Réalisation d'applications logicielles

Compétences ou capacités qui seront évaluées	Critères d'évaluation	Exemples d'activités et tâches	Activités pratiquées	Origine de l'acquisition	Preuves apportées & ref. annexe
Encapsuler des solutions logicielles spécifiques dans des services logiciels génériques.	le service d'accès aux données est opérationnel.	Programmation. Investigations documentaires fonctionnelles ou techniques complémentaires.	Utilisation d'ETL	E(1A)	Page 26 à 30
Produire du logiciel générique réutilisable.	Des services logiciels internes sont réutilisables.	Transcription des spécifications fonctionnelles en algorithmes.	Lire une SFD fonctionnelle en solution technique	E(1S)	Annexe 2
Produire du logiciel partageable.	Des services logiciels sont partageables en local.	Transcription des algorithmes en code source. Compilation, déverminage du code source.	Utilisation des dossiers de partage		
Intégrer des éléments logiciels hétérogènes et produire des exécutables livrables.	Des services logiciels sont partageables à distance.	Agglomération des différents éléments logiciels en unités de traitement, réalisation des tests unitaires.	API en Stateless	E(1M)	Page 38
Modifier un algorithme sans générer de dysfonctionnements.	Le logiciel est livrable, prêt pour la mise en production.	Mise à jour du planning de réalisation.	Remplissage bon de livraison	E(3M)	Annexe 6
Contrôler des délais.	La modification n'entraîne pas de régression fonctionnelle.		Utilisation de GOJIRA et Participation aux morning	E(1A)	Page 23 & Annexe 8
	Le compte-rendu d'activité est renseigné, les écarts sont constatés.				

Bloc de compétences : Communiquer avec les acteurs du projet

Compétences ou capacités qui seront évaluées	Critères d'évaluation	Exemples d'activités et tâches	Activités pratiquées	Origine de l'acquisition	Preuves apportées & ref. annexe
User d'une communication professionnelle tant en français qu'en anglais. Interagir efficacement dans un environnement de travail collaboratif.	<p>Le compte-rendu de la réunion est validé.</p> <p>Le score du TOEIC est > 749</p> <p>Le document collectant l'expression des besoins des utilisateurs est validé.</p> <p>L'aide du logiciel est rédigée.</p> <p>La documentation du livrable est diffusée.</p> <p>La présentation est appréciée.</p>	<p>Élaboration et rédaction de documents techniques, commerciaux ou internes à destination, des utilisateurs, des clients ou des collaborateurs, ...</p> <p>Rédaction des spécifications fonctionnelles de la solution informatique.</p> <p>Écriture des interfaces homme/machine.</p> <p>Relations avec les clients.</p> <p>Animation de réunions de travail et interviews d'utilisateurs.</p> <p>Démonstrations, recettement de livrables.</p>	<p>Création de MODOP a destination des développeurs</p> <p>SFD</p> <p>Travail en collaboration avec les PO et les développeurs</p> <p>Sprint review, présenter les tâches au Product Owner</p>	<p>E(1s)</p> <p>E(3J)</p> <p>E(1A)</p>	<p>Annexe 5</p> <p>Annexe 1</p> <p>Page 23</p>
	Les utilisateurs sont opérationnels, le transfert des nouvelles compétences est validé.	Formation des utilisateurs au logiciel.	Formation d'utilisation d'un navigateur internet à des personnes âgées/ Passage de connaissances,	B(2J) E(1S)	

Réaliser des échanges de données informatisés (EDI).		Réalisation d'un procédé d'échange de données informatisées. Rétro-documentation de logiciels et de bases de données. Consolidation, agrégation de données. Programmation de l'interface d'échange de données.	Utilisation d'un ETL afin de récupérer et de transformer des données Utilisation des bases de données de Développement, recette, MCO	E(1J)	Pages 26 à 30
Automatiser des traitements.		Réalisation d'un environnement de tests. Création, configuration de machines virtuelles. Installation, configuration de serveurs d'applications, Web et base de données. Écriture de scripts systèmes pour adapter l'environnement d'exécution.	Environnement de développement, création de jeux de données, modification des scripts ksh. Création de machines virtuelles de test	E(1J)	Pages 26 à 30
Programmer des scripts systèmes.					Pages 33-34

Bloc de compétences : Adapter l'environnement d'exécution, échanger des données entre logiciels

Compétences ou capacités qui seront évaluées	Critères d'évaluation	Exemples d'activités et tâches	Activités pratiquées	Origine de l'acquisition	Preuves apportées & ref. annexe
Réaliser des échanges de données informatisés (EDI).	Les données sont consolidées.	Réalisation d'un procédé d'échange de données informatisées.	Utilisation d'un ETL afin de récupérer et de transformer des données	E(1J)	Pages 26 à 30
Automatiser des traitements.	La base de données tierce est accédée.	Rétro-documentation de logiciels et de bases de données. Consolidation, agrégation de données. Programmation de l'interface d'échange de données.	Utilisation des bases de données de Développement, recette, MCO	E(1J)	Pages 26 à 30
Programmer des scripts systèmes.	L'environnement de tests est opérationnel.	Réalisation d'un environnement de tests. Création, configuration de machines virtuelles. Installation, configuration de serveurs d'applications, Web et base de données. Écriture de scripts systèmes pour adapter l'environnement d'exécution.	Environnement de développement, création de jeux de données, modification des scripts ksh. Création de machines virtuelles de test	E(1J)	Pages 33-34

Annexe 2 : Extrait d'une spécification fonctionnelle de création de vue

A	B	C	D	E	F
1	CIBLE	SOURCE			
2	Colonne	Flux src	Champ source	Type	Commentaire / Règle de gestion
3	ANNEE	A_SUIV_CONTRAT A_RELIEVE_MENSU(H) A_GES_CONTRAT_M A_PUBLIC_PROD A_RELIEVE_MENS A_MENS_NATION A_MENS A_FACT_PROD	D_PERIODE	-	Récupérer l'année
4	D_PERIODE	A_SUIV_CONTRAT A_RELIEVE_MENSU(H) A_GES_CONTRAT_M A_PUBLIC_PROD A_RELIEVE_MENS A_MENS_NATION A_MENS A_FACT_PROD	D_PERIODE	-	
5	F_DENOMINATEUR	A_SUIV_CONTRAT A_RELIEVE_MENSU(H) A_GES_CONTRAT_M A_PUBLIC_PROD A_RELIEVE_MENS A_MENS_NATION A_MENS A_FACT_PROD	F_DENOMINATEUR	-	
7	CODE_UNIQ_INDICATEUR	TEC_INDICATEUR TEC_REC_INDICATEUR A_MENS A_RELIEVE_MENS A_MENS_NATION A_MENS A_FACT_PROD	CODE_UNIQ_INDICATEUR	-	
8	LIB_INDICATEUR F_NUMERATEUR_CUMUL_A	TEC_INDICATEUR TEC_REC_INDICATEUR A_MENS A_RELIEVE_MENS A_MENS_NATION A_MENS A_FACT_PROD	LIB_INDICATEUR F_NUMERATEUR_CUMUL_A	-	-
11	F_DENOMINATEUR_CUMUL_G	A_SUIV_CONTRAT A_RELIEVE_MENSU(H) A_GES_CONTRAT_M A_PUBLIC_PROD A_RELIEVE_MENS A_MENS_NATION A_MENS A_FACT_PROD	F_DENOMINATEUR_CUMUL_G	-	
14	CODE_PLAT	D_ORG_COM_PLAT	CODE_PLAT	VARCHAR(10 CHAR)	-
15	LIBELLE_PLAT	D_ORG_COM_PLAT	LIBELLE_PLAT	VARCHAR(50 CHAR)	-
16	TRIGRAMME_PLAT	D_ORG_COM_PLAT	TRIGRAMME_PLAT	VARCHAR(10 CHAR)	-
17	LIBELLE_DIR	D_ORG_COM_DIR	CODE_DIR	VARCHAR(20 CHAR)	-
18	CODE_DIR	D_ORG_COM_DIR	LIBELLE_DIR	VARCHAR(20 CHAR)	-
19	CODE_DR	D_ORG_COM_DR	CODE_DR	VARCHAR(10 CHAR)	-
20	TRIGRAMME_DR	D_ORG_COM_DR	TRIGRAMME_DR	VARCHAR(10 CHAR)	-
21	LIBELLE_DR	D_ORG_COM_DR	LIBELLE_DR	VARCHAR(50 CHAR)	-
22	D_CODE_SECTEUR_TENSION	D_SECTEUR_TENS D_SECTEUR_TENS_PROD A_GES_CONTRAT_M A_PUBLIC_PROD A_RELIEVE_MENS A_MENS_NATION A_MENS A_FACT_PROD	D_SECTEUR_TENSION D_CODE_SECTEUR_TENSION_SGE D_CODE_SECTEUR_TENSION	-	
23	D_LIBELLE_SECTEUR_TENSION	D_SECTEUR_TENS D_SECTEUR_TENS_PROD A_GES_CONTRAT_M A_PUBLIC_PROD A_RELIEVE_MENS A_MENS_NATION A_MENS A_FACT_PROD	D_SECTEUR_TENSION_TARIF D_LIBELLE_SECTEUR_TENSION D_LIBELLE_SECTEUR_TENSION	-	
24	SECTION	D_SECTION_CALCULE D_SECTION_PROD D_SECTION A_GES_CONTRAT_M A_PUBLIC_PROD A_RELIEVE_MENSU(H) A_SUIV_CONTRAT A_RELIEVE_MENS A_FACT_PROD	D_CODE_SECTION D_CODE_SECTION D_SECTION_CALCULE	-	

Annexe 3 : Script de création d'indicateurs avec conventions de nommage

```

File Edit Selection Find View Goto Tools Project Preferences Help
1 ...
2 ...
3 ... CREATION DE L'indicateur RAW-den
4 ...
5 ... Prestataire : EGI
6 ... Version date : 26/09/2019
7 ... Version auteur : mlaure
8 ...
9 ... US-55 calcul d'indicateur de suivi de la qualité RED
10 ...
11 ...
12 ...
13 ...
14 ...
15 ...
16 SET ESCAPE `;
17 SET DEFINE OFF;
18 ALTER SESSION SET CURRENT_SCHEMA=OTM;
19 PROMPT SUPPRESSION DE L'INDICATEUR RAW-DEN ET CREATION
20 LIBRARY;
21 S_REQ_M VARCHAR2(32000);
22 S_REQ_H VARCHAR2(32000);
23
24 BEGIN
25
26 delete from tec_mm_indicateur
27 where d_id_Indicateur = 1037
28 and code_uniq_Indicateur = 'RAW-DEN'
29 and version = 1;
30
31 S_REQ_M:='select
32 #PERIODE as PERIODE,
33 #ID_IND as d_id_Indicateur,
34 '#DNI code_insee' AS niveau,
35 RED_FK_CODE_INSEE AS code_niveau,
36 '#ANGERS' AS processus,
37 COUNT(NULL AS numerateur) AS D_OBJET_01,
38 CAST(NULL AS NUMBER) AS D_OBJET_02,
39 CAST(NULL AS NUMBER) AS D_OBJET_03,
40 CAST(NULL AS NUMBER) AS D_OBJET_04,
41 CAST(NULL AS NUMBER) AS D_OBJET_05,
42 # ID DIM 01,
43 1 AS D_ID_DIM_02,
44 1 AS D_ID_DIM_03,
45 2 AS D_ID_DIM_04,
46 2 AS D_ID_DIM_05,
47 1 AS D_ID_DIM_06,
48 2 AS D_ID_DIM_07,
49 2 AS D_ID_DIM_08,
50 1 AS D_ID_DIM_09,
51 2 AS D_ID_DIM_10,
52 1 AS D_ID_DIM_11,
53 2 AS D_ID_DIM_12,
54 2 AS D_ID_DIM_13,
55 2 AS D_ID_DIM_14,
56 2 AS D_ID_DIM_15,
57 2 AS D_ID_DIM_16,
58 2 AS D_ID_DIM_17,
59 2 AS D_ID_DIM_18,
60 2 AS D_ID_DIM_19,
61 2 AS D_ID_DIM_20,
62 2 AS D_ID_DIM_21,
63 2 AS D_ID_DIM_22,
64 2 AS D_ID_DIM_23,
65 2 AS D_ID_DIM_27,
66 2 AS D_ID_DIM_24,
67 2 AS D_ID_DIM_25,
68 2 AS D_ID_DIM_26,
69 2 AS D_ID_DIM_28,
70 2 AS D_ID_DIM_29,
71 2 AS D_ID_DIM_30,
72 RED_FK_ATTENTE as numerateur,
73 COUNT(NULL AS numerateur) as denominateur,
74 cast(null as number) as f_numerateur_cumul_a,
75 cast(null as number) as f_denominateur_cumul_a,
76 cast(null as number) as f_numerateur_cumul_g,
77 cast(null as number) as f_denominateur_cumul_g
78
79 RECHERCHE_RED_RED
80 INNER JOIN PDM_PDM ON PDM.id_pdm = RED_FK_CODE_INSEE
81 INNER JOIN DMN_STATUTATION_SUP_SS ON SS.FK_ID_PDM = PDM.ID_PDM
82 WHERE DMN_STATUTATION_SS = 'ATTENTE'
83 AND SS.DATE_FIN_VALIDATION > last_day(to_date(#PERIODE, 'YYYYMM'))
84 AND SS.DATE_FIN_VALIDATION > last_day(to_date(#PERIODE, 'YYYYMM'))
85 AND RED.PERIODE=#PERIODE
86 ';
87
88 S_REQ_H:='';
89
90 insert into tec_mm_indicateur (
91 D_ID_INDICATEUR,
92 CODE_UNIQ_INDICATEUR,
93 LIB_INDICATEUR,
94 VERSION,
95 CODE_UNIQ_GROUPE_INDICATEUR,
96 CUMULABLE,
97 REQUETE_MENSUELLE,
98 REQUETE_TRIMESTRIELLE,
99 REQUETE_ANNUELLE,
100 DATE_DEBUT_VALID,
101 DATE_FIN_VALID,
102 ACTIF_MENSUEL,
103 ACTIF_TRIMESTRIEL,
104 ACTIF_ANNUEL,
105 COMMENTAIRE,
106 BASE_SOURCE,
107 TECH_DATE_CREA,
108 ORDRE_INDICATEUR,
109 LIB_UNIQ_INDICATEUR_NUM,
110 CODE_UNIQ_INDICATEUR_DEN,
111 CODE_UNIQ_INDICATEUR_SOMME,
112 ORGA_RECEPTION,
113 PUBLICATION,
114 PUBLICATION,
115 ACTIVITE,
116 TYPE_VALEUR,
117 SOURCE,
118 ADDECTIVITE_ORGANISATIONNELLE,
119 DR_VERS_AGENCE)
120 values ('1037','RAW-DEN','Nombre de publications RAW attendu','1','1001','0',S_REQ_M,null,S_REQ_H,to_date('01/01/2018','DD/MM/YYYY'),
121 to_date('31/12/2999','DD/MM/YYYY'),'0','N','N',null,'Dni',to_date('26/09/2019','DD/MM/YYYY'),null,null,null,'INSEE',null,
122 '0','AW','Valeur mensuelle','RAW',null,null);
123
124 END ;
125 /
126
127 commit;
128
129 SET ESCAPE OFF;
130

```

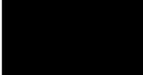
Line 1, Column 1

Tab Size: 4

SQL

Annexe 4 : fiche de test unitaires avec tests de non régrression





Résultat	
Etat	Final
OK	24
KO	0
INTESTABLE	0
TOTAL	24

Round : 1
Testé par : MLAU Charge consommée : Sécurité Date Etat

Document de référence	Élement testé	Prérequis	Description du test	Données d'entrée	Résultat attendu	Alimentation	
						Réultat Objetif si KO / Commentaire	Sécurité
							Date
							Etat
1	Script SQL création table_OL_4_CM42283_CREATE_TABLE_COLLECTE_RED_MENSUEL.q		lancer la commande sqlplus sansoublier la commande grant_all_on_synonyme.ksh		Le script s'exécute sans erreur et la table est créée dans DWH.COLLECTE_RED_MENSUEL	Critique	27/03/2019 OK
2	Script SQL création table_02_dwh_US_123456789_Joiner_info_TECH_mme_INDICATEU_R_R4W-RED.q		lancer la commande sqlplus sansoublier la commande grant_all_on_synonyme.ksh		Le script s'exécute sans erreur et la table est créée dans DWH.public_RED_MENSUEL	Critique	27/03/2019 OK
3	Script SQL insertion d'indicateur _01_dwh_US_123456789_Joiner_info_TECH_mme_INDICATEU_R_R4W-RED.q		lancer la commande sqlplus sansoublier la commande grant_all_on_synonyme.ksh		Le script s'exécute sans erreurs et la ligne est insérée select * from tech_mme_indicateur where d_id_indicator = 1036;	Critique	27/03/2019 OK
4	Script SQL insertion d'indicateur _04_dwh_US_123456789_Joiner_info_TECH_mme_INDICATEU_R_R4W-RED.q		lancer la commande sqlplus sansoublier la commande grant_all_on_synonyme.ksh		Le script s'exécute sans erreurs et la ligne est insérée select * from tech_mme_indicateur where d_id_indicator = 1037;	Critique	27/03/2019 OK
5	05_dwh_US_123456789_Joiner_info_TECH_mme_INDICATEU_R_R4W.q		lancer la commande sqlplus sansoublier la commande grant_all_on_synonyme.ksh		Le script s'exécute sans erreurs et la ligne est insérée select * from tech_mme_indicateur where d_id_indicator = 1038;	Critique	27/03/2019 OK
6	06_dwh_US_123456789_Joiner_info_TECH_mme_INDICATEU_R_R4W-RED-9M.q		lancer la commande sqlplus sansoublier la commande grant_all_on_synonyme.ksh		Le script s'exécute sans erreurs et la ligne est insérée select * from tech_mme_indicateur where d_id_indicator = 1039;	Critique	27/03/2019 OK
7	Script SQL insertion d'indicateur _01_dwh_US_123456789_Joiner_info_TECH_mme_INDICATEU_R_R40-RED-DEB.q		lancer la commande sqlplus sansoublier la commande grant_all_on_synonyme.ksh		Le script s'exécute sans erreurs et la ligne est insérée select * from tech_mme_indicateur where d_id_indicator = 1040;	Critique	27/03/2019 OK
8	Script SQL insertion d'indicateur _08_dwh_US_123456789_Joiner_info_TECH_mme_INDICATEU_R_R40-RED.q		lancer la commande sqlplus sansoublier la commande grant_all_on_synonyme.ksh		Le script s'exécute sans erreurs et la ligne est insérée select * from tech_mme_indicateur where d_id_indicator = 1041;	Critique	27/03/2019 OK
9	Script SQL insertion d'indicateur _03_dwh_US_123456789_Joiner_info_TECH_mme_INDICATEU_R_R4H-RED-9M.q		lancer la commande sqlplus sansoublier la commande grant_all_on_synonyme.ksh		Le script s'exécute sans erreurs et la ligne est insérée select * from tech_mme_indicateur where d_id_indicator = 1042;	Critique	27/03/2019 OK
10	Script SQL insertion d'indicateur _10_dwh_US_123456789_Joiner_info_TECH_mme_INDICATEU_R_R4H-RED-DEB.q		lancer la commande sqlplus sansoublier la commande grant_all_on_synonyme.ksh		Le script s'exécute sans erreurs et la ligne est insérée select * from tech_mme_indicateur where d_id_indicator = 1043;	Critique	27/03/2019 OK
11	11_dwh_US_123456789_Joiner_info_TECH_mme_INDICATEU_R_R4H-RED.q		lancer la commande sqlplus sansoublier la commande grant_all_on_synonyme.ksh		Le script s'exécute sans erreurs et la ligne est insérée select * from tech_mme_indicateur where d_id_indicator = 1044;	Critique	27/03/2019 OK
12	Script SQL insertion d'indicateur _12_dwh_US_123456789_Joiner_info_TECH_mme_INDICATEU_R_R4M-RED-9M.q		lancer la commande sqlplus sansoublier la commande grant_all_on_synonyme.ksh		Le script s'exécute sans erreurs et la ligne est insérée select * from tech_mme_indicateur where d_id_indicator = 1045;	Critique	27/03/2019 OK
13	Script SQL insertion d'indicateur _13_dwh_US_123456789_Joiner_info_TECH_mme_INDICATEU_R_R4M-RED-DEB.q		lancer la commande sqlplus sansoublier la commande grant_all_on_synonyme.ksh		Le script s'exécute sans erreurs et la ligne est insérée select * from tech_mme_indicateur where d_id_indicator = 1046;	Critique	27/03/2019 OK
14	Script SQL insertion d'indicateur _14_dwh_US_123456789_Joiner_info_TECH_mme_INDICATEU_R_R4M-RED.q		lancer la commande sqlplus sansoublier la commande grant_all_on_synonyme.ksh		Le script s'exécute sans erreurs et la ligne est insérée select * from tech_mme_indicateur where d_id_indicator = 1047;	Critique	27/03/2019 OK
					Duré chargement des fichiers		
	Les fichiers se récupèrent bien dans le sas	Déposer les fichiers de test dans echanges/R4W via un client FTP comme Filezilla	Pour la public: recip_file_ksh_R4W_public		La récupération se déroule sans erreur et les bons fichiers sont placés dans parentéading(R4W_public)	Critique	27/03/2019 OK
		Déposer les fichiers de test dans echanges/R4W	Pour la collecte: recip_file_ksh_R4W_COLLECTE		La récupération se déroule sans erreur et les bons fichiers sont placés dans parentéading(R4W_collecte)	Critique	27/03/2019 OK
	Test des fichiers_P4V_20190101_20190201L20190412205306.csv ,R4W_20190101_20190201_20190412205410.csv ,R4W_20190301_20190401_20190412205052.csv						
	Alimentation de la table public_RED_MENSUEL		alim.ksh nppg R4W_public -p 201901 alim.ksh nppg R4W_public -p 201902 alim.ksh nppg R4W_public -p 201903		L'alimentation se déroule sans erreurs et des données sont présentes dans la table public_RED_MENSUEL select * from public_RED_MENSUEL	Critique	27/03/2019 OK
			Vérification du remplissage de la bonne partition select * from public_RED_MENSUEL PARTITION(PARTITION_201901);		Données présente dans la partition	Critique	27/03/2019 OK
			Vérification du remplissage de la bonne partition select * from public_RED_MENSUEL PARTITION(PARTITION_201902);		Données présente dans la partition	Critique	27/03/2019 OK
			Vérification du remplissage de la bonne partition select * from public_RED_MENSUEL PARTITION(PARTITION_201903);		Données présente dans la partition	Critique	27/03/2019 OK
			Vérification du remplissage de la bonne partition select * from public_RED_MENSUEL PARTITION(PARTITION_201904);		Aucunes données présente dans la partition	Critique	27/03/2019 OK
	Alimentation de la table COLLECTE_RED_MENSUEL	Test du fichier Export_R4V_Tr_Collecte_201906.csv	alim.ksh nppg R4W_COLLECTE -p 201905		L'alimentation se déroule sans erreurs et des données sont présentes dans la table COLLECTE_RED_MENSUEL	Critique	27/03/2019 OK
			Vérification du remplissage de la bonne partition select * from public_COLLECTE_RED_MENSUEL PARTITION(PARTITION_201905);		Données présente dans la partition	Critique	27/03/2019 OK
			Vérification du remplissage de la bonne partition SELECT * FROM COLLECTE_RED_MENSUEL PARTITION(PARTITION_201906)		Pas de données dans la partition	Critique	27/03/2019 OK

				Indicateurs		
	Alimentation de l'indicateur R4W R4W_1NUM R4W R4W	Données présentes dans la table public_RED_MENSUEL	Mensuel: alim.ksh dm.mat.indic.m->201905-1036,1037,1038	L'alimentation se déroule sans erreurs et les données sont présentes dans la table A_MAT_M select * from A_MAT_M where d_id_indicateur in (1036,1037,1038) and d_periode =201905; select DISTINCT d_id_indicateur from A_MAT_M where d_id_indicateur in (1036,1037,1038) and d_periode =201905;	Critique	27/09/2019
					OK	
	Alimentation de l'indicateur R4HQ R4Q-RED 1NUM R4Q R4Q-RED	Données présentes dans la table COLLECTE_RED_MENSUEL	Mensuel: alim.ksh dm.mat.indic.m->201905-1033,1040,1041	L'alimentation se déroule sans erreurs et les données sont présentes dans la table A_MAT_M select * from A_MAT_M where d_id_indicateur in (1033,1040,1041) and d_periode =201905; select DISTINCT d_id_indicateur from A_MAT_M where d_id_indicateur in (1033,1040,1041) and d_periode =201905;	Critique	27/09/2019
					OK	
	Alimentation de l'indicateur R4H R4H_1NUM R4H R4H-RED	Données présentes dans la table COLLECTE_RED_MENSUEL	Mensuel: alim.ksh dm.mat.indic.m->201905-1042,1043,1044	L'alimentation se déroule sans erreurs et les données sont présentes dans la table A_MAT_M select * from A_MAT_M where d_id_indicateur in (1042,1043,1044) and d_periode =201905; select DISTINCT d_id_indicateur from A_MAT_M where d_id_indicateur in (1042,1043,1044) and d_periode =201905;	Critique	27/09/2019
					OK	
	Alimentation de l'indicateur R4M R4M_1NUM R4M R4M-RED	Données présentes dans la table COLLECTE_RED_MENSUEL	Mensuel: alim.ksh dm.mat.indic.m->201905-1045,1046,1047	L'alimentation se déroule sans erreurs et les données sont présentes dans la table A_MAT_M select * from A_MAT_M where d_id_indicateur in (1045,1046,1047) and d_periode =201905; select DISTINCT d_id_indicateur from A_MAT_M where d_id_indicateur in (1039,1040,1041) and d_periode =201905;	Critique	27/09/2019
					OK	

Script SQL

```

01_dtm_us123456789_CREATE_TABLE_COLLECTE_RED_MENSUEL.sql
02_dwh_US-123456789_CREATE_TABLE_public_RED_MENSUEL.sql
03_dtm_US-123456789_insert_into_TECH_mm_INDICATEUR_R4W-1NUM.sql
04_dtm_US-123456789_insert_into_TECH_mm_INDICATEUR_R4W-DEN.sql
05_dtm_US-123456789_insert_into_TECH_mm_INDICATEUR_R4W.sql
06_dtm_US-123456789_insert_into_TECH_mm_INDICATEUR_R4Q-RED-1NUM.sql
07_dtm_US-123456789_insert_into_TECH_mm_INDICATEUR_R4Q-RED-DEN.sql
08_dtm_US-123456789_insert_into_TECH_mm_INDICATEUR_R4Q-RED.sql
09_dtm_US-123456789_insert_into_TECH_mm_INDICATEUR_R4H-RED-1NUM.sql
10_dtm_US-123456789_insert_into_TECH_mm_INDICATEUR_R4H-RED-DEN.sql
11_dtm_US-123456789_insert_into_TECH_mm_INDICATEUR_R4H-RED.sql
12_dtm_US-123456789_insert_into_TECH_mm_INDICATEUR_R4M-RED-1NUM.sql
13_dtm_US-123456789_insert_into_TECH_mm_INDICATEUR_R4M-RED-DEN.sql
14_dtm_US-123456789_insert_into_TECH_mm_INDICATEUR_R4M-RED.sql

```

UNIX

```

mg/config/alim_mg.cfg
mg/batch/alim.ksh
mg/batch/pre_trt_unitaire_mg.ksh
mg/batch/recup_fic.ksh
ref/config/alim_seq.csv
mg/config/domaines_mg.csv
Commun/config/nom_flux_module.csv
Créer dossier R4W dans echanges/confr

```

Informatica

Workflow	W_DWH_DC_MENS_COLLE
Mapping	m_DWH_DC_MENS_COLLE
Workflow	W_DWH_CDC_MENS_PUBLIC
Mapping	m_DWH_DC_MENS_PUBLIC

Annexe 5 : MODOP

The screenshot shows the first page of a document titled "Modop Crédation groupe indicateur". The page features a network diagram at the top left and a table of contents on the right. The table of contents includes sections like "PRESENTATION ET RÉSUMÉ", "OBJECTIFS", "PRÉREQUIS", and "Création de la vue et des triggers". Below the title, there's a section for "Création et suivi des modifications" with a "Historique du document" table.

Version	Date	Rédacteur de la version	Statut	Commentaires
V0.1	01/08/19	Maxime LAURIN	Draft	
V1.0	01/08/19	confidential	Release	Relecture et approbation
V1.1	20/09/19	confidential	Release	Modif de l'ordre des colonnes de la table A. Qui doit également être pareil avec celles de la vue.

© 2019 CGI INC. Inténe

The screenshot displays two sections of the document: "1 PRÉSENTATION ET RÉSUMÉ" and "2 PRÉREQUIS".

1 PRÉSENTATION ET RÉSUMÉ

1.1 Objectifs

- L'objectif de ce document est d'aider le développeur à créer un groupe d'indicateur dans MySQL.

2 PRÉREQUIS

- Les composants logiciels, décrits ci-dessous, doivent être installés et configurés (paramètres, création des dépôts git, accès aux serveurs) sur le poste développeur.
Se référer au [guide d'installation du poste de développeur](#).

Logiciels	Version	Commentaires
Sql Developer		
Notepad ++		
Tortoisegit		
Firefox		En fonction de la mise à disposition du client
Putty		
Superputty		
Git		
informatics		Facultatif
KeePass		

© 2019 CGI INC. 1 Inténe

© 2019 CGI INC. 2 Inténe

3 Création du nouveau groupe indicateur dans tech_mm

- Cette première étape consiste à créer notre nouveau groupe d'indicateurs

Tout d'abord, se référer aux PO pour choisir le nom du groupe.
Ensuite, insérer 3 nouvelles lignes correspondant aux sous-groupes :

- La Requête
- La somme
- Le taux

Voici un exemple plus explicite sous forme de tableau ainsi qu'un script Générique à adapter suivant les besoins :

ID GROUPE INDICATEUR	NOM DATAMART	TECH_CREA	ORDRE_CALCUL	TYPE INDICATEUR	LIB GROUPE INDICATEUR
13001	<nom_datamart>	<date de creation>	1	R	<nom_datamart> requête
13002	<nom_datamart>	<date de creation>	2	S	<nom_datamart> SOMME
13003	<nom_datamart>	<date de creation>	3	T	<nom_datamart> TAUX

L'ID groupe indicateur se choisit en fonction du précédent groupe créé, en effet on augmente d'un millier à chaque fois qu'on crée un nouveau groupe :

Exemple : Si le précédent groupe possède `des_id` de 12000, le notre sera à 13000.

L'ordre de calcul doit commencer par la requête, la somme puis le taux.

Les types d'indicateurs :

- R pour une Requête
- S pour une Somme
- T pour un Taux

Le script tout prêt à adapter :

```
insert into TECH_MN_GROUPE_INDICATEUR_XXXXXXsql
```

© 2019 CGI INC. 3 Interna

4 Création de la table Agrégée

Maintenant que nous avons créé le nouveau groupe, il faut créer une table A pour récupérer les agrégations de nos indicateurs :

La table Agrégée doit contenir les champs de type number de la table F ainsi que d'autres champs obligatoires :

K. Nom	Type
D_PERIODE	number not null
D_PERIODE_CALCUL	number
D_ID_INDICATEUR	number
D_ID_DIM	number
D_ID_DIM_03	number
D_ID_DIM_04	number
D_ID_DIM_05	number
D_ID_DIM_06	number
D_ID_DIM_07	number
D_ID_DIM_08	number
D_ID_DIM_09	number
D_ID_DIM_10	number
D_ID_DIM_11	number
D_ID_DIM_12	number
D_ID_DIM_13	number
D_ID_DIM_14	number
D_ID_DIM_15	number
D_ID_DIM_16	number
D_ID_DIM_17	number
D_ID_DIM_18	number
D_ID_DIM_19	number
D_ID_DIM_20	number
D_ID_DIM_21	number
D_ID_DIM_22	number
D_ID_DIM_23	number
D_ID_DIM_24	number
D_ID_DIM_25	number
D_ID_DIM_26	number
D_ID_DIM_27	number
D_ID_DIM_28	number
D_ID_DIM_29	number
D_ID_INDICATEUR	number
F_NUMERATEUR	number
F_DENOMINATEUR	number
F_NUMERATEUR_CUMUL_A	number
F_DENOMINATEUR_CUMUL_A	number
F_NUMERATEUR_CUMUL_G	number
F_DENOMINATEUR_CUMUL_G	number
TECH_DATE_CREA	date
TECH_ID_TRT_UNITAIRE	number

On peut voir la structure de la table Générique dans le mapping `m_DTM_A_Mn_IND_SEQUETE` dans Informatica PowerCenter Designer.

Notre table doit avoir la structure de la target `structure_table_a_com_moteur_regle`.

Le Script à adapter est disponible ici :

create_table_A_XXXXXXX.sql

© 2019 CGI INC. 4 Interna

5 Création de la vue et des triggers

La table agrégée étant créée, nous pouvons maintenant créer sa vue associée ainsi que son trigger afin de faire la passerelle entre la table F et la table A.

La vue doit contenir tous les champs de la table A mais avec quelques substitutions.

Nous allons prendre comme modèle la `structure_vue_mm_moteur_regle` pour créer notre vue.

- Pour le script, le select de la vue doit obligatoirement contenir tous les champs de la table A.
- Elle doit aussi contenir tous les `d_id_dim`, de 1 jusqu'à 30

Un script tout prêt à adapter est disponible ci-dessous pour la création :

```
CREATE_VIEW_VUE_TABLE_A_XXXXXXXsql
```

© 2019 CGI INC. 5 Interna

6 Modification ksh

- Nous avons 2 fichiers à modifier : Il faut pour cela accéder au dossier git Unix (branche develop).

Le second fichier à modifier est `alm_dim_mm_indic`.

- Faire un git pull sur le dépôt unix(branche develop)
- Ouvrir le fichier `alm_dim_mm_indic` avec notepad++ dans dmrbatchv
- Chercher la ligne <`# appels intencts`> et ajouter notre groupe précédé de <`-indc`>
- Chercher la ligne <`# appels autorisés`> et ajouter dans l'écho notre groupe
- Enregistrer le document et ouvrir le fichier `alm_dim_mm_indic` dans le même dossier
- Chercher la ligne <`# appels autorisés`> et ajouter dans le <`#app`> notre appel de groupe

© 2019 CGI INC. 6 Interna

CGI



Chercher la ligne « # détermination du nom des tables et des vues associées ».

Si notre groupe est calculé pour le menu, Ajouter notre nom de départ et sa table Annexée dans le cas 2.

6 Si le groupe est pour une table Holdemdata, l'ajouter dans le cas 1. Sinon, l'ajouter aux 2 si on a 2 tables agrégées basses et hautes.

7 Le groupe est maintenant ajouté. Il ne nous reste plus qu'à commit et push nos modifications.

Notre groupe est désormais prêt à être utilisé. Il ne reste plus qu'à créer nos indicateurs pour pouvoir pleinement le tester.
Je vous invite donc à suivre ce [MOOP](#) pour la création des indicateurs.

© 2019 CGI INC. 7 Interno

CGI



ANNEXES

© 2019 CGI INC. 8 Interno

Annexe 6 : Bon de livraison

Annexe 7 : Modification et versionning d'un indicateur

```
-- CREATION DE l'indicateur PAC2M
-- 
-- Prestataire : CGI
-- Version date : 26/09/2019
-- Version auteur : MLAU
-- 
-- SCG-89 [US] ARGCE : Calcul d'indicateur de suivi courbe de charge
-- 
-- 

SET ESCAPE ^;
SET DEFINE OFF;
ALTER SESSION SET CURRENT_SCHEMA=DTM;
PROMPT SUPPRESSION DE L INDICATEUR PAC2M ET CREATION
declare
S_REQ_M VARCHAR2(32000);
S_REQ_H VARCHAR2(32000);

BEGIN
UPDATE TECH_MR_INDICATEUR
SET DATE_FIN_VALID = to_date('31/12/2018', 'DD/MM/YYYY')
where d_id_indicateur = 1038
and code_uniq_indicateur = 'PAC2M'
AND version = 1 ;

delete from tech_mr_indicateur
where d_id_indicateur = 1038
and code_uniq_indicateur = 'PAC2M'
and version =2;

S_REQ_M:='';
S_REQ_H:='';

Insert into dtm.TECH_MR_INDICATEUR (
D_ID_INDICATEUR,
CODE_UNIQ_INDICATEUR,
LIB_INDICATEUR,
VERSION,
FK_D_ID_GROUPE_INDICATEUR,
CUMULABLE,
REQUETE_MENSUELLE,
REQUETE_TRIMESTRIELLE,
REQUETE_HEBDO,
DATE_DEB_VALID,
DATE_FIN_VALID,
ACTIF_MENSUEL,
ACTIF_TRIMESTRIEL,
ACTIF_HEBDO,
COMMENTAIRE,
BASE_SOURCE,
TECH_DATE_CREA,
ORDRE_CALCUL_REQUETE,
CODE_UNIQ_INDICATEUR_NUM,
CODE_UNIQ_INDICATEUR_DEN,
CODE_UNIQ_INDICATEUR_SOMME,
ORGANISATION,
UNITE,
PUBLICATION,
ACTIVITE,
TYPE_VALEUR,
SOURCES,
ADDITIONNITE_ORGANISATIONNELLE,
DR_VERS_AGENCE)
values ('1038','PAC2M', 'TAUX des courbes de charges au marché ','2', '1003','0',S_REQ_M, null,S_REQ_H, to_date('01/01/2019', 'DD/MM/YYYY'),
to_date('31/12/2999', 'DD/MM/YYYY'), '0', 'N', 'N', null, 'DWH', to_date('26/09/2019', 'DD/MM/YYYY'), null, 1036 ,1037, null, 'INSEE', null, 'N', 'ACM', 'Valeur_mensuelle', 'PAC2M', null, null);
END ;
/
commit;
SET ESCAPE OFF;
```

Annexe 8 : Outil GOJIRA

[US] : Calcul d'indicateur de suivi de la qualité

Informations

- Type: User Story
- Priorité: Moyenne
- Composants: Aucune
- Déroulées:
- Équipe:
- Estimation: 12
- Etat d'exigence: V9.8 - UNCOUPLED

Etat: EN RÉALISATION (Afficher le workflow)

Résolution: Non résolu

Version(s) corrélée(s): V9.8

Personnes

- Réponsable: Me l'affecter
- Rapporteur: Voter pour ce ticket
- Voter: Gérer les observateurs
- Démarrer l'observation di

Dates

- Création: 27/08/19 16:39
- Mise à jour: 08/10/19 15:56

Suivi temporel

- Estimé:
- Restant:
- Consigné:
- Inclure les sous-tâches

Agile

- Active Sprint:
- Terminé Sprint:
- Afficher sur le Tableau

Description

En tant qu'interlocuteur CRS

Je souhaite avoir à disposition les indicateurs de suivi de la qualité de mise à disposition des courbes de charges des clients

Afin de proposer des indicateurs de régulation, initiative sur les services de courbes de charges.

Pour cela, il faut:

- Intégrer le flux de suivi de la qualité (implémentation de publication des courbes de charges: mises en dépôt sur un dépôt interne (à créer))
- Intégrer le flux de suivi de publication des courbes de charges dans les cédes (Avec un fil mensuel déposé sur un dépôt interne (à créer))
- Calculer les nouveaux indicateurs de suivi des courbes de charges et les intégrer sur les remontées dans l'univers

Couverture du Test

Aucun Test trouvé pour tester l'Exigence.

Fiches jointes

(G) Glissez-déposez des fichiers pour les joindre ou parcourir.

Liens des tickets

est bloqué par EN INSTRUCT...

Sous-tâches

- checkeur
- BL feuille de route
- livraison en récette en NCO
- TUTU
- création des indicateurs
- modification des fichiers de config
- ish pour lancement sur VKF
- créer les mappings d'alimentation

Statut	Nom attribué	Progression
A FAIRE	Non attribuée	0%
A FAIRE	MAXIME LAURIN	0%
TERMINÉ	MAXIME LAURIN	0%
TERMINÉ	MAXIME LAURIN	100%
EN COURS	MAXIME LAURIN	100%
TERMINÉ	MAXIME LAURIN	100%

<http://gojira.enedis.fr/secure/MyJiraHome.jspa>

Annexe 9 : Script de conception de base de donnée

```
1 drop DATABASE IF EXISTS progiplus;
2 CREATE DATABASE progiplus;
3 use progiplus;
4
5 create table civilite(
6     id_civilite int(10) PRIMARY KEY AUTO_INCREMENT,
7     libelle VARCHAR(50) NOT NULL
8 );
9 CREATE TABLE ville(
10    id_ville INT(10) PRIMARY KEY AUTO_INCREMENT,
11    nom_ville VARCHAR(50) NOT NULL,
12    cp_ville VARCHAR(5)
13 );
14 CREATE TABLE adresse(
15     id_adresse INT(10) PRIMARY KEY AUTO_INCREMENT,
16     ligne1 VARCHAR(50) NOT NULL,
17     ligne2 VARCHAR(50),
18     id_ville INT NOT NULL,
19     CONSTRAINT fk_id_ville FOREIGN KEY(id_ville) REFERENCES ville(id_ville)
20 );
21 CREATE TABLE client(
22     id_client INT(10) PRIMARY KEY AUTO_INCREMENT,
23     code_client VARCHAR(25) NOT NULL UNIQUE,
24     raison_sociale VARCHAR(150),
25     actif BOOL NOT NULL,
26     id_adresse_facturation INT(10),
27     CONSTRAINT fk_client_adresse FOREIGN KEY(id_adresse_facturation) REFERENCES adresse(id_adresse)
28 );
29 CREATE TABLE contact(
30     id_contact INT(10) PRIMARY KEY AUTO_INCREMENT,
31     nom VARCHAR(30) NOT NULL,
32     prenom VARCHAR(30) NOT NULL,
33     service VARCHAR(50),
34     id_client INT NOT NULL,
35     id_civilite INT NOT NULL,
36     CONSTRAINT fk_id_client FOREIGN KEY(id_client) REFERENCES client(id_client),
37     CONSTRAINT fk_id_civilite FOREIGN KEY(id_civilite) REFERENCES civilite(id_civilite)
38 );
39 create TABLE type_moyen_comm(
40     id_type_moyen_comm int(10) PRIMARY KEY AUTO_INCREMENT,
41     libelle VARCHAR(50)NOT NULL
42 );
43 CREATE TABLE moyen_comm(
44     id_mcomm INT (10) PRIMARY KEY AUTO_INCREMENT,
45     valeur VARCHAR(50) NOT NULL,
46     id_type_moyen_comm INT(10) NOT NULL,
47     CONSTRAINT fk_id_type_moyen_comm FOREIGN KEY(id_type_moyen_comm) REFERENCES type_moyen_comm(id_type_moyen_comm)
48 );
49 CREATE TABLE contact_comm(
50     id_contact_comm INT(10) PRIMARY KEY AUTO_INCREMENT,
51     id_contact INT NOT NULL,
52     id_mcomm INT NOT NULL,
53     CONSTRAINT fk_id_contact FOREIGN KEY(id_contact) REFERENCES contact(id_contact),
54     CONSTRAINT fk_id_mcomm FOREIGN KEY(id_mcomm) REFERENCES moyen_comm(id_mcomm)
55 );
56 CREATE TABLE liste_adresse(
57     id_liste_adresse INT(10) PRIMARY KEY AUTO_INCREMENT,
58     libelle VARCHAR(50) NOT NULL,
59     actif BOOL NOT NULL,
60     id_client INT(10) NOT NULL,
61     id_adresse INT(10) NOT NULL,
62     CONSTRAINT fk_id_client_l_adresse FOREIGN KEY(id_client) REFERENCES client(id_client),
63     CONSTRAINT fk_id_adresse_l_adresse FOREIGN KEY(id_adresse) REFERENCES adresse(id_adresse)
64 );
65 CREATE TABLE tva(
66     id_tva int (10) PRIMARY KEY AUTO INCREMENT,
```

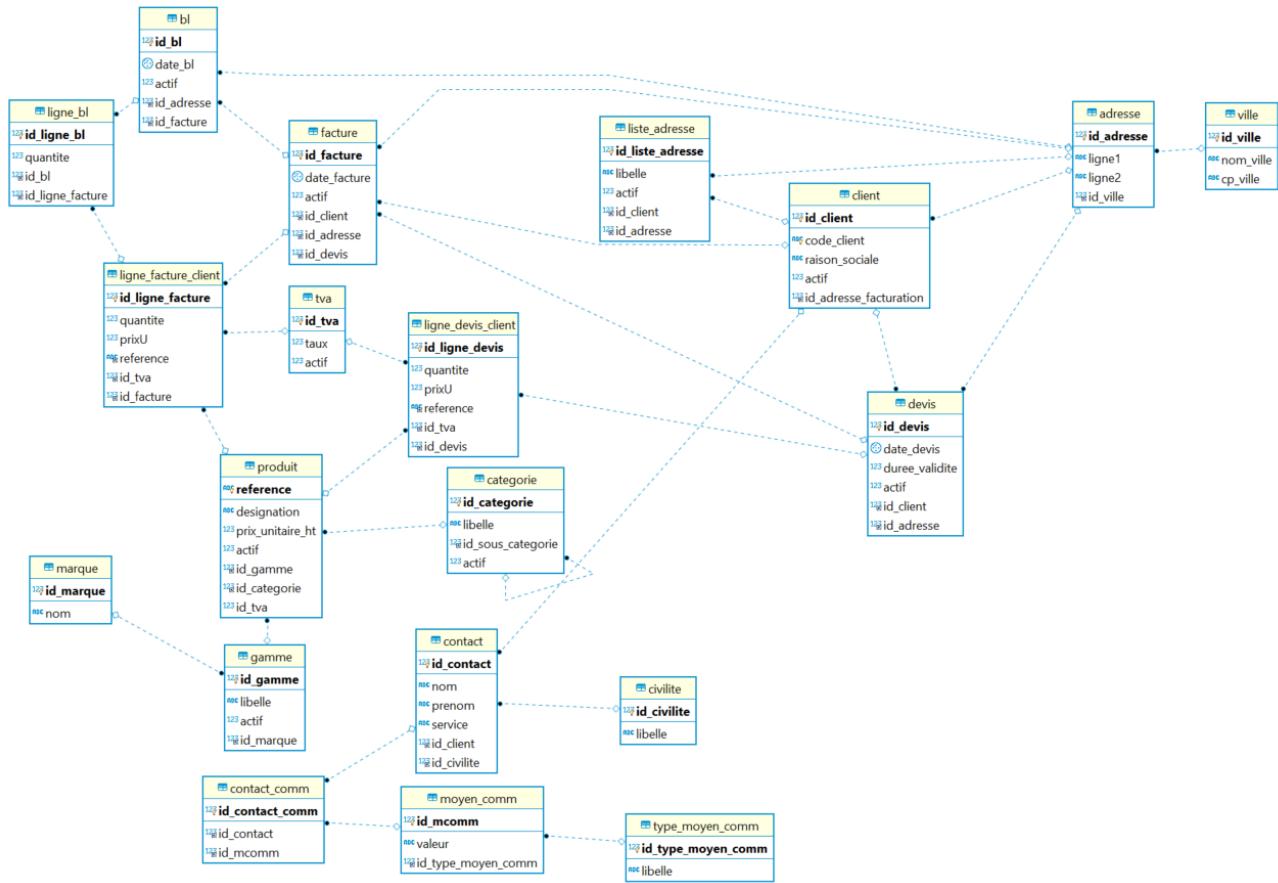
```

67     taux FLOAT(4,2) NOT NULL,
68     actif BOOL NOT NULL,
69     CONSTRAINT chk_taux CHECK (taux > 0 )
70   );
71 CREATE TABLE categorie(
72   id_categorie INT(10) PRIMARY KEY AUTO_INCREMENT,
73   libelle VARCHAR(50) NOT NULL,
74   id_sous_categorie int(10),
75   actif BOOL NOT NULL,
76   CONSTRAINT fk_id_sous_categorie FOREIGN KEY(id_sous_categorie) REFERENCES categorie(id_categorie)
77 );
78 CREATE TABLE marque(
79   id_marque INT(10) PRIMARY KEY AUTO_INCREMENT,
80   nom VARCHAR(100) NOT NULL
81 );
82 CREATE TABLE gamme(
83   id_gamme INT(10) PRIMARY KEY AUTO_INCREMENT,
84   libelle VARCHAR(50) NOT NULL,
85   actif BOOL NOT NULL,
86   id_marque INT(10) NOT NULL,
87   CONSTRAINT fk_id_marque FOREIGN KEY(id_marque) REFERENCES marque(id_marque)
88 );
89 CREATE TABLE produit(
90   reference VARCHAR(50) PRIMARY KEY,
91   designation VARCHAR(50) NOT NULL,
92   prix_unitaire_ht FLOAT(6,2) NOT NULL,
93   actif BOOL NOT NULL,
94   id_gamme INT(10) NOT NULL,
95   id_categorie INT NOT NULL,
96   id_tva INT NOT NULL,
97   CONSTRAINT fk_id_gamme FOREIGN KEY(id_gamme) REFERENCES gamme(id_gamme),
98   CONSTRAINT fk_id_categorie_produit FOREIGN KEY(id_categorie) REFERENCES categorie(id_categorie),
99   CONSTRAINT chk_prixUHT CHECK (prix_unitaire_ht > 0 )
100 );
101
102 CREATE TABLE devis(
103   id_devis INT(10) PRIMARY KEY AUTO_INCREMENT,
104   date_devis DATE NOT NULL,
105   duree_validite INT(3) NOT NULL,
106   actif BOOL NOT NULL,
107   id_client INT(10) NOT NULL,
108   id_adresse INT(10) NOT NULL,
109   CONSTRAINT chk_validite CHECK (duree_validite > 0 ),
110   CONSTRAINT fk_id_client_devis FOREIGN KEY(id_client) REFERENCES client(id_client),
111   CONSTRAINT fk_id_adresse_devis FOREIGN KEY(id_adresse) REFERENCES adresse(id_adresse)
112 );
113
114 CREATE TABLE facture(
115   id_facture INT(10) PRIMARY KEY AUTO_INCREMENT,
116   date_facture DATE NOT NULL,
117   actif BOOL NOT NULL,
118   id_client INT(10) NOT NULL,
119   id_adresse INT(10) NOT NULL,
120   id_devis INT(10),
121   CONSTRAINT fk_id_client_facture FOREIGN KEY(id_client) REFERENCES client(id_client),
122   CONSTRAINT fk_id_adresse_facture FOREIGN KEY(id_adresse) REFERENCES adresse(id_adresse),
123   CONSTRAINT fk_id_devis FOREIGN KEY(id_devis) REFERENCES devis(id_devis)
124 );
125 CREATE TABLE ligne_devis_client(
126   id_ligne_devis INT(10) PRIMARY KEY AUTO_INCREMENT,
127   quantite FLOAT(5,2) NOT NULL,
128   prixU FLOAT(6,2) NOT NULL,
129   reference VARCHAR(50) NOT NULL,
130   id_tva INT(10) NOT NULL,
131   id_devis INT(10) NOT NULL,

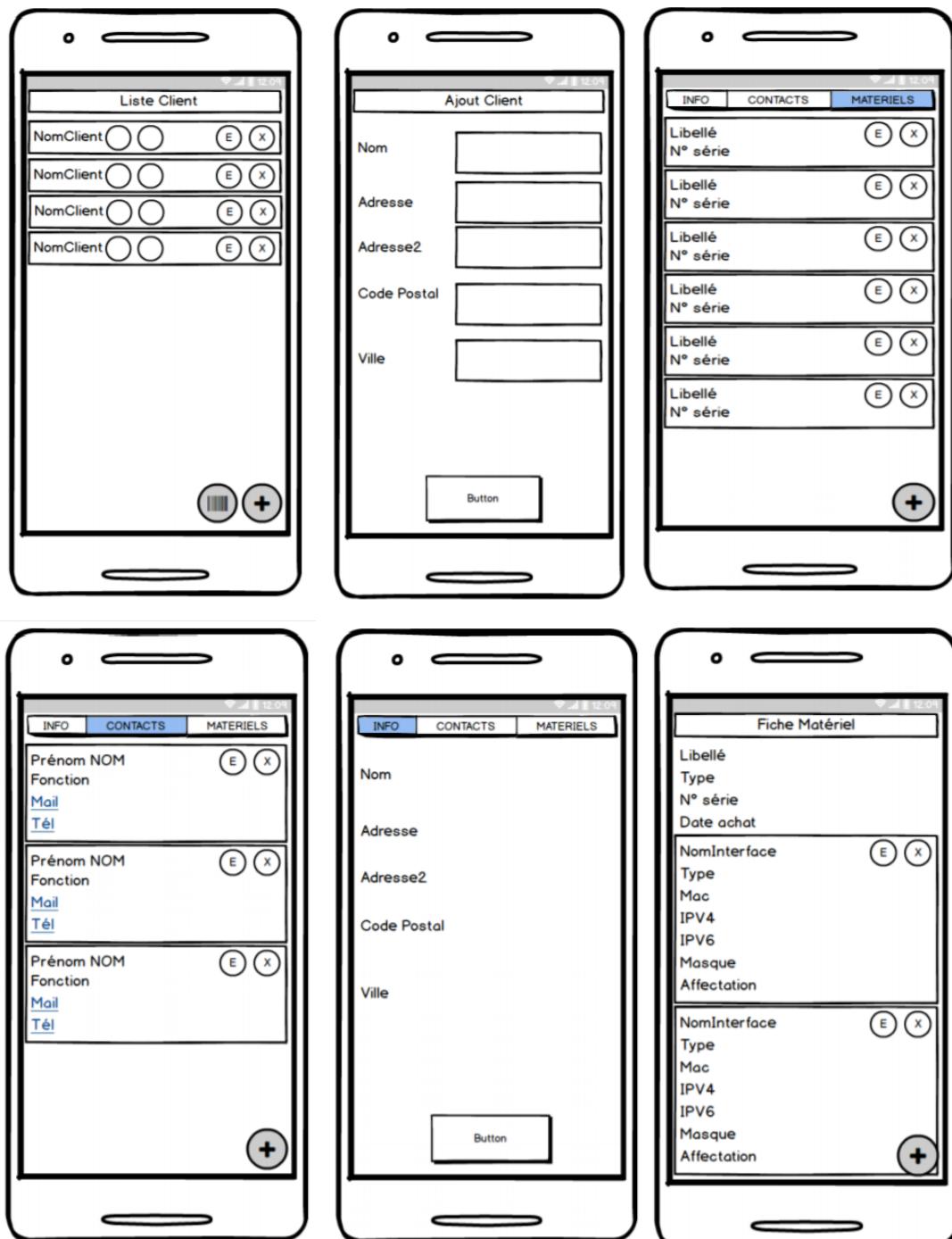
```

```
132     CONSTRAINT chk_quantiter CHECK (quantite > 0 ),]
133     CONSTRAINT fk_reference_devis FOREIGN KEY(reference) REFERENCES produit(reference),
134     CONSTRAINT fk_devis_tva FOREIGN KEY(id_tva) REFERENCES tva(id_tva),
135     CONSTRAINT fk_devis FOREIGN KEY(id_devis) REFERENCES devis(id_devis)
136   );
137   CREATE TABLE ligne_facture_client(
138     id_ligne_facture INT(10) PRIMARY KEY AUTO_INCREMENT,
139     quantite FLOAT(5,2) NOT NULL,
140     prixU FLOAT(6,2) NOT NULL,
141     reference VARCHAR(50) NOT NULL,
142     id_tva INT(10) NOT NULL,
143     id_facture INT(10) NOT NULL,
144     CONSTRAINT chk_quantiterlfcc CHECK (quantite > 0 ),
145     CONSTRAINT fk_reference_facture FOREIGN KEY(reference) REFERENCES produit(reference),
146     CONSTRAINT fk_facture_tva FOREIGN KEY(id_tva) REFERENCES tva(id_tva),
147     CONSTRAINT fk_facture FOREIGN KEY(id_facture) REFERENCES facture(id_facture)
148   );
149   CREATE TABLE bl(
150     id_bl INT(10) PRIMARY KEY AUTO_INCREMENT,
151     date_bl DATE NOT NULL,
152     actif BOOL NOT NULL,
153     id_adresse INT(10) NOT NULL,
154     id_facture INT(10) NOT NULL,
155     CONSTRAINT fk_id_adresse_bl FOREIGN KEY(id_adresse) REFERENCES adresse(id_adresse),
156     CONSTRAINT fk_id_facture_bl FOREIGN KEY(id_facture) REFERENCES facture(id_facture)
157   );
158   CREATE TABLE ligne_bl(
159     id_ligne_bl INT(10) PRIMARY KEY AUTO_INCREMENT,
160     quantite FLOAT(5,2) NOT NULL,
161     id_bl INT(10) NOT NULL,
162     id_ligne_facture INT(10) NOT NULL,
163     CONSTRAINT fk_bl FOREIGN KEY(id_bl) REFERENCES bl(id_bl),
164     CONSTRAINT fk_ligne_bl_ligne_facture FOREIGN KEY(id_ligne_facture) REFERENCES ligne_facture_client(id_ligne_facture)
165   );
166   );
167 
```

Annexe 10 : Schéma de Base de Donnée



Annexe 11 : Maquette Projet Android



Maxime Laurin

Né le 19/07/1996

17 rue des Loutres

17390 LA TREMBLADE

06 13 90 38 83

17.laurinmaxime@gmail.com

Titulaire du permis B (possession d'un véhicule)



CURRICULUM VITAE Analyste développeur Junior

FORMATIONS ET DIPLÔMES

- 2018-2019 : **Préparation au titre Niveau II CDAN (Concepteur développeur d'applications Numérique) U'DEV**
U'DEV -Bordeaux (33000) Ecole créée par CGI en coopération avec L'IPI et l'EPSI
- 2017 : **BTS SIO (Services informatiques aux organisations) option réseau**
Fénélon Notre-Dame - La Rochelle (17000)
- Juin 2015 : **Bac STI2D (sciences technique de l'industrie et du développement durable) option SIN**
Lycée Marcel Dassault – Rochefort (17300)

Langue : Anglais (bonnes notions) - Espagnol (notions)

COMPETENCES

- Développement : Langages : Java, Korn Shell, SQL, PHP, Html
Framework : java Spring, Angular
Outils : ETL(Informatica), Oracle SQL Developer, SAP BI 4, Android Studio, IntelliJ
- Réseaux : Installation et configuration d'un routeur, configuration d'un serveur web sous linux (Debian), adressage IP (packet tracer)
Configuration de serveur de déploiement et déploiement d'applications sous MacOS

EXPÉRIENCES PROFESSIONNELLES

- Décembre 2018/ Décembre 2019 : Alternance Analyste développeur Junior BI – CGI le Haillan : Développement SQL, scripts ksh, manipulations d'ETL
- Décembre 2017 : Saisonnier Ouvrier conchylicole - Huitres Geay : Préparation, emballage
- Janvier-Février 2017 : Stage DSI Université La Rochelle :Technicien réseau Déploiement , **maintenance**, configuration serveur déploiement
- Mai-juin 2016 : Stage C-imedia (Assistance et services informatiques) La Tremblade . Réparation , maintenance

CENTRES D'INTERÊT

Cinéma, Histoire, Musculation, Musique, Informatique (Overclocking, réparation), voyage...

STAGIAIRE

Nom : Lounim
Prénom : Maxime

Certification visée : Titre RNCP

Niveau II Concepteur - Développeur d'applications Numériques
«tuteur_Civilité».

Afin de parfaire la validation de la certification professionnelle de votre stagiaire, nous vous remercions de remplir exhaustivement cet **original** et nous le retourner dans les *meilleurs délais*. Ce document sera examiné par le jury en vue de l'attribution de la certification professionnelle de votre stagiaire.

L'intégration dans l'équipe de travail est-elle satisfaisante ?

Oui / Non

Le comportement en situation professionnelle est-il adapté à la culture de l'entreprise ?

Oui / Non

Le comportement relationnel est-il adapté au métier visé ?

Oui / Non

La capacité de travail fournie correspond-elle au niveau d'un professionnel du métier ?

Oui / Non

Le niveau de responsabilités attendu est-il satisfait ?

Oui / Non

Des questions sont-elles posées à bon escient ?

Oui / Non

Des difficultés en relation avec le niveau de responsabilités sont-elles surmontées ?

Oui / Non

Le niveau d'obligation de réserve demandé est-il pris en compte ?

Oui / Non

Les activités effectuées concourent-elles à la couverture du champ de compétences ?

Oui / Non

Le métier associé à la certification visée est-il compris ?

Oui / Non

Le stagiaire est-il apte à occuper la fonction visée même comme débutant ?

Oui / Non

Appréciation générale suite au stage en entreprise :

Maxime a intégré une équipe Agile sur des technologies non connues sur lesquelles il est monté en compétences. Il s'est vite et bien intégré au sein de ce projet à forte enjeux matériels et financiers. Il doit maintenant gagner en assurance.

Fait à le Haillan, le 26 septembre 2019

Signature du tuteur
et cachet de l'entreprise

CGI

Signature du stagiaire