

Option : Méthodes quantitatives et modélisation
pour l'entreprise

Stage Data : Gestion, modélisation de données et Data visualisation

Entreprise : SIMPLON.CO
Équipe : Data
Direction : Système d'information (SI)
Période : 02 juin 2025 – 28 Novembre 2025

Formez-vous à la Tech simplement

Simplon vous accompagne simplement pour développer vos compétences numériques et rend les métiers de la tech accessibles à toutes et à tous.

Du niveau Découverte au niveau Bac +5, rejoignez nos formations en Développement, Réseaux & Cybersécurité, Data / IA.

[Candidater](#)

[Découvrir les filières](#)



Rapport rédigé par
Ahmed HADAD

Tutrice en entreprise
Juliana CAVALCANTE

Tutrice universitaire
Sophie DABO

Remerciements

Je tiens à exprimer ma profonde gratitude envers toutes les personnes qui m'ont accompagné tout au long de mon stage au sein de la **direction du Système d'Information (SI) de Simplon.co.**

Je remercie tout particulièrement ma tutrice, **Juliana CAVALCANTE**, cheffe des projets Data, pour son encadrement attentif et sa disponibilité constante, ainsi que mon collègue Data Engineer, **Antony HENRION**, pour son accompagnement technique et ses précieux retours sur mes travaux.

Je souhaite également remercier **Hanifa ABDELAZIZ**, directrice de la **direction du Système d'Information (SI) de Simplon.co**, pour son soutien et sa bienveillance de l'équipe SI, qui ont contribué à la réussite de ce stage et à mon intégration au sein de l'équipe.

Ce stage m'a permis de développer mes compétences en **gestion, contrôle et modélisation de données** via **DBT** et **Google Cloud Platform**, ainsi qu'en **visualisation de données** avec **Looker Studio**. Il m'a également offert un contact permanent avec les équipes métiers, ce qui m'a permis de répondre efficacement à leurs besoins.

Ce stage m'a permis donc de couvrir l'ensemble du cycle de la donnée : de la **collecte et du contrôle de qualité, l'analyse, la modélisation des données à la restitution visuelle**, en développant rigueur, précision. Il m'a surtout permis de comprendre les fonctionnalités des outils techniques du métiers, notamment : **Google Cloud Platform** pour le stockage, le maintien et l'analyse des données massives, **DBT** pour la modélisations de données et **Looker Studio** pour la visualisation des données.

Les échanges avec les équipes métiers et la participation aux ateliers de data visualisation m'ont offert une expérience concrète et enrichissante, renforçant ma compréhension des enjeux de la donnée en entreprise.

Enfin, je remercie l'ensemble de la direction du Système d'Information (SI) de Simplon.co pour leur accueil chaleureux et leur collaboration tout au long de cette expérience professionnelle.

INTRODUCTION

Dans le cadre de mon stage au sein de l'**équipe Data** de la direction du **Système d'Information (SI)** de Simplon.co, j'ai eu l'occasion de travailler sur plusieurs projets visant à améliorer la qualité, la cohérence, la modélisation et la visualisation des données utilisées par les équipes opérationnelles, pédagogiques, administratives, commerciales et internationales.

Comme dans toute organisation, la gestion et l'exploitation de la donnée constituent des enjeux majeurs pour **Simplon.co** : **pilotage des formations, suivi des candidat · es, optimisation des placements ou encore diffusion d'outils internes**. Mon rôle a principalement consisté à contribuer à la **modélisation des données, à l'automatisation de leur traitement, la création de tableaux de bord** à destination des équipes métiers ainsi qu'aux **traitements des tickets** des équipes métiers liés aux sujets Data .

Je présente dans ce rapport certaines missions réalisées au cours de mon stage, notamment un projet central intitulé **Projet placement**, ainsi que plusieurs **travaux transverses**.

Ce rapport expose les enjeux, les approches techniques mobilisées et les résultats obtenus pour chacun de ces travaux. Il est structuré comme suit :

1. Une brève présentation de **Simplon.co** et du contexte métier,
2. Trois chapitres composés de :
 - (a) **Cycle de la donnée et environnement technique** : présentation des outils techniques utilisés et du cycle d'étude au sein de l'équipe Data.
 - (b) **Projet Placement** : mise en place de tableaux de bord interactifs permettant :
 - a) aux commerciaux d'identifier rapidement les profils des candidat · es en recherche d'entreprise (filtres par région, formation, etc.), adaptés dès qu'un besoin est détecté chez un client ;
 - b) au directeur du pôle alternance à distance de suivre les statistiques de placement des sessions en alternance.
 - (c) **Travaux transverses** :
 - a) résolution de tickets : réponses aux besoins des équipes (exports, analyses, vérifications spécifiques) ;
 - b) mise en place des contrôles de données internationales permettant d'assurer la fiabilité des indicateurs.
3. Une conclusion

Table des matières

1 Présentation de Simplon.co	4
2 Cycle de la donnée et environnement technique	5
2.1 Gestion des données	5
2.2 Extraction et stockage des données : Talend	6
2.3 Outils de modélisation, d'analyse et de visualisation	6
2.3.1 DBT et GCP : Modélisation et hébergement de données	6
2.3.2 Looker Studio : Visualisation	8
2.3.3 Outils d'analyse et de reporting	8
2.4 Stratégie de l'équipe Data	11
3 Projet PLACEMENT	12
3.1 Contexte	12
3.2 Demande initiale du pôle alternance à distance	12
3.3 Proposition de l'équipe Data	12
3.4 Traitement de la demande	13
3.4.1 Identification des modèles techniques	13
3.4.2 Flux de données Talend et Extraction ponctuelle des données	14
3.4.3 Modélisation des données	14
3.5 Data visualisation	21
3.5.1 Construction du tableau de bord	21
3.5.2 Atelier de restitution	22
4 Travaux transverses	23
4.1 Résolution des tickets	23
4.1.1 Données certification Agilité	23
4.1.2 Apple Reporting	25
4.2 Contrôle des données internationales	28
4.2.1 Données renseignées dans les fichiers pays	28
4.2.2 Problèmes observés dans le tableau de bord existant	29
4.2.3 Initiative de l'équipe Data	29
5 Conclusion	33
6 Annexe	34

Présentation de Simplon.co

1. Présentation de l'entreprise

Créée en 2013, **Simplon.co** est un réseau de fabriques numériques engagé pour un numérique inclusif. L'organisation forme aux métiers du numérique, notamment des publics éloignés de l'emploi, via des bootcamps intensifs fondés sur une pédagogie active et par projets. Aujourd'hui, Simplon.co constitue l'un des plus grands réseaux de bootcamps inclusifs au monde, présent dans **34 pays** avec plus de **84 écoles** et plus de **19 937 apprenant · es** formé · es à l'international.

2. Organisation internationale

L'activité internationale de Simplon.co s'appuie sur des partenaires partout dans le monde principalement en Afrique mais aussi Europe, Asie, Moyen Orient et Amérique Latine.

L'activité internationale de Simplon.co s'appuie sur deux principaux pôles :

- a) **Simplon Maghreb** opérant des programmes sur l'ensemble du territoire ;
- b) **Simplon Africa**, implantée dans plusieurs pays d'Afrique subsaharienne.

Ces pôles pilotent des projets de formation adaptés aux besoins locaux, en partenariat avec des acteurs publics, privés ou institutionnels.

3. Projets et partenariats clés

Simplon.co mène une large variété de projets : consortiums européens, programmes financés par l'**AFD** ou l'**Union européenne**, déploiement de bootcamps "**Powered by Simplon**".

Un exemple notable est le partenariat avec les **Orange Coding Academies** en Jordanie, où plus de **742 apprenant · es** ont été formé · es depuis 2019, avec un taux de **sortie positive de 60%**.

4. Savoirs-faire et expertise

L'entreprise intervient sur l'ensemble de la chaîne de valeur d'un projet de formation numérique :

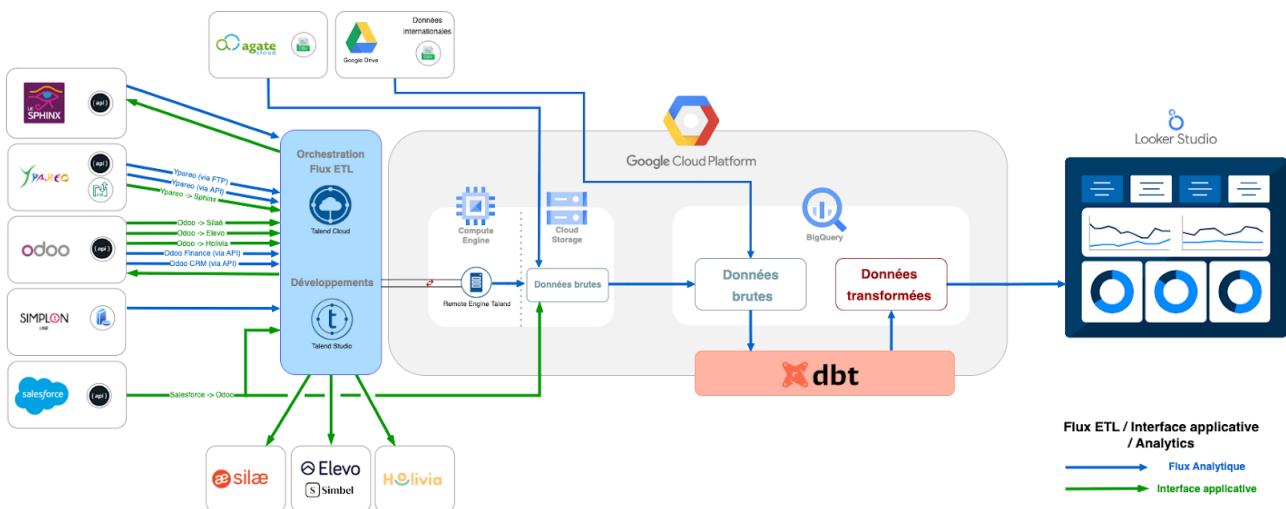
- a) études de besoins et faisabilité ;
- b) ingénierie pédagogique et création de parcours ;
- c) formation des formateurs ;
- d) dispositifs qualité, sourcing inclusif et certification ;
- e) mobilisation d'outils numériques, dont la plateforme pédagogique *Simplonline*.

Simplon.co combine ainsi expertise pédagogique, innovation sociale et outils numériques pour former aux compétences du futur et accompagner les transformations du marché de l'emploi.

Cycle de la donnée et environnement technique

Avant de présenter les missions réalisées, il est essentiel de comprendre le cycle de vie de la donnée chez Simplon.co : d'où proviennent les données ? Quels outils techniques sont mobilisés ? Comment sont-elles migrées et exploitées par l'équipe DATA ?

Chez Simplon.co, le cycle de la donnée permet de collecter, de contrôler, de modéliser et de disponibiliser les données liées à l'activité de formation, depuis les candidatures des futurs apprenants jusqu'aux questionnaires d'insertion envoyés après la fin des formations. Il inclut notamment les données relatives aux candidatures et au placement des candidats, ainsi que, plus largement, les données du SI Formation et du SI Entreprise (flux RH, etc.), afin d'accompagner de manière fiable les décisions stratégiques :



2.1 Gestion des données

Les données sont hébergées dans différents outils internes comme par exemple : Odoo, CRM, Yparéo, Sphinx et Simplonline. Chacun possède un rôle spécifique selon le type de données :

1. **Odoo** : Plateforme qui centralise des données concernant les employés et les finances.
2. **CRM** : Plateforme qui centralise les inscriptions des apprenants et gère leur suivi administratif.
3. **Yparéo** : Plateforme contenant les informations sur les apprenants, les groupes et les sessions de formation.
4. **Sphinx** : Plateforme permettant de collecter des données issues de questionnaires. Simplon suit attentivement ses apprenants après la formation : satisfaction, suggestions, insertion professionnelle, etc.
5. **Simplonline** : Plateforme interne pédagogique permettant de suivre l'activité des apprenants.

2.2 Extraction et stockage des données : Talend

Les données sont extraites via l'outil ETL *Talend*. Les flux sont exécutés quotidiennement.

Utilisé chez Simplon.co pour extraire les données de sources externes (APIs, bases de données, fichiers), et les charger dans un entrepôt centralisé sans traitement préalable, on parle de *Google Cloud Storage (GCS)*, *Talend* est un outil ETL (Extract, Transform, Load) open source. Les flux Talend sont planifiés et exécutés quotidiennement pour alimenter l'écosystème data de manière automatisée et fiable.

Note : Cette étape relève principalement de la responsabilité du Data Engineer. Mon rôle consistait à contrôler les exécutions, alerter en cas d'échec ou d'incohérence détectée. Pour les modèles et analyses ad-hoc, j'interviens sur l'extraction des données via des APIs.

2.3 Outils de modélisation, d'analyse et de visualisation

Une fois que les flux des données sont déposées dans *Google Cloud Storage (GCS)*, les données brutes sont utilisées pour créer des modèles structurés et optimisés répondant aux besoins des équipes métiers. Ces modèles sont développés dans *DBT* et hébergés sur *BigQuery*.

2.3.1 DBT et GCP : Modélisation et hébergement de données

1. **DBT** est un outil de transformation de données qui permet d'écrire des modèles en SQL, de les versionner et de les organiser sous forme de projets. Au sein de Simplon.co, il est utilisé pour modéliser les données, gérer les environnements (développement, recette, production) et assurer la traçabilité des modifications grâce au contrôle de version.
2. **BigQuery** est l'entrepôt de données cloud de Google utilisé par Simplon.co pour stocker et interroger les données. Il héberge les modèles créés avec DBT et permet d'exécuter des requêtes analytiques à grande échelle pour répondre aux besoins des équipes métiers.

Ma mission principale consistait à transformer et structurer les données via *DBT* sur *BigQuery*. J'ai construit des modèles analytiques adaptés aux besoins métiers, reposant sur *l'architecture de Data Warehouse*, notamment les modèles à plat et en étoile :

1. Modélisation à plat :

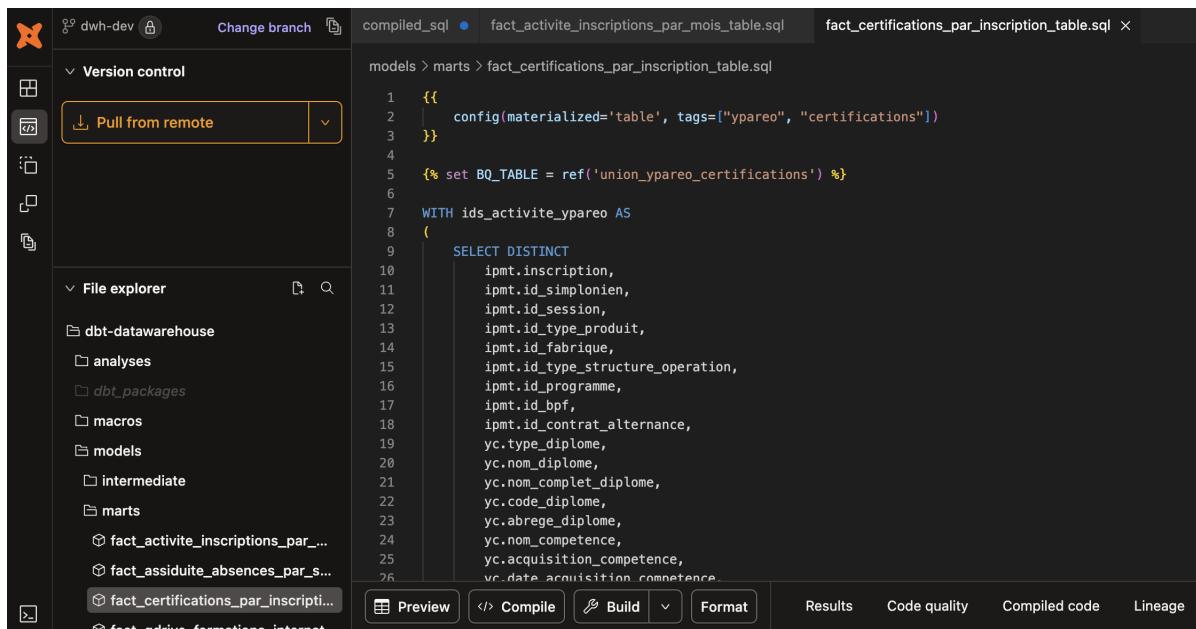
La modélisation à plat consiste à stocker toutes les informations dans une seule table, sans séparation entre faits et dimensions. Elle simplifie l'accès aux données mais entraîne souvent des redondances pour les modèles complexes. Ce modèle est adapté aux datasets de petite taille.

2. Modélisation en étoile :

La modélisation en étoile organise les données autour d'une table de faits centrale reliée à plusieurs tables de dimensions. Elle réduit la redondance, améliore les performances analytiques et facilite la compréhension des données. C'est le modèle le plus utilisé dans notre équipe DATA. Qu'est ce qu'une table de fait et de dimension ?

- (a) La **table de faits** contient les événements mesurables, comme chaque certification obtenue ou chaque inscription à une formation. Elle enregistre les résultats, dates et indicateurs de performance pour chaque apprenant. Les clés étrangères permettent de la relier aux dimensions (apprenant, session, formation).
- (b) Les **tables de dimensions** décrivent les entités de Simplon.co : les apprenants (nom, date de naissance, genre), les formations (titre, code programme), ou les sessions (dates, lieux). Elles servent à qualifier et filtrer les faits lors des analyses. Chaque dimension possède une clé primaire qui se relie aux faits.

Voici par exemple un modèle visant à structurer les informations d'un apprenant en lien avec la certification associée à sa formation :



The screenshot shows the dbt CLI interface. On the left, there's a sidebar with 'Version control' (branch: 'dwh-dev') and 'File explorer'. The 'File explorer' section shows a tree structure with nodes like 'dbt-datawarehouse', 'analyses', 'dbt_packages', 'macros', 'models', 'intermediate', 'marts', and several fact models under 'marts'. The main area is a code editor with the following SQL query:

```

models > marts > fact_certifications_par_inscription_table.sql
fact_certifications_par_inscription_table.sql

1  {{ config(materialized='table', tags=['ypareo', "certifications"])}
2
3  }
4
5  {% set BQ_TABLE = ref('union_ypareo_certifications') %}
6
7  WITH ids_activite_ypareo AS
8  (
9      SELECT DISTINCT
10          ipmt.inscription,
11          ipmt.id_simplonien,
12          ipmt.id_session,
13          ipmt.id_type_produit,
14          ipmt.id_fabrique,
15          ipmt.id_type_structure_operation,
16          ipmt.id_programme,
17          ipmt.id_bpf,
18          ipmt.id_contrat_alternance,
19          yc.type_diplome,
20          yc.nom_diplome,
21          yc.nom_complet_diplome,
22          yc.code_diplome,
23          yc.abrege_diplome,
24          yc.nom_competence,
25          yc.acquisition_competence,
26          vc.date_acquisition_competence

```

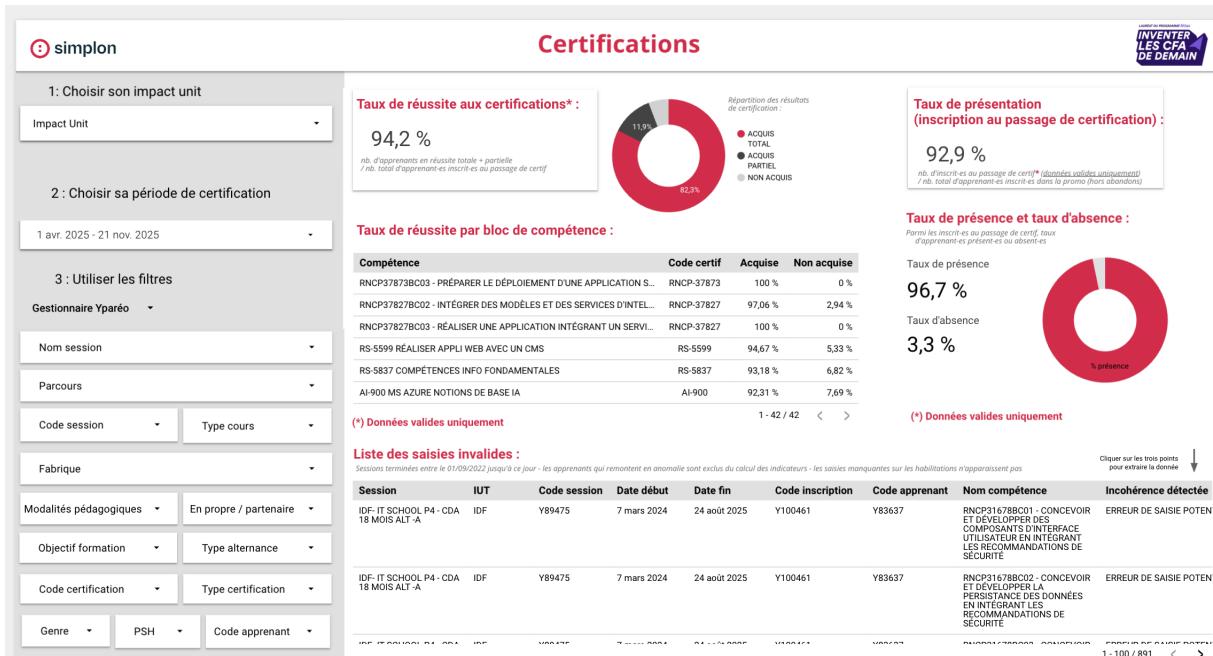
At the bottom of the code editor, there are buttons for 'Preview', 'Compile', 'Build', 'Format', 'Results', 'Code quality', 'Compiled code', and 'Lineage'.

Les modèles sont hébergé dans *BigQuery*, prêt à être interrogé et visualiser sur Looker studio :

The screenshot shows the Google Cloud search interface. At the top, there's a header with the Google Cloud logo and the project name "DWH-Dev". Below the header is a search bar with placeholder text "Tapez / pour rechercher des ressources, des docume...". The main area has a search bar labeled "Rechercher des ressources BigQuery" and a button "N'afficher que les favoris". A specific search result titled "fact_certifications_par_inscription_table" is highlighted.

2.3.2 Looker Studio : Visualisation

Anciennement *Google Data Studio*, *Looker Studio* est l'outil principal de visualisation chez Simplon.co. Il permet de connecter les modèles BigQuery pour créer des tableaux de bord interactifs, personnalisables et partagés avec les équipes métiers, facilitant le suivi des indicateurs clés (KPIs) comme les KPIs de certification ci-dessous :

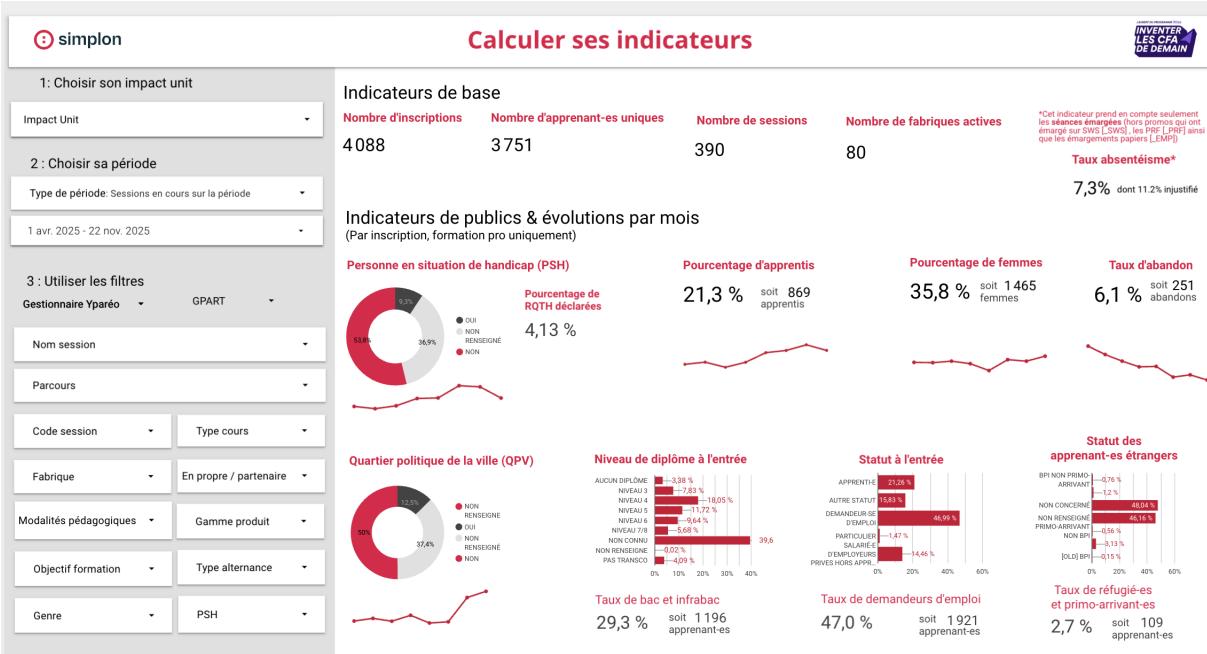


2.3.3 Outils d'analyse et de reporting

Pour répondre aux besoins métiers, plusieurs approches sont utilisées :

1. Tableaux de bord de pilotage :

Des tableaux de bord opérationnels sont mis en place et permettent aux équipes de suivre leurs indicateurs (nombre d'apprenants, sessions, taux de femmes, etc.).



Google Cloud DWH-Prod Tapez / pour rechercher des ressources, des documents, des produits, etc

Requête sans titre

```

1 WITH base AS (
2     SELECT *EXCEPT(situation_pro, repondant),
3         CASE situation_pro
4             WHEN "NON RENSEIGNÉ" THEN NULL
5             WHEN "EN ATTENTE" THEN NULL
6             WHEN "NON APPLICABLE" THEN NULL
7             WHEN "EN RECHERCHE D'EMPLOI" THEN "INACTIF"
8             WHEN "SANS ACTIVITÉ" THEN "INACTIF"
9             WHEN "INACTIF" THEN "INACTIF"
10            WHEN "ACTIF OCCUPÉ HORS ALTERNANCE" THEN "SORTIE POSITIVE"
11            WHEN "EN EMPLOI" THEN "SORTIE POSITIVE"
12            WHEN "EN FORMATION" THEN "SORTIE POSITIVE"
13        ELSE situation_pro
14    END AS situation_pro_calcul
15    FROM fact_sphinx_questionnaires_par_inscription_table_temp
16
17    SELECT DISTINCT
18        code_inscription,
19        CASE
20            WHEN situation_pro_calcul IS NOT NULL THEN 1
21            ELSE 0
22        END AS repondant,
23        situation_pro_calcul,
24        sortie_positive AS tag_sortie_positive,
25        code_session,
26        nom_session,
27        objectif_session,
28        type_produit,
29        parcours_formation,
30        date_debut_session,
31        date_fin_session
32    FROM base
33    WHERE IUT_session LIKE "%GDO%"
34    AND (date_fin_session BETWEEN '2024-05-01' AND '2025-10-31')

```

Cette requête traitera 6,64 Mo lors de son exécution.

Les données ont ensuite été exportées et analysées dans Google Sheets via un tableau croisé dynamique :

nom_session	Nombre d'apprenants inscrits	Nombre répondants aux questionnaires	Nombre sortie positif	Taux de sortie positive
GDO_ANGERS_CIF_ARTIS_INT_P08	16	8	2	25,00%
GDO_ANGERS_CIF_REFUGIES_142808_SAS	14	5	1	20,00%
GDO_ANGERS_TAI_142816_INT	16	0	0	#DIV/0!
GDO_ANGERS/NANTES_TAI_J0_P5	16	6	4	66,67%
GDO_ANGERS/NANTES_TAI_P07_J0	22	0	0	#DIV/0!
GDO_CAEN_ISEN_DEV IA1_ALT_P1	14	6	3	50,00%
GDO_FABNUMÉRIQUE41_BLOIS_SASCMS_P1_EMP	20	0	0	#DIV/0!
GDO_ISEN_BREST_DEVIA_39232_ALT_P3_EMP	14	1	0	0,00%
GDO_NANTES_CDA_DSI_INT_P11	15	4	3	75,00%
GDO_NANTES_DEV COBOL POEI BNPP_INT_P19	12	4	4	100,00%
GDO_NANTES_POEI_REL_CLIENT_INT_P18	13	2	2	100,00%
GDO_NANTES_SAS_INFRA_INT_P15	16	9	1	11,11%
GDO_NANTES_SAS_REFUGIES_INT_P17	13	2	1	50,00%
GDO_NANTES_TAI_INT_P14	18	10	8	80,00%
GDO_NANTES_TAI_P20_INT	15	8	7	87,50%
GDO_NANTES_TAI_UP_INTERIM_P24_INT	10	0	0	#DIV/0!
GDO_RENNES_DEVFRONT_INT_P15	15	8	3	37,50%
GDO_RENNES_DEVIAP3_ALT_24_P10	13	4	2	50,00%
GDO_RENNES_SAS_CMS_FEMMES_INT_P18	17	9	6	66,67%
GDO_RENNES_T_HELPDESK_INT_P16	15	10	6	60,00%
GDO_RENNES_TECHNICIEN HELPDESK_INT_P19	16	0	0	#DIV/0!
GDO_SAINT-BRIEUC_MSDEV_IA_P2_INT_EMP	16	3	2	66,67%
GDO_TOURS_GRETA_DEV MSIA3_ALT_P6_EMP	13	0	0	#DIV/0!
GDO_TOURS_GRETA_DEV MSIA3_INT_P6	14	9	9	100,00%
GDO_TOURS_GRETA_DEVIA4_INT_P7_EMP	12	0	0	#DIV/0!
Total général	375	108	64	59,26%

2.4 Stratégie de l'équipe Data

La force de l'équipe Data réside dans sa rigueur et son organisation, permettant de garantir la fiabilité et la pérennité des informations :

« *Traiter la donnée, c'est plus qu'un travail technique : c'est une responsabilité envers l'information.* »

Trois environnements structurent les projets data :

1. **Phase de collecte (RAW) :**

Environnement dédié aux données collectées, données brutes, sans la moindre transformation.

2. **Phase de Développement (DEV) :**

Environnement dédié au développement, aux tests et aux vérifications. Le cycle n'y est pas automatisé.

3. **Phase de Recette (REC) :**

Espace interne à l'équipe DATA permettant de valider les traitements et de disposer d'un secours en cas de problème en production.

4. **Phase de Production (PROD) :**

Environnement final partagé avec l'ensemble de l'entreprise. Les tableaux de bord y sont accessibles aux équipes métiers.

Ce cycle constituait le cadre technique de mon travail quotidien.

Projet PLACEMENT

3.1 Contexte

L'un des enjeux majeurs de l'entreprise est d'accompagner efficacement les apprenants dans leur recherche d'entreprise. Il s'agit d'un accompagnement continu tout au long de leur parcours de prospection, notamment concernant l'accès aux offres des entreprises partenaires de Simplon, l'amélioration du CV et de la lettre de motivation, ainsi que la préparation aux entretiens.

Habituellement, les **Chefs de Projets (CDP)**, alimentaient quotidiennement et manuellement un tableau **Google Sheets** recensant les apprenants en recherche d'alternance. Ce fichier était ensuite utilisé par l'équipe commerciale pour identifier les candidats correspondant aux besoins exprimés par les entreprises partenaires (filtres par région, formation, niveau, etc.).

Cependant, ce fonctionnement reposant sur une saisie manuelle présentait plusieurs limites : risques d'erreurs, oublis de candidats, absence d'harmonisation, perte de temps et difficulté à garantir la fiabilité de l'information.

3.2 Demande initiale du pôle alternance à distance

Face à ces difficultés, le pôle alternance à distance a sollicité l'équipe Data pour mettre en place un tableau de bord **Looker Studio** reprenant les informations du tableau **Google Sheets**, avec les mêmes capacités de filtrage (région, ville, formation, etc.).

3.3 Proposition de l'équipe Data

Lors d'un échange dédié, ma tutrice et moi avons proposé une solution plus robuste et entièrement automatisée, fondée sur l'utilisation des données déjà présentes dans le **CRM**.

Les champs nécessaires à la construction du tableau de bord sont répartis en deux catégories :

1. **Champs administratifs :**
nom, prénom, mail, formation, région, ville, lien vers la fiche candidat, lien vers le Cv.
2. **Champs liés au placement et au caractère sociales :**
mobilité, QPV, RSA, RQTH, type de contrat recherché, remarques du candidat.

Ces informations existent nativement dans le **CRM** mais sont réparties dans plusieurs modèles techniques :

Par respect du RGPD, toutes les informations personnelles visibles sur la photo (emails, prénoms et code apprenant) ont été masquées.

The screenshot shows the Odoo CRM interface with several panels:

- Pipeline:** Shows leads categorized by status: "Jrm. candidature envoyé" (2), "Candidature reçue" (2), "Invité au Starter (chgmt nom)" (409), "Entretien technique" (410), "JDP 5 Après-Midi" (1), "JDP 5 Matin" (0), "JDP 4 Après-Midi" (0), "JDP 4 Matin" (25), "JDP 3 Après-Midi" (17), "JDP 2 Après-Midi" (43), "JDP 2 Matin" (17), "JDP 1 Après-Midi" (159), and "JDP 1 Matin" (1226).
- Recherche entreprise:** A search bar for companies.
- Candidature validée:** A list of validated leads.
- Recherche...**: A search bar for leads.
- Filtres**, **Regrouper par**, and **Favoris** buttons.
- NOUVEAU** button.
- Vivier (21442)** sidebar.
- Liste d'attente (183)** sidebar.

Au lieu d'utiliser les données du [Google Sheet](#), extraire les données à partir du [CRM](#) :

1. centraliser l'ensemble des données dans une base unifiée,
2. automatiser l'alimentation quotidienne,
3. garantir l'unicité et la qualité de la donnée,
4. produire un tableau de bord fiable à partir de cette base.

Avantages :

1. Processus totalement automatisé, sans saisie manuelle.
2. Mise à jour quotidienne des données donc rien ne nous échappe.
3. Accès en lecture seule dans Looker Studio pour les utilisateurs, limitant les erreurs.

Points de vigilance :

1. La qualité des données dépend fortement de la bonne complétude des informations dans le [CRM](#). Par exemple, un candidat en recherche d'entreprise, doit absolument être à l'étape "[Recherche d'entreprise](#)" dans le [CRM](#).
2. Certaines données proviennent des réponses à des questionnaires : leur remplissage doit être vérifié par l'équipe Qualité.

3.4 Traitement de la demande

3.4.1 Identification des modèles techniques

La première étape a consisté à identifier les modèles techniques dans [Odoo](#) contenant les champs nécessaires, ainsi que les clés permettant les jointures. Les modèles retenus sont :

1. **ir_attachment** : pièces jointes (dont les CV).
2. **survey_user_input** : réponses principales du questionnaire de candidature.
3. **survey_user_input_line** : détail des réponses aux questions.
4. **crm_lead** : informations administratives (déjà présent dans Google Cloud Storage).

3.4.2 Flux de données Talend et Extraction ponctuelle des données

J'ai listé les champs techniques nécessaires pour chaque modèle et les ai transmis au Data Engineer. Celui-ci a mis en place des **flux de données Talend** déposant quotidiennement les données dans **Google Cloud Storage**.

1. Flux de données Talend

Les principaux flux de données Talend : modèles et champs techniques extraits

1. **raw_odoo_ir_attachment** : id, res_id, name, res_model, access_token, create_date.
2. **raw_odoo_survey_user_input** : id, access_token, survey_id, dates, scoring, e-mail, statut.
3. **raw_odoo_survey_user_input_line** : id, user_input_id, question_id, display_name, skipped, create_date.

2. Extraction des données

Pour le flux de données **raw_odoo_survey_user_input_line**, une **technique d'incrémentation** était nécessaire dans la modélisation qui suit. J'ai développé un **script Python personnalisé** pour extraire l'**historique complet** de ce modèle par **appel API Odoo**, assurant ainsi une couverture exhaustive des données historiques.

Cette technique mérite d'être explicité en détail (nous verrons cela dans la partie [2.4.3 Modélisation des données](#) sur la construction des **tables de Staging**, notamment la construction de la table **stg_odoo_survey_user_input_line**).

3.4.3 Modélisation des données

Dans ce projet, j'ai travaillé sur le modèle de faits existant **fact_odoo_crm_opportunité_table**, qui regroupe les informations principales sur les candidats Simplon.co.

Ce modèle de faits manquait d'informations clés pour analyser le **placement des candidats** (mobilité, QPV, etc.) et leurs **caractéristiques socio-démographiques et administratives** (région, ville, lien fiche candidat, CV, etc.).

J'ai donc pensé à créer une **table dimension candidats** pour structurer ces attributs descriptifs manquantes, puis l'ai reliée au modèle de faits via une clé étrangère.

Cette approche est ce qu'on appelle **modélisations en étoile** qui permet :

1. de réduire la redondance dans la table de faits,
2. de faciliter les jointures pour les analyses métiers,
3. d'améliorer la performance des requêtes BigQuery et Looker Studio.

Ainsi, j'ai réalisé les modélisation dans **DBT**, en suivant une approche conceptuelle et pratique **d'architecture de data warehouse** (Staging pour les données brutes, ODS et dimension pour les données transformées, puis modélisation finale pour les opérationnel opérationnelles) :

tables staging → tables ods → table dimension → table de faits

Je vais détailler la construction techniques des ses tables.

1. Tables de Staging

Dans cette étape, juste on renomme les champs techniques en champs compréhensifs pour les équipes métiers.

```
models > staging > odoo > stg_odoo_survey_user_input.sql

1  {{{
2  |   config(tags=["odoo", "crm"])
3  }}}
4
5  {% set BQ_TABLE = source('raw_odoo', 'raw_odoo_survey_user_input') %}
6
7  SELECT
8      id AS envoi_sondage_id,
9      access_token AS jeton_identification,
10     user_input_line_ids AS reponse_sondage_id,
11     survey_id ,
12     create_date AS date_envoi_sondage,
13     start_datetime AS date_debut_sondage,
14     end_datetime AS date_fin_sondage,
15     attempts_number AS nombre_tentatives_envoi,
16     scoring_percentage AS pourcentage_notation,
17     scoring_total AS notation_totale,
18     email,
19     state AS statut_sondage,
20
21  FROM {{ BQ_TABLE }}
22
```

stg_odoo_ir_attachment

```
models > staging > odoo > stg_odoo_ir_attachment.sql

1  {{{
2  |   config(tags=["odoo", "crm"])
3  }}}
4
5  {% set BQ_TABLE = source('raw_odoo', 'raw_odoo_ir_attachment') %}
6
7  SELECT
8      id AS piece_jointe_id,
9      res_id AS source_id,
10     name AS nom_piece_jointe,
11     res_model AS modele_source,
12     access_token AS jeton_identification,
13     create_date AS date_creation
14
15  FROM {{ BQ_TABLE }}
```

stg_odoo_survey_user_input

Pour la construction de la table `stg_odoo_survey_user_input_line` suivante, une [technique d'incrémentation](#) était nécessaire et mérite d'être explicité en détail :

La table `stg_odoo_survey_user_input_line` qui centralise les réponses aux questionnaires des

candidats issues d’Odoo, présente des contraintes de volumétrie lors de l’extraction des données. En effet, le volume élevé de réponses accumulées au fil du temps rendait un rechargeement complet quotidien très lourd, celui-ci reposant sur des **appels API** vers le CRM Odoo via **Talend**, avec un risque de surcharge du **serveur Odoo** et d’instabilité des **flux d’extraction**.

Afin de sécuriser ce processus,

1. J’ai fait une extraction complète ponctuelle de l’**historique des réponses** par **appel API Odoo** à l’aide d’un **script Python** : voir annexe. Cette base historique sert de socle initial.
2. En parallèle, un **flux Talend** a été mis en place par le Data Engineer pour récupérer quotidiennement uniquement les nouvelles réponses soumises, limitant ainsi la charge sur l’API et garantissant la continuité de l’alimentation des données.

J’ai donc implémenté la table **stg_odoo_survey_user_input_line** en mode incrémental, pour fusionner l’historique et les nouvelles données sans retraitier l’ensemble du jeu de données à chaque exécution. Concrètement, la logique SQL s’appuie sur une **clé unique (id)** pour assurer l’unicité des enregistrements :

Lors du chargement initial, la table historique est utilisée, tandis que lors des exécutions incrémentales, seules les nouvelles données issues du flux Talend sont intégrées.

Cette implémentation permet d’ajouter quotidiennement les nouvelles réponses aux questionnaires sans impacter les données existantes :

```

models > staging > odoo > stg_odoo_survey_user_input_line_combined.sql

4
5  {{
6    config(
7      materialized='incremental',
8      unique_key='id',
9      on_schema_change='append_new_columns',
10     pre_hook="#% if not is_incremental() %}{ log('Loading initial historical data...') }% endif %"
11   )
12 }
13
14 {% set BQ_TABLE =
15   source('raw_odoo',
16         'raw_odoo_survey_user_input_line'
17         if is_incremental()
18       else 'raw_odoo_survey_user_input_line_hst'
19   )
20 %}
21
22 select
23   id,
24   user_input_id AS envoi_sondage_id,
25   question_id,
26   cast(ifnull(skipped, false) AS bool) AS skipped,
27   display_name AS reponse,
28   CAST(IFNULL(SAFE_CAST(answer_is_correct AS BOOL), FALSE) AS BOOL) AS reponse_est_correcte,
29   answer_score AS note_reponse,
30   create_date AS date_creation_reponse,
31   answer_type AS type_reponse,
32   value_text_box AS valeur_texte_reponse,
33   value_numerical_box AS valeur_numerique_reponse,
34   suggested_answer_id,
35   value_char_box AS valeur_char_reponse,
36   CURRENT_TIMESTAMP() AS insertion_timestamp
37   from {{ BQ_TABLE }}

```

stg_odoo_survey_user_input_line

FIGURE 3.4.1 – Modèles de staging des données Odoo

2. Tables ODS

Les tables `ods` permettent d'effectuer les premières jointures et traitements métiers. J'ai principalement construit :

1. **ods_odoo_crm_ir_attachment** : récupération du CV le plus récent pour chaque apprenant. La technique est de recuperer le dernier fichier ajouté par le candidat contenant le mot Cv. Ce dernier est recuperé via un lien construit par la concaténation de la page web CRM de simplon avec l'id du fichier rajouté.
2. **ods_odoo_crm_survey_question_filtered** : extraction des questions pertinentes, c'est à dire que parmis toutes les questions posées, on filtre les questions demandant le nom, la nationalité, le QPV, RSA, etc.), et traquer les reponses de ces questions pour chaque candidat.
3. **ods_odoo_crm_survey_reponse_pivot** : reconstruction du profil candidat à partir des réponses au sondage (pivot). Il s'agit de reconstruire les champs souhaitées tirés par les réponses des candidatnt. En principe, on construit les champs nécessaire, le nom, la nationalité, le QPV, RSA, etc.) pour chaque apprenant, donc chaque ligne renseigne les informations du candidat.

```
models > intermediate > ods > ods_odoo_crm_ir_attachment.sql

1  {{{
2    config(tags=["odoo", "crm"])
3  }}}
4
5  {% set BQ_TABLE = ref('stg_odoo_ir_attachment') %}
6
7  WITH attachment AS (
8    SELECT
9      nom_piece_jointe,
10     piece_jointe_id,
11     source_id,
12     CONCAT('https://simplon.odoo.com/web/content/', CAST(piece_jointe_id AS STRING)) AS lien_cv
13   FROM {{ BQ_TABLE }}
14  WHERE modele_source = 'crm.lead'
15    AND (
16      nom_piece_jointe LIKE '%cv%'
17      OR nom_piece_jointe LIKE '%Cv%'
18      OR nom_piece_jointe LIKE '%CV%'
19    )
20  )
21
22  SELECT
23    source_id,
24    ARRAY_AGG(piece_jointe_id ORDER BY piece_jointe_id DESC LIMIT 1)[OFFSET(0)] AS piece_jointe_id,
25    ARRAY_AGG(nom_piece_jointe ORDER BY piece_jointe_id DESC LIMIT 1)[OFFSET(0)] AS nom_piece_jointe,
26    ARRAY_AGG(lien_cv ORDER BY piece_jointe_id DESC LIMIT 1)[OFFSET(0)] AS lien_cv
27  FROM attachment
28  GROUP BY source_id
```

ods_odoo_crm_ir_attachment

```
ods_odoo_crm_survey_question_filtered.sql >
```

```
models > intermediate > ods > ods_odoo_crm_survey_question_filtered.sql

1   {{  
2     config(tags=["odoo", "crm"])  
3   }}  
4  
5   {% set BQ_TABLE = ref('stg_odoo_survey_user_input_line_combined') %}  
6  
7   --- modele survey_input_line : filtrer les questions  
8   SELECT  
9     id AS reponse_sondage_id,  
10    -- extraire seulement l'id de la envoi_sondage  
11    CAST(JSON_EXTRACT_SCALAR(envoi_sondage_id, '$[0]') AS INT64) AS envoi_sondage_id,  
12    -- extraire seulement l'id de la question  
13    CAST(JSON_EXTRACT_SCALAR(question_id, '$[0]') AS INT64) AS question_id,  
14    -- extraire seulement le libelé de la question  
15    JSON_EXTRACT_SCALAR(question_id, '$[1]') AS titre_question,  
16    skipped,  
17    reponse,  
18    reponse_est_correcte,  
19    note_reponse,  
20    date_creation_reponse,  
21    type_reponse,  
22    valeur_texte_reponse,  
23    valeur_numerique_reponse,  
24    valeur_char_reponse  
25  
26   FROM {{ BQ_TABLE }}  
27   -- On filtre uniquement les questions pertinentes  
28   WHERE  (JSON_EXTRACT_SCALAR(question_id, '$[1]') LIKE "%rénom%"  
29         | AND JSON_EXTRACT_SCALAR(question_id, '$[1]') NOT LIKE "%férent%")  
30   OR JSON_EXTRACT_SCALAR(question_id, '$[1]') IN (  
31     "Nom", "Nom de naissance", "Nom d'usage / Marital")  
32   OR JSON_EXTRACT_SCALAR(question_id, '$[1]') LIKE "%âge de naissance%"  
33   ---- Si on est de nationalité européenne  
34   OR (JSON_EXTRACT_SCALAR(question_id, '$[1]') LIKE "%nationalité%"  
35         | AND JSON_EXTRACT_SCALAR(question_id, '$[1]') LIKE "%europ%"")  
36   ---- Si on est pas de nationalité européenne donc renseigne la nationalité ou même le pays d'origine UE  
37   OR (JSON_EXTRACT_SCALAR(question_id, '$[1]') LIKE "%nationalité%"  
38         | AND JSON_EXTRACT_SCALAR(question_id, '$[1]') NOT LIKE "%europ%")
```

```
39  
40   OR JSON_EXTRACT_SCALAR(question_id, '$[1]') LIKE "%numéro de téléphone%"  
41  
42   OR JSON_EXTRACT_SCALAR(question_id, '$[1]') LIKE "%QPV%"  
43  
44   OR (JSON_EXTRACT_SCALAR(question_id, '$[1]') LIKE "%RSA%"  
45         | AND JSON_EXTRACT_SCALAR(question_id, '$[1]') NOT LIKE "%statut%")  
46   -- inscription france travail  
47   OR (JSON_EXTRACT_SCALAR(question_id, '$[1]') LIKE "%france%"  
48         | AND JSON_EXTRACT_SCALAR(question_id, '$[1]') LIKE "%avail%"  
49         | AND JSON_EXTRACT_SCALAR(question_id, '$[1]') LIKE "%inscri%"  
50         | AND JSON_EXTRACT_SCALAR(question_id, '$[1]') NOT LIKE "%gence%"  
51         | AND JSON_EXTRACT_SCALAR(question_id, '$[1]') NOT LIKE "%conseiller%")  
52   ---- Contact france travail ou agence inscrit ou numero d'inscription de france travail  
53   OR (  
54     (JSON_EXTRACT_SCALAR(question_id, '$[1]') LIKE "%rance%"  
55       | and JSON_EXTRACT_SCALAR(question_id, '$[1]') LIKE "%avail%"  
56       | and (JSON_EXTRACT_SCALAR(question_id, '$[1]') LIKE "%nscri%"  
57           | OR JSON_EXTRACT_SCALAR(question_id, '$[1]') LIKE "%emandeu%"  
58           | and (JSON_EXTRACT_SCALAR(question_id, '$[1]') LIKE "%gence%"  
59             | OR JSON_EXTRACT_SCALAR(question_id, '$[1]') LIKE "%dentifiant%"  
60             | OR JSON_EXTRACT_SCALAR(question_id, '$[1]') LIKE "%uméro%"  
61             | OR JSON_EXTRACT_SCALAR(question_id, '$[1]') LIKE "%allocat%"  
62             | OR JSON_EXTRACT_SCALAR(question_id, '$[1]') LIKE "%conseil%"  
63             | and JSON_EXTRACT_SCALAR(question_id, '$[1]') NOT LIKE "%fera foi%")  
64  
65     | OR (JSON_EXTRACT_SCALAR(question_id, '$[1]') LIKE "%dentifiant%"  
66         | AND JSON_EXTRACT_SCALAR(question_id, '$[1]') LIKE "%rance%"  
67         | AND JSON_EXTRACT_SCALAR(question_id, '$[1]') LIKE "%avail%")  
68   )  
69   OR JSON_EXTRACT_SCALAR(question_id, '$[1]') LIKE "%RQTH%"
```

ods_odoo_crm_survey_question_filtered

```

models > intermediate > ods > ods_odoo_crm_survey_reponse_pivot.sql

1  {{{
2    | config(tags=['odoo', "crm"])
3  }}}
4
5  {% set BQ_TABLE = ref('ods_odoo_crm_survey_question_filtered') %}
6
7  -- model survey_input_ine : creer les colonnes nom, prenom ... à partir des question/reponses
8  SELECT
9    envoi_sondage_id,
10   -- Prenom et nom, on va utiliser cles renseigner dans crm.lead
11   MAX(CASE WHEN titre_question LIKE "%rénom%" THEN reponse END) AS prenom,
12   MAX(CASE WHEN titre_question IN ("Nom", "Nom de naissance", "Nom d'usage / Marital") THEN reponse END) AS nom,
13
14   MAX(CASE WHEN titre_question LIKE "%âge de naissance%" THEN reponse END) AS date_naissance,
15   MAX(CASE WHEN titre_question LIKE "%nationalité%" AND titre_question LIKE "%europ%" THEN reponse END) AS nationalite_europeenne,
16   MAX(CASE WHEN titre_question LIKE "%nationalité%" AND titre_question NOT LIKE "%europ%" THEN reponse END) AS nationalite,
17   MAX(CASE WHEN titre_question LIKE "%numéro de téléphone%" THEN reponse END) AS telephone,
18   MAX(CASE WHEN titre_question LIKE "%QPV%" THEN reponse END) AS qpv,
19   MAX(CASE WHEN titre_question LIKE "%RSA%" and titre_question NOT LIKE "%statut%" THEN reponse END) AS rsa,
20   MAX(CASE WHEN titre_question LIKE "%rance%"
21         AND titre_question LIKE "%ravail%"
22         AND titre_question LIKE "%nscri%"
23         AND titre_question NOT LIKE "%gence%"
24         AND titre_question NOT LIKE "%conseiller%" THEN reponse END) AS inscrit_france_travail,
25   MAX(CASE WHEN
26         (titre_question LIKE "%rance%"
27          AND titre_question LIKE "%ravail%"
28          AND (titre_question LIKE "%nscri%"
29             OR titre_question LIKE "%emandeu%"
30             AND titre_question LIKE "%gence%"
31             OR titre_question LIKE "%dentifiant%"
32             OR titre_question LIKE "%uméro%"
33             OR titre_question LIKE "%allocat%"
34             OR titre_question LIKE "%conseil%"
35             AND titre_question NOT LIKE "%fera foi%")
36
37         OR (titre_question LIKE "%dentifiant%"
38             AND titre_question LIKE "%rance%"
39             AND titre_question LIKE "%ravail%")
40
41         THEN reponse END) AS contact_france_travail,
42   MAX(CASE WHEN titre_question LIKE "%RQTH%" THEN reponse END) AS rqth
43 FROM {{ BQ_TABLE }}
44 GROUP BY envoi_sondage_id

```

ods_odoo_crm_survey_reponse_pivot

FIGURE 3.4.2 – Modèles Ods des données Odoo

3. Table de dimension

J'ai construit la table dimension `dim_odoo_crm_reponse_sondage`, qui combine les données administratives de l'envoi du sondage et les informations personnelles reconstruites dans les ODS.

```

dim_odoo_crm_reponse_sondage.sql ×

models > intermediate > odwh > dim_odoo_crm_reponse_sondage.sql

1  {{{
2  |     config(tags=["odoo", "crm"])
3  }}}
4
5  {% set BQ_TABLE = ref('stg_odoo_survey_user_input') %}
6
7  -- jointure entre STG_input et ods_input_line_reponse via envoi_sondage_id
8
9  SELECT
10   -- 1. Infos du sondage
11   i.envoi_sondage_id,
12   JSON_EXTRACT_SCALAR(i.survey_id, '$[1]') AS nom_sondage,
13   -- 2. Dates sondage
14   i.date_envoi_sondage,
15   i.date_debut_sondage,
16   i.date_fin_sondage,
17   -- 3. Tentatives et identification
18   i.nombre_tentatives_envoi,
19   i.jeton_identification,
20   -- 4. Résultats et notation
21   i.notation_totale,
22   i.pourcentage_notation,
23   i.statut_sondage,
24   -- 5. Infos personnelles
25   r.prenom,
26   r.nom,
27   r.date_naissance,
28   r.telephone,
29   i.email,
30   r.nationalite,
31   r.nationalite_europeenne,
32   r.qpv,
33   r.rqth,
34   r.rsa,
35   r.inscrit_france_travail,
36   r.contact_france_travail
37 FROM {{ BQ_TABLE }} AS i
38 LEFT JOIN {{ ref('ods_odoo_crm_survey_reponse_pivot') }} AS r
39   |   ON i.envoi_sondage_id = r.envoi_sondage_id

```

dim_odoo_crm_reponse_sondage

FIGURE 3.4.3 – Modèles Ods des données Odoo

4. table de fait

Le table `fact_odoo_crm_opportunite_table` est le modèle final utilisé pour le tableau de bord est le modèle de fait. Ce modèle est déjà existant avant même mon arrivé, mais cette fois ci je l'enrichie à partir de la table dimension `dim_odoo_crm_reponse_sondage` que j'ai créée pour centraliser les informations des candidats, notamment les informations issues des questionnaires, le lien vers le CV, le lien vers le fiche candidat et les données de placement (mobilité, QPV, RQTH, etc.), ces données n'était pas incluses dans cette modélisation avant.

Ainsi cette fois-ci le modèle de fait est complet et chaque ligne de cette modèle de fait, renseigne toute information nécessaire aux candidats de Simplon :

```

1 {{ config(materialized='table',tags=['odoo', "crm"] ) }}
2 {% set BQ_TABLE = ref('dim_odoo_opportunite') %}
3
4 SELECT
5   odo.id AS id,odo.type AS type,odo.name AS nom_opportunité,
6   odo.parcours AS parcours,odo.impact_unit AS impact_unit,
7   odo.active AS actif,odo.create_date AS date_creation,odo.stage_id AS etape,
8   odo.date_last_stage_update AS dernière_mise_a_jour_de_l_etape,
9   odo.user_id AS vendeur,odo.function AS poste,odo.answer_id AS answer_id,
10  odo.answer_name AS réponse,odo.answer_token AS jeton_de_la_réponse,
11  odo.survey_url AS url_du_sondage,odo.priority AS priorité,
12  odo.team_id AS équipe_commerciale,odo.campaign_id AS campagne,
13  odo.medium_id AS moyen,odo.source_id AS source,
14  odo.day_open AS jours_pour_assigner,odo.day_close AS jours_pour_cloturer,
15  odo.website AS site_web,odo.display_name AS étiquettes,sts.state AS statut,
16  stlr.name AS motif_de_perte,odo.activity_state AS status_activité,
17  odo.contact_id,odo.partner_id AS client,odo.contact_gender AS genre,
18  odo.contact_name AS nom,odo.contact_firstname AS prénom,
19  odo.phone AS téléphone,odo.mobile AS mobile,odo.region_id AS région,
20  odo.city AS ville,odo.street AS rue,odo.zip AS code_postal,
21  odo.contact_technical_level AS niveau_technique,
22  odo.referred AS référencé,odo.description AS notes_internes,
23  odo.email_from AS email_from,
24
25 -- infos venant de dim_odoo_crm_reponse_sondage
26   dimrs.envoi_sondage_id, dimrs.nom_sondage,dimrs.date_envoi_sondage,
27   dimrs.date_debut_sondage, dimrs.date_fin_sondage,
28   dimrs.nombre_tentatives_envoi, dimrs.jeton_identification,
29   dimrs.notation_totale, dimrs.pourcentage_notation,dimrs.statut_sondage,
30   dimrs.date_naissance,dimrs.email,dimrs.nationalité,
31   dimrs.nationalité_européenne,dimrs.qpv, dimrs.rqth, dimrs.rsa,
32   dimrs.inscrit_france_travail,dimrs.contact_france_travail,
33   -- cv du candidat
34   a.nom_piece_jointe,a.lien_cv
35
36 FROM {{ BQ_TABLE }} odo
37 LEFT JOIN {{ ref('stg_odoo_survey') }} sts
38   ON odo.answer_id = sts.id
39 LEFT JOIN {{ ref('stg_odoo_lost_reason') }} stlr
40   ON odo.lost_reason_id = stlr.id
41 LEFT JOIN {{ ref('dim_odoo_crm_reponse_sondage') }} dimrs
42   ON odo.answer_id = dimrs.envoi_sondage_id
43 LEFT JOIN {{ ref('ods_odoo_crm_ir_attachment') }} a
44   ON odo.id = a.source_id

```

fact_odoo_crm_opportunite_table

FIGURE 3.4.4 – Modèle de fait

L'ensemble de ces modèles est géré dans **DBT** et hébergé dans **BigQuery**. Les données sont mises à jour automatiquement chaque jour.

3.5 Data visualisation

3.5.1 Construction du tableau de bord

Le tableau de bord Looker Studio s'appuie directement sur le modèle de fait. Il permet notamment de filtrer les candidats selon : leur région d'habitation, la région de formation, la ville, le nom de la campagne, le parcours suivi :

Par respect du RGPD, toutes les informations personnelles visibles sur la photo (emails, prénoms et téléphone) ont été masquées.

simplon

PLACEMENT : CANDIDATS EN RECHERCHE D'ENTREPRISE i

Note importante : Comment lire les remarques ?

Le champ **Remarque** est affiché de manière abrégée dans le tableau. Pour consulter le **texte complet** d'une remarque, il suffit de placer le pointeur de la souris sur la cellule correspondante : le **contenu intégral apparaîtra** alors en info-bulle.

Si le texte est trop long, exporter la donnée sinon consulter le candidat directement dans le CRM.

ADMINISTRATIF

CRITERES DE PLACEMENT

Fiche candidat	CV	Prénom	Nom	Formation	Email	Téléphone	Région	Ville	QPV	RSA	RQTH	Contrat alternance	Mobilité	Remarque
Christelle Morello - Test														
			Rémi Dubois - Concepteur Développeur d'Applications DevOps (Alternance Simplon)				Hauts-de-France	La Chapelle d'Armentières			No	Contrat de professionnalisation	Régional	
		<input checked="" type="checkbox"/>	Noui	Adel Noui - FRB2B_WS_AIS_12M	[REDACTED]@gmail.com	[REDACTED]	Auvergne-Rhône-Alpes	Chanaïn			No	Contrat de professionnalisation	Régional	Marlane et l'ancien contrat pris passé 20 ans dans Kotelle en tant que chef responsable logistique mais fermeture
		<input checked="" type="checkbox"/>	Boncompain	Agathe Boncompain - Concepteur Développeur d'Applications DevOps	[REDACTED]n@gmail.com	[REDACTED]	Nouvelle-Aquitaine	Anglet			No	Contrat d'apprentissage	Régional	
		<input checked="" type="checkbox"/>	All	PASTOR	Alexandra PASTOR - Concepteur Développeur d'Applications DevOps	[REDACTED]	Auvergne-Rhône-Alpes	HEYRIEUX			No	Contrat de professionnalisation	Régional	
		<input checked="" type="checkbox"/>	Billon	CAMPUSDIST_TSSR_Surveiller_2025_ALT	[REDACTED]aux-air.fr	[REDACTED]	GSO	Bordeaux	Oui	Non	Non	Contrat d'apprentissage		
		<input checked="" type="checkbox"/>	Cieslar	Anais Cieslar - Concepteur Développeur d'Applications DevOps	[REDACTED]ail.com	[REDACTED]	Hauts-de-France	Lille			Yes	Contrat d'apprentissage	Régional	
		<input checked="" type="checkbox"/>	Diez	Anais Diez - Concepteur Développeur d'Applications DevOps	[REDACTED]com	[REDACTED]	Occitanie	Toulouse			No	Contrat de professionnalisation	Régional	
			LAHBIB	CAMPUSDIST_DataEng_2025_ALT	[REDACTED]ail.com	[REDACTED]	SUD	92110	Non	Non	Non	Contrat de professionnalisation		

1 - 100 / 113 < >

FIGURE 3.5.1 – Tableau de bord de Placement

3.5.2 Atelier de restitution

Après la mise en production du tableau de bord, j'ai animé un atelier de prise en main avec l'équipe métier afin de :

1. présenter la logique de construction,
2. expliquer les filtres et possibilités d'analyse,
3. valider l'adéquation avec les besoins,
4. former les utilisateurs à la navigation.

Travaux transverses

4.1 Résolution des tickets

L'un des rôles les plus importants de l'équipe Data est de répondre aux différents besoins exprimés par les équipes métiers, notamment les demandes liées aux exports et analyses de données. La plupart du temps, les tableaux de bord existants ne permettent pas aux équipes métiers d'analyser ou d'exporter les données souhaitées. Dans ces situations, c'est à l'équipe Data de répondre aux demandes dans les meilleurs délais et avec précision.

Accompagné par ma tutrice, j'ai été amené quotidiennement à répondre à ces demandes en investiguant les différents modèles de données ou réadapter les modèles de données existantes ou même construire des nouvelles modèles de données afin de pouvoir répondre à la demande.

Je vais présenter ici en détail quelques demandes que j'ai répondu :

4.1.1 Données certification Agilité

Demande

Bonjour,

Dans le cadre du renouvellement de la certification Agilité, j'aurais besoin des données suivantes (entre le 01/06/2021 et le 30/09/2025) : typologie des apprenants certifiés en agilité :

1. Répartition entre hommes et femmes ;
2. Âge moyen.

Merci beaucoup !

Je reste à votre disposition si besoin.

Cheffe de projet qualité et certification

Étapes de résolution

1. Compréhension de la demande :

Dans la majorité des cas, les demandes métiers nécessitent une clarification afin d'obtenir un besoin précis et, si nécessaire, de proposer une approche plus adaptée côté DATA. Pour cette demande, les éléments étaient clairs, il n'a donc pas été nécessaire d'organiser un échange supplémentaire.

2. Identification des modèles :

Nous hébergeons un grand nombre de modèles dans *Google Cloud Storage (GCS)*, tous reliés entre eux via des clés uniques. La première difficulté consiste donc à identifier les modèles pertinents pour répondre au besoin.

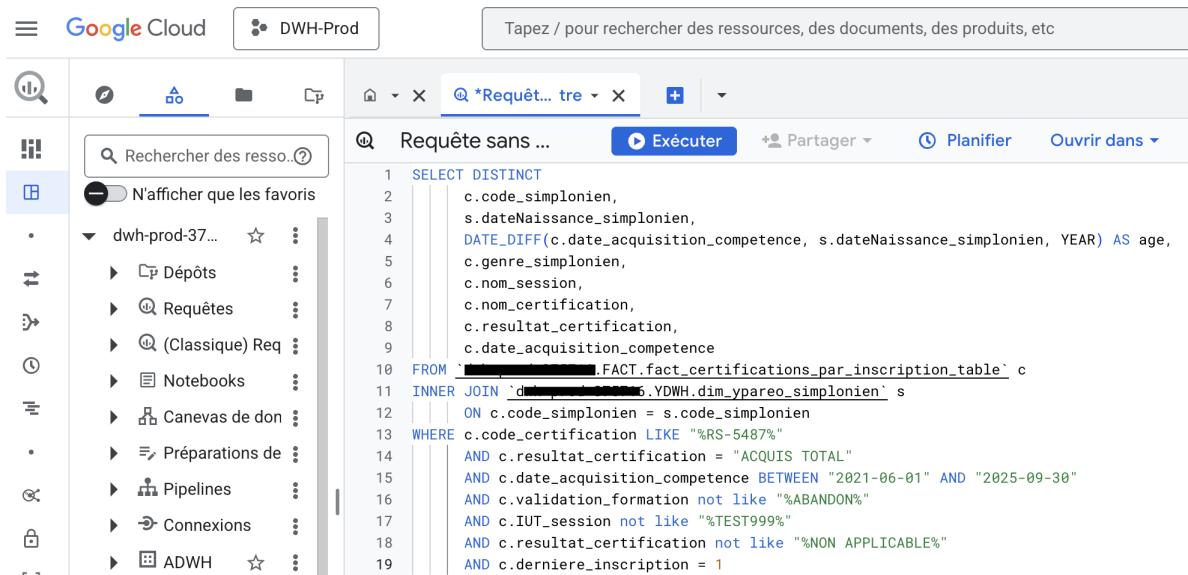
Les modèles mobilisés pour cette demande sont :

- Le modèle de faits `fact_certifications_par_inscription_table`, qui fournit les événements de certification pour chaque apprenant.
- Le modèle de dimension `dim_ypareo_simplonien`, qui apporte les informations complémentaires apprenant (date de naissance, genre, etc.) nécessaires pour enrichir la table de faits.

3. Résolution :

- Extraction des données via requête BigQuery :

Les identifiants sensibles donnant accès à nos projets data ont été volontairement cachés sur la photo pour des raisons de sécurité :



```
SELECT DISTINCT
    c.code_simplonien,
    s.dateNaissance_simplonien,
    DATE_DIFF(c.date_acquisition_competence, s.dateNaissance_simplonien, YEAR) AS age,
    c.genre_simplonien,
    c.nom_session,
    c.nom_certification,
    c.resultat_certification,
    c.date_acquisition_competence
FROM [REDACTED].FACT.fact_certifications_par_inscription_table` c
INNER JOIN `dwh-prod-376.YDWH.dim_ypareo_simplonien` s
ON c.code_simplonien = s.code_simplonien
WHERE c.code_certification LIKE "%RS-5487%"
    AND c.resultat_certification = "ACQUIS TOTAL"
    AND c.date_acquisition_competence BETWEEN "2021-06-01" AND "2025-09-30"
    AND c.validation_formation not like "%ABANDON%"
    AND c.IUT_session not like "%TEST999%"
    AND c.resultat_certification not like "%NON APPLICABLE%"
    AND c.derniere_inscription = 1
```

- Analyse de la répartition via Excel :

A	B	C	D
Pour l'année 2025 (jusqu'à septembre 2025)			
genre_simplonien	Moyenne d'âge	Répartition entre Homme et Femme	Répartition (%) entre Homme et Femme
FEMME	34,43023256	86	36,60%
HOMME	36,29530201	149	63,40%
Total général	35,61276596	235	100,00%
Pour l'année 2024			
genre_simplonien	Moyenne d'âge	Répartition entre Homme et Femme	Répartition (%) entre Homme et Femme
FEMME	34,46341463	82	25,95%
HOMME	36,0982906	234	74,05%
Total général	35,67405063	316	100,00%
Pour l'année 2023			
genre_simplonien	Moyenne d'âge	Répartition entre Homme et Femme	Répartition (%) entre Homme et Femme
FEMME	35,60645161	155	24,11%
HOMME	35,00614754	488	75,89%
Total général	35,15085537	643	100,00%
Pour l'année 2022			
genre_simplonien	Moyenne d'âge	Répartition entre Homme et Femme	Répartition (%) entre Homme et Femme
FEMME	33,57894737	39	32,50%
HOMME	29,60759494	81	67,50%
Total général	30,8974359	120	100,00%
Pour l'année 2021 (à partir de juin 2021)			
genre_simplonien	Moyenne d'âge	Répartition entre Homme et Femme	Répartition (%) entre Homme et Femme
FEMME	34	2	16,67%
HOMME	33,9	10	83,33%
Total général	33,91666667	12	100,00%

4.1.2 Apple Reporting

Demande

Hello,

Dans le cadre de notre partenariat avec Apple, nous avons reçu une demande concernant le suivi des parcours de nos apprenants. Voici les besoins :

1. Nombre d'apprenants Apple Foundation Program ayant poursuivi une formation Simplon (et si possible lesquelles)
2. Nombre d'apprenants Extended Apple Foundation Program ayant poursuivi une formation Simplon (et si possible lesquelles)

Merci beaucoup !

Je reste à votre disposition si besoin.

Coordinatrice nationale Apple Foundation Program

Étapes de résolution

1. Compréhension de la demande :

La demande initiale n'était pas suffisamment précise d'un point de vue DATA. J'ai organisé un échange avec la demandeuse afin de clarifier les objectifs et proposer une structuration adaptée. À l'issue de ce point, nous avons convenu de fournir :

- (a) Le nombre d'apprenants ayant réalisé un AFPE et poursuivi une autre formation.
- (b) Le nombre d'apprenants ayant réalisé un AFP puis poursuivi un AFPE (sans formation intermédiaire).
- (c) Le nombre d'apprenants ayant réalisé un AFP puis poursuivi une autre formation que l'AFPE (en identifiant la première formation suivie après l'AFP).

2. Identification des modèles :

Le modèle mobilisé est : `fact_activite_inscriptions_par_mois_table`, qui décrit mois par mois les activités d'inscription de chaque apprenant.

3. Résolution :

- Extraction des données via requêtes BigQuery :

```

1 ----- Aprennant ayant fait l'AFPE et ont poursuivi une autre formation
2 WITH Suivi_AFPE AS (
3     SELECT
4         code_simplonien,
5         parcours_formation,
6         date_debut_session,
7         date_fin_session
8     FROM [REDACTED].fact_activite_inscriptions_par_mois_table'
9     WHERE LOWER(parcours_formation) LIKE '%apple foundation program extended%'
10        AND derniere_inscription = 1
11        AND validationFormation != "ABANDON"
12        AND code_simplonien != "-1"),
13
14 Formations_apres_AFPE AS (
15     SELECT
16         f.code_simplonien,
17         a.parcours_formation AS parcours_initial,
18         a.date_debut_session AS date_debut_parcours_initial,
19         f.parcours_formation AS parcours_fait_apres_AFPE,
20         f.date_debut_session AS date_debut_suite_parcours,
21         ROW_NUMBER() OVER (PARTITION BY f.code_simplonien
22                           ORDER BY f.date_debut_session) AS ordre
23     FROM [REDACTED].fact_activite_inscriptions_par_mois_table' f
24     INNER JOIN Suivi_AFPE a
25         ON f.code_simplonien = a.code_simplonien
26     WHERE f.date_debut_session > a.date_fin_session
27        AND derniere_inscription = 1
28        AND f.validationFormation != "ABANDON")
29
30     SELECT code_simplonien, parcours_initial, date_debut_parcours_initial,
31           |parcours_fait_apres_AFPE, date_debut_suite_parcours
32     FROM Formations_apres_AFPE
33     WHERE ordre = 1
34     ORDER BY code_simplonien DESC;

```

```

1 -- Aprennant ayant fait 1 AFP et ont poursuivi une formation AFPE
2 -- (directement c'est à dire sans faire une autre formation)
3 WITH Suivi_AFP AS (
4     SELECT
5         code_simplonien, parcours_formation, date_debut_session, date_fin_session
6     FROM [REDACTED].fact_activite_inscriptions_par_mois_table'
7     WHERE LOWER(parcours_formation) LIKE '%apple foundation program%'
8        AND LOWER(parcours_formation) NOT LIKE '%extended%'
9        AND derniere_inscription = 1 AND validationFormation != "ABANDON"
10       AND code_simplonien != "-1"),
11
12 Formations_AFP_apres_AFP AS (SELECT
13     f.code_simplonien,
14     a.parcours_formation AS parcours_initial,
15     a.date_debut_session AS date_debut_parcours_initial,
16     f.parcours_formation AS parcours_fait_apres_AFP,
17     f.date_debut_session AS date_debut_suite_parcours,
18     ROW_NUMBER() OVER (PARTITION BY f.code_simplonien ORDER BY f.date_debut_session) AS ordre,
19     CASE
20         WHEN LOWER(f.parcours_formation) LIKE '%extended%' THEN 'extended'
21         ELSE 'autre'
22     END AS libelle_suivi_formation
23     FROM [REDACTED].fact_activite_inscriptions_par_mois_table' f
24     INNER JOIN Suivi_AFP a
25         ON f.code_simplonien = a.code_simplonien
26     WHERE f.date_debut_session > a.date_fin_session
27        AND derniere_inscription = 1
28        AND f.validationFormation != "ABANDON"
29        ORDER BY code_simplonien DESC)
30
31     SELECT code_simplonien, parcours_initial, date_debut_parcours_initial,
32           |parcours_fait_apres_AFP, date_debut_suite_parcours
33     FROM Formations_AFP_apres_AFP
34     WHERE ordre = 1 AND libelle_suivi_formation = 'extended'
35     ORDER BY code_simplonien DESC;

```

```

    Requête sans titre
    Exécuter
    Partager
    Planifier
    Ouvrir dans ▾

    Tapez / pour rechercher des ressources, des documents, des produits, etc

    1 -- Aprenant ayant fait l'AFP et ont poursuivi une formation autre que AFPE
    2 -- (directement c'est à dire sans faire AFPE)
    3 WITH Suivi_AFP AS (
    4     SELECT
    5         code_simplonien, parcours_formation, date_debut_session, date_fin_session
    6     FROM [REDACTED]fact_activite_inscriptions_par_mois_table'
    7     WHERE LOWER(parcours_formation) LIKE '%apple foundation program%'
    8         AND LOWER(parcours_formation) NOT LIKE '%extended%'
    9         AND dernière_inscription = 1 AND validation_formation != "ABANDON"
    10        AND code_simplonien != "-1"),
    11
    12 Formations_apres_AFP AS (SELECT
    13     f.code_simplonien,
    14     a.parcours_formation AS parcours_initial,
    15     a.date_debut_session AS date_debut_parcours_initial,
    16     f.parcours_formation AS parcours_fait_apres_AFP,
    17     f.date_debut_session AS date_debut_suite_parcours,
    18     ROW_NUMBER() OVER (PARTITION BY f.code_simplonien
    19                     ORDER BY f.date_debut_session) AS ordre,
    20     CASE
    21         WHEN LOWER(f.parcours_formation) LIKE '%extended%' THEN 'extended'
    22         ELSE 'autre'
    23     END AS libelle_suivi_formation
    24     FROM [REDACTED]fact_activite_inscriptions_par_mois_table' f
    25     INNER JOIN Suivi_AFP a
    26     ON f.code_simplonien = a.code_simplonien
    27     WHERE f.date_debut_session > a.date_fin_session
    28     AND dernière_inscription = 1 AND f.validation_formation != "ABANDON"
    29     ORDER BY code_simplonien DESC)
    30
    31     SELECT code_simplonien, parcours_initial, date_debut_parcours_initial,
    32             parcours_fait_apres_AFP, date_debut_suite_parcours
    33     FROM Formations_apres_AFP
    34     WHERE ordre = 1
    35     AND libelle_suivi_formation = 'autre'
    36     ORDER BY code_simplonien DESC;

```

- (b) Analyse des résultats dans Excel : Analyse de l'export de données en tableau croisé dynamique sur Excel :

Nombre aprenants ayant fait l'AFPE et ont poursuivi une autre formation (on a renseigné la première formation suivie après l'AFPE)		
parcours_initial	parcours_fait_apres_AFPE	NB_apprenants
DÉVELOPPEMENT MOBILE IOS - APPLE FOUNDATION PROGRAM EXTENDED	ADMINISTRATEUR/ICE CLOUD - ECOLE CLOUD MICROSOFT BY SIMPLON	1
DÉVELOPPEMENT MOBILE IOS - APPLE FOUNDATION PROGRAM EXTENDED	ARTIS	1
DÉVELOPPEMENT MOBILE IOS - APPLE FOUNDATION PROGRAM EXTENDED	CONCEPTEUR-RICE DÉVELOPPEUR-EUSE D'APPLICATIONS IOS ET OSX - ADVANCED APPLE FOUNDATION PROGRAM	19
DÉVELOPPEMENT MOBILE IOS - APPLE FOUNDATION PROGRAM EXTENDED	CONCEPTEUR-RICE DÉVELOPPEUR SE EN ENVIRONNEMENT DEVOPS	2
DÉVELOPPEMENT MOBILE IOS - APPLE FOUNDATION PROGRAM EXTENDED	DÉVELOPPEMENT MOBILE IOS - APPLE FOUNDATION PROGRAM EXTENDED	1
DÉVELOPPEMENT MOBILE IOS - APPLE FOUNDATION PROGRAM EXTENDED	DÉVELOPPEUR/EUSE INTEGRATEUR/RICE EN RÉALISATION D'APPLICATIONS WEB	1
DÉVELOPPEMENT MOBILE IOS - APPLE FOUNDATION PROGRAM EXTENDED	DÉVELOPPEUR/EUSE RÉALITÉ VIRTUELLE/RÉALITÉ AUGMENTÉE - 3D TEMPS RÉEL	1
DÉVELOPPEMENT MOBILE IOS - APPLE FOUNDATION PROGRAM EXTENDED	DÉVELOPPEUR/EUSE INTELLIGENCE ARTIFICIELLE	1
DÉVELOPPEMENT MOBILE IOS - APPLE FOUNDATION PROGRAM EXTENDED	DÉVELOPPEUR/SE INTELLIGENCE ARTIFICIELLE - ECOLE IA MICROSOFT BY SIMPLON	4
DÉVELOPPEMENT MOBILE IOS - APPLE FOUNDATION PROGRAM EXTENDED	DÉVELOPPEUR/SE WEB ET WEB MOBILE	5
DÉVELOPPEMENT MOBILE IOS - APPLE FOUNDATION PROGRAM EXTENDED	EXPLOITER LA BLOCKCHAIN DANS LE DÉVELOPPEMENT D'APPLICATIONS	1
DÉVELOPPEMENT MOBILE IOS - APPLE FOUNDATION PROGRAM EXTENDED	HACKEUSES	2
DÉVELOPPEMENT MOBILE IOS - APPLE FOUNDATION PROGRAM EXTENDED	MOBILISER LES COMPÉTENCES INFORMATIQUES FONDAMENTALES	5
DÉVELOPPEMENT MOBILE IOS - APPLE FOUNDATION PROGRAM EXTENDED	SAS DE PRÉQUALIFICATION - FILIÈRE DÉVELOPPEMENT D'APPLICATIONS	1
DÉVELOPPEMENT MOBILE IOS - APPLE FOUNDATION PROGRAM EXTENDED	SAS DE PRÉQUALIFICATION - FILIÈRE IA MICROSOFT BY SIMPLON	1
DÉVELOPPEMENT MOBILE IOS - APPLE FOUNDATION PROGRAM EXTENDED	UPSKILLING NON-CERTIFIANT	1
TOTAL		47

Nombre aprenants ayant fait l'AFP et ont poursuivi une AFPE (sans faire une autre formation intermédiaire)		
parcours_initial	parcours_fait_apres_AFPE	NB_apprenants
DÉVELOPPEMENT MOBILE IOS - APPLE FOUNDATION PROGRAM	DÉVELOPPEMENT MOBILE IOS - APPLE FOUNDATION PROGRAM EXTENDED	203

Nombre apprenants ayant fait l'AFP et ont poursuivi autre formation que AFPE (on a renseigné la première formation suivie après l'AFP)		
parcours_initial	parcours_fait_apres_AFP	NB_apprenants
RICE DÉVELOPPEUR/EUSE D'APPLICATIONS IOS ET OSX - ADVANCED APPLE FOUNDATION PROGRAM	CONCEPTEUR-RICE DÉVELOPPEUR/EUSE D'APPLICATIONS IOS ET OSX	8
DÉVELOPPEMENT MOBILE IOS - APPLE FOUNDATION PROGRAM	ACCÉSSIBILITÉ NUMÉRIQUE	1
DÉVELOPPEMENT MOBILE IOS - APPLE FOUNDATION PROGRAM	ADMINISTRATEUR-ICE SYSTÈME DEVOPS	1
DÉVELOPPEMENT MOBILE IOS - APPLE FOUNDATION PROGRAM	ADMINISTRATEUR-ICE CLOUD - ECOLE CLOUD MICROSOFT BY SIMPLON	3
DÉVELOPPEMENT MOBILE IOS - APPLE FOUNDATION PROGRAM	ADMINISTRATEUR-ICE D'INFRASTRUCTURES SÉCURISÉES	4
DÉVELOPPEMENT MOBILE IOS - APPLE FOUNDATION PROGRAM	ARTS	15
DÉVELOPPEMENT MOBILE IOS - APPLE FOUNDATION PROGRAM	ATELIER, SENSIBILISATION, ACCULTURATION	2
DÉVELOPPEMENT MOBILE IOS - APPLE FOUNDATION PROGRAM	AWS RE/START - FONDAMENTAUX DU CLOUD	1
DÉVELOPPEMENT MOBILE IOS - APPLE FOUNDATION PROGRAM	COMPÉTENCES NUMÉRIQUES FONDAMENTALES	7
DÉVELOPPEMENT MOBILE IOS - APPLE FOUNDATION PROGRAM	CONCEPTEUR-RICE DÉVELOPPEUR/EUSE D'APPLICATIONS IOS ET OSX - ADVANCED APPLE FOUNDATION PROGRAM	15
DÉVELOPPEMENT MOBILE IOS - APPLE FOUNDATION PROGRAM	CONCEPTEUR-RICE DÉVELOPPEUR SE D'APPLICATIONS	18
DÉVELOPPEMENT MOBILE IOS - APPLE FOUNDATION PROGRAM	CONCEPTEUR-RICE DÉVELOPPEUR SE EN ENVIRONNEMENT DEVOPS	3
DÉVELOPPEMENT MOBILE IOS - APPLE FOUNDATION PROGRAM	CONSULTANT E DEVOPS	2
DÉVELOPPEMENT MOBILE IOS - APPLE FOUNDATION PROGRAM	DATA ENGINEER	1
DÉVELOPPEMENT MOBILE IOS - APPLE FOUNDATION PROGRAM	DÉVELOPPEMENT MOBILE IOS - APPLE FOUNDATION PROGRAM	93
DÉVELOPPEMENT MOBILE IOS - APPLE FOUNDATION PROGRAM	DÉVELOPPEUR INFORMATIQUE OPTION JEU VIDÉO	1
DÉVELOPPEMENT MOBILE IOS - APPLE FOUNDATION PROGRAM	DÉVELOPPEUR/EUSE INTÉGRATEUR-ICE EN RÉALISATION D'APPLICATIONS WEB	25
DÉVELOPPEMENT MOBILE IOS - APPLE FOUNDATION PROGRAM	DÉVELOPPEUR/EUSE RÉALITÉ VIRTUELLE/RÉALITÉ AUGMENTÉE - 3D TEMPS RÉEL	3
DÉVELOPPEMENT MOBILE IOS - APPLE FOUNDATION PROGRAM	DÉVELOPPEUR-SE CLOUD - ECOLE CLOUD MICROSOFT BY SIMPLON	1
DÉVELOPPEMENT MOBILE IOS - APPLE FOUNDATION PROGRAM	DÉVELOPPEUR-SE DATA	3
DÉVELOPPEMENT MOBILE IOS - APPLE FOUNDATION PROGRAM	DÉVELOPPEUR-SE DATA ET INTELLIGENCE ARTIFICIELLE	8
DÉVELOPPEMENT MOBILE IOS - APPLE FOUNDATION PROGRAM	DÉVELOPPEUR-SE INTELLIGENCE ARTIFICIELLE	3
DÉVELOPPEMENT MOBILE IOS - APPLE FOUNDATION PROGRAM	DÉVELOPPEUR-SE INTELLIGENCE ARTIFICIELLE - ECOLE IA MICROSOFT BY SIMPLON	12
DÉVELOPPEMENT MOBILE IOS - APPLE FOUNDATION PROGRAM	DÉVELOPPEUR-SE WEB ET WEB MOBILE	126
DÉVELOPPEMENT MOBILE IOS - APPLE FOUNDATION PROGRAM	HACKEUSES	28
DÉVELOPPEMENT MOBILE IOS - APPLE FOUNDATION PROGRAM	MÉDIATEUR-ICE NUMÉRIQUE	1
DÉVELOPPEMENT MOBILE IOS - APPLE FOUNDATION PROGRAM	MÉTHODES AGILES DE GESTION ET AMORÇAGE DE PROJET	6
DÉVELOPPEMENT MOBILE IOS - APPLE FOUNDATION PROGRAM	MOBILISER LES COMPÉTENCES INFORMATIQUES FONDAMENTALES	9
DÉVELOPPEMENT MOBILE IOS - APPLE FOUNDATION PROGRAM	OPÉRATEUR-TRICE DE SOLUTIONS DE SÉCURITÉ CLOUD ET HYBRIDE - ECOLE CYBER MICROSOFT BY SIMPLON	2
DÉVELOPPEMENT MOBILE IOS - APPLE FOUNDATION PROGRAM	POEC DATA ANALYST	3
DÉVELOPPEMENT MOBILE IOS - APPLE FOUNDATION PROGRAM	RÉALISER DES APPLICATIONS WEB AVEC UN CMS	11
DÉVELOPPEMENT MOBILE IOS - APPLE FOUNDATION PROGRAM	RÉFÉRENT-E DIGITAL	1
DÉVELOPPEMENT MOBILE IOS - APPLE FOUNDATION PROGRAM	REFUGEKS	1
DÉVELOPPEMENT MOBILE IOS - APPLE FOUNDATION PROGRAM	SAS DE PRÉQUALIFICATION - FILIÈRE CYBERSÉCURITÉ MICROSOFT BY SIMPLON	5
DÉVELOPPEMENT MOBILE IOS - APPLE FOUNDATION PROGRAM	SAS DE PRÉQUALIFICATION - FILIÈRE DÉVELOPPEMENT D'APPLICATIONS	19
DÉVELOPPEMENT MOBILE IOS - APPLE FOUNDATION PROGRAM	SAS DE PRÉQUALIFICATION - FILIÈRE IA	3
DÉVELOPPEMENT MOBILE IOS - APPLE FOUNDATION PROGRAM	SAS DE PRÉQUALIFICATION - FILIÈRE IA MICROSOFT BY SIMPLON	5
DÉVELOPPEMENT MOBILE IOS - APPLE FOUNDATION PROGRAM	SAS DE PRÉQUALIFICATION - FILIÈRE INFRASTRUCTURES	5
DÉVELOPPEMENT MOBILE IOS - APPLE FOUNDATION PROGRAM	SURVEILLER UN SYSTÈME D'INFORMATION SUR DES CRITÈRES DE SÉCURITÉ INFORMATIQUE	1
DÉVELOPPEMENT MOBILE IOS - APPLE FOUNDATION PROGRAM	TECHNICIEN NE D'ASSISTANCE INFORMATIQUE	4
DÉVELOPPEMENT MOBILE IOS - APPLE FOUNDATION PROGRAM	TECHNICIEN NE SUPÉRIEUR SYSTÈMES ET RÉSEAUX	7
DÉVELOPPEMENT MOBILE IOS - APPLE FOUNDATION PROGRAM	TECHNICIEN NE SUPÉRIEUR SYSTÈMES ET RÉSEAUX - SPÉCIALISATION ANALYSE CYBER	2
DÉVELOPPEMENT MOBILE IOS - APPLE FOUNDATION PROGRAM	TESTEUR JEU VIDÉO ASSURANCE QUALITÉ Q/A	1
	TOTAL	470

4.2 Contrôle des données internationales

L'un des plus grands défis de l'équipe internationale consiste à suivre les statistiques de l'entreprise sur ses fabriques à l'international. Chaque pays, notamment en Afrique (ex. : Sénégal), en Europe (ex. : Suisse), au Moyen-Orient (ex. : Liban) et en Amérique latine (ex. : Colombie), renseigne ses données dans un fichier Google Sheet dédié.

L'ensemble de ces fichiers pays est ensuite centralisé dans un unique fichier appelé *Master Sheet*. Ce dernier est utilisé par l'équipe DATA pour construire un tableau de bord permettant à l'équipe internationale de suivre certains indicateurs stratégiques de l'entreprise.

Cette saisie manuelle comporte cependant des risques : erreurs de saisie, oubli de champs, incohérences entre indicateurs ... ect. Ces problèmes dégradent la qualité des données et faussent les indicateurs de pilotage.

4.2.1 Données renseignées dans les fichiers pays

Les fichiers de chaque pays contiennent trois grandes catégories de données :

1. Informations de géolocalisation sur la formation :

Région, Pays et ville.

2. Informations sur la formation :

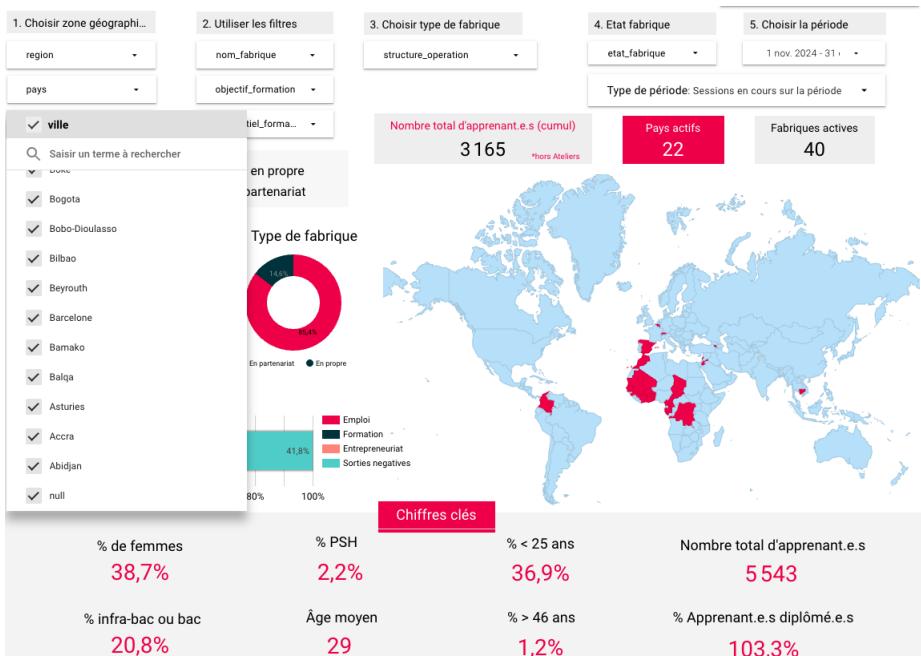
Structure d'opération (Simplon ou partenaire), État de la fabrique (si le centre dde formation est fermée ou active), Public cible (particuliers ou professionnels), Type de formation (métiers, découverte, atelier), Référentiel de formation, Nom de la promotion, Date de début de la formation et Date de fin de la formation.

3. Champs d'indicateurs agrégés :

Nombre total d'apprenants au début (hommes + femmes), Dont nombre total de femmes au début, Nombre total d'apprenants ayant abandonné, Nombre total d'apprenants ayant mené la formation à terme, Dont nombre de femmes ayant mené la formation à terme, Nombre d'apprenants diplômés, Nombre de femmes diplômées, Nombre d'apprenants partiellement diplômés, Nombre de femmes partiellement diplômées, ··· etc.

4.2.2 Problèmes observés dans le tableau de bord existant

À mon arrivée dans l'équipe DATA, le tableau de bord a été construit directement à partir du fichier *Master Sheet* sans contrôle de la donnée renseignées. Cette méthode a entraîné plusieurs anomalies visibles dans les indicateurs :



1. Exemple : un taux de **103,3% de diplômés**, ce qui est un biais. L'erreur provenait des pays ayant renseigné plus de diplômés que d'apprenants ayant mené la formation à terme.
2. Certaines valeurs de filtres, comme la ville « **Nulle** », indiquent que le champ n'a pas été renseigné dans certains fichiers pays.

Ces anomalies faussent fortement les analyses et biaissent la décision.

4.2.3 Initiative de l'équipe Data

On a décidé de mettre en place un système de contrôle automatique permettant :

1. d'identifier les incohérences susceptibles de biaiser les indicateurs ;
2. d'exclure les lignes incorrectes des analyses ;
3. et de remonter les lignes anormales aux équipes internationales afin qu'elles puissent être corrigées rapidement.

Technique de résolution

1. Côté Data Engineer : Un flux Talend a été développé pour récupérer quotidiennement de façon automatique l'ensemble des fichiers pays et de les déposer dans **Google Cloud Storage**.

2. Mon expertise :

- (a) J'ai modélisé les fichiers dans DBT, connecté à BigQuery, afin de centraliser l'ensemble des données dans une base unique et contrôlée.
- (b) J'ai implémenté des règles de contrôle de qualité des données et ajouté une colonne permettant d'indiquer automatiquement chaque anomalie détectée sur la ligne correspondante. Les règles mises en place portent notamment sur :

- i. **Fichier pays vide** : aucun enregistrement n'est présent pour le pays concerné. Il s'agit du cas prioritaire : dès qu'il est détecté, aucun autre contrôle n'est réalisé.
- ii. **Format invalide** : un champ numérique ou une date contient une valeur non conforme.
- iii. **Champ obligatoire manquant** : notamment les champs de géolocalisation (région, pays, ville) et les champs nominatifs liés à la formation.
- iv. **Valeur invalide dans un champ nominatif** : une valeur hors liste déroulante est renseignée (ex. type de formation, public cible, état de la fabrique, etc.).
- v. **Incohérence sur les effectifs** : le nombre final ou le nombre d'abandons dépasse le nombre initial.
$$\text{nombre_final} > \text{nombre_initial} \quad \text{ou} \quad \text{nombre_abandons} > \text{nombre_initial}$$
- vi. **Anomalie sur les validations** : au moins une des conditions suivantes est vraie :
$$\text{nombre_validations} > \text{nombre_final}$$
$$\text{nombre_validation_partielle} > \text{nombre_final}$$
$$\text{nombre_validations} + \text{nombre_validation_partielle} > \text{nombre_final}$$
Cela indique un nombre de validations supérieur au nombre réel d'apprenants présents en fin de parcours.
- vii. **Effectifs féminins incohérents** : les effectifs féminins dépassent les totaux déclarés.
$$\text{nombre_initial_femmes} > \text{nombre_initial}$$
$$\text{nombre_final_femmes} > \text{nombre_final}$$
$$\text{nombre_validations_femmes} > \text{nombre_validations}$$
$$\text{nombre_validation_partielle_femmes} > \text{nombre_validation_partielle}$$
$$\text{nombre_validations_femmes} > \text{nombre_final_femmes}$$
$$\text{nombre_validation_partielle_femmes} > \text{nombre_final_femmes}$$

Résultats obtenus

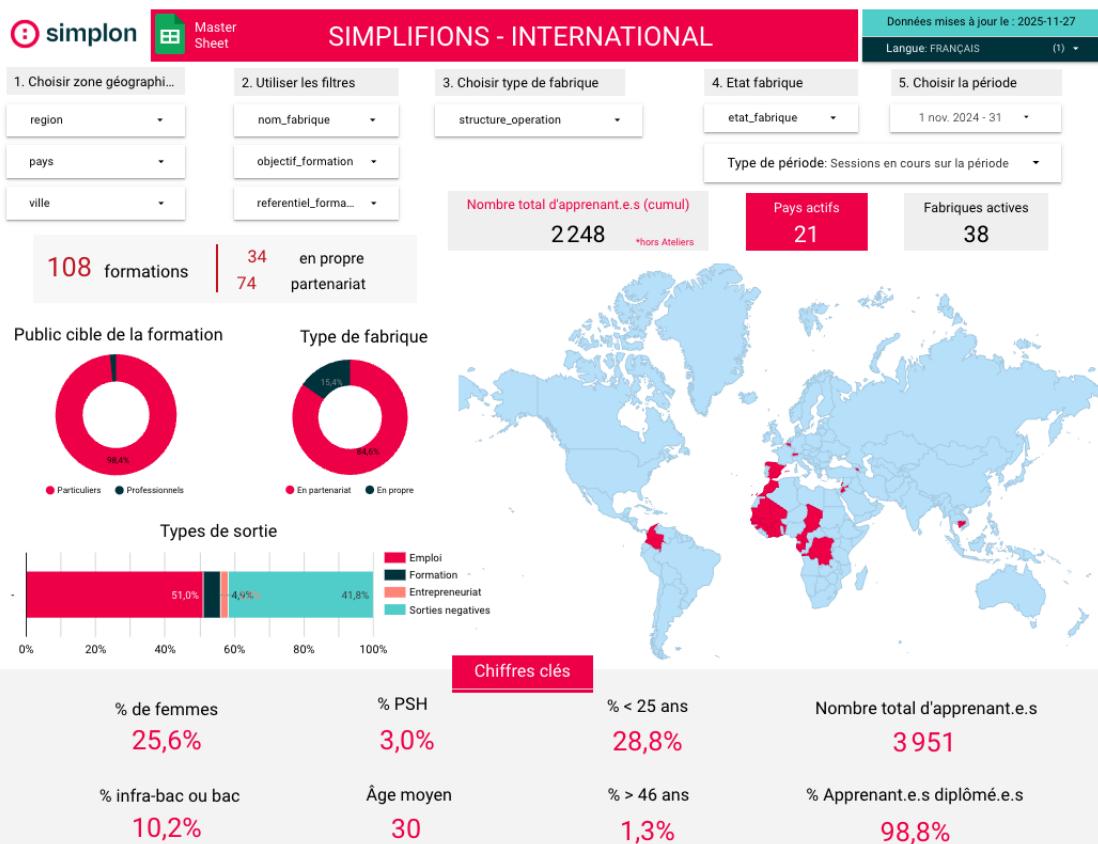
Après la modélisation et la mise en place des contrôles, plusieurs améliorations majeures ont été apportées :

1. Adaptation du tableau de bord à la nouvelle modélisation :

Le tableau de bord utilise désormais uniquement les lignes ne présentant aucune anomalie. J'ai également ajouté un bandeau affichant la fraîcheur des données pour chaque pays (collecte quotidienne via Talend, assurée par le Data Engineer). Le bandeau apparaît en vert si les données sont récoltées à jour, sinon en rouge si elles ne le sont pas, avec l'indication de la dernière mise à jour (ce qui permet aux utilisateurs du tableau de bords de nous alerter en cas de problème de mise à jour de la donnée).

2. Fiabilisation des indicateurs :

Les indicateurs sont désormais calculés uniquement à partir des lignes conformes, ce qui garantit des indicateurs fiables et cohérentes.



3. Création d'une table des anomalies :

J'ai mis en place une table listant toutes les lignes contenant au moins une anomalie ainsi que la nature de l'erreur détectée. Cette table permet aux responsables de saisie d'identifier rapidement les lignes incorrectes et de corriger la donnée dans les fichiers pays :

Message d'anomalie détectée sur la ligne	Région	Pays	Ville	Nom promotion
Au moins un des champs region, ville, pays, nom fabrique, structure operation, etat fabrique, public cible, objectif formation, referentiel formation ou nom de la promotion n'est pas renseigné.	Afrique	Togo		Formateurs Edumérik
Au moins un des champs region, ville, pays, nom fabrique, structure operation, etat fabrique, public cible, objectif formation, referentiel formation ou nom de la promotion n'est pas renseigné.	Afrique	Togo		Edumérik
Au moins un des champs region, ville, pays, nom fabrique, structure operation, etat fabrique, public cible, objectif formation, referentiel formation ou nom de la promotion n'est pas renseigné. et une ou des valeurs de nombre de femmes dépassent les totaux correspondants	Maghreb	Maroc	Tout le Maroc	Projet Kids
une des valeurs de nombre de validations dépasse le nombre final	Afrique	Gabon	Libreville	Ecole 241 Promo 7 Référent Digital
une des valeurs de nombre de validations dépasse le nombre final	Afrique	Gabon	Port-Gentil	Ecole 241 Promo 9 Référent Digital
une des valeurs de nombre de validations dépasse le nombre final	Afrique	Gabon	Libreville	Ecole 241 Promo 10 Développeur Web / Web Mobile
une des valeurs de nombre de validations dépasse le nombre final	Afrique	Gabon	Libreville	Ecole 241 Promo 7 Développeur Web
Au moins un des champs region, ville, pays, nom fabrique, structure operation, etat fabrique, public cible, objectif formation, referentiel formation ou nom de la promotion n'est pas renseigné.	Afrique	Côte d'Ivoire	Abidjan	PSH
Au moins un des champs region, ville, pays, nom fabrique, structure operation, etat fabrique, public cible, objectif formation, referentiel formation ou nom de la promotion n'est pas renseigné.	Afrique	Côte d'Ivoire	Abidjan	P6 DEV
Au moins un des champs region, ville, pays, nom fabrique, structure operation, etat fabrique, public cible, objectif formation, referentiel formation ou nom de la promotion n'est pas renseigné.	Afrique	Côte d'Ivoire	Abidjan	MTN Academy
une ou des valeurs de nombre de femmes dépassent les totaux correspondants	Afrique	Côte d'Ivoire	Abidjan	P1 MONTAGE VIDÉO ET GESTION DE PROJET DIGITAL
Au moins un des champs region, ville, pays, nom fabrique, structure operation, etat fabrique, public cible, objectif formation, referentiel formation ou nom de la promotion n'est pas renseigné.	Afrique	Côte d'Ivoire	Abidjan	PAFESDIA COHORTE 1
Au moins un des champs region, ville, pays, nom fabrique, structure operation, etat fabrique, public cible, objectif formation, referentiel formation ou nom de la promotion n'est pas renseigné. et une des valeurs de nombre de validations dépasse le nombre final	Afrique	Bénin	Cotonou	P1 DCLIC MN Bougelabs Ouidah
Au moins un des champs region, ville, pays, nom fabrique, structure operation, etat fabrique, public cible, objectif formation, referentiel formation ou nom de la promotion n'est pas renseigné. et une des valeurs de nombre de validations dépasse le nombre final	Afrique	Bénin	Cotonou	P1 DCLIC Dev Web Ecole 229 Cotonou
Au moins un des champs region, ville, pays, nom fabrique, structure operation, etat fabrique, public cible, objectif formation, referentiel formation ou nom de la promotion n'est pas renseigné. et une des valeurs de nombre de validations dépasse le nombre final	Afrique	Bénin	Parakou	P1 DCLIC Dev Web Afrikedutech Parakou
une des valeurs de nombre de validations dépasse le nombre final	Afrique	Bénin	Port-Novo	P2 DCLIC MN Simplon

1 - 25 / 25 < >

Conclusion

Au terme de ce stage au sein de l'équipe Data de la direction du Système d'Information de Simplon.co, j'ai pu mesurer l'importance stratégique de la donnée dans le pilotage des activités pédagogiques, administratives et commerciales de l'organisation. Cette expérience m'a permis de participer pleinement à l'amélioration de la qualité, de la fiabilité et de la valorisation des données, en intervenant sur l'ensemble du cycle de la donnée : de la collecte et du contrôle jusqu'à la modélisation et la visualisation.

L'un des apports majeurs de ce stage réside dans la montée en compétence que j'ai pu développer autour des outils structurants de la chaîne data : **Google Cloud Platform**, **DBT**, **Looker Studio** et **Google Sheet**. Leur maîtrise m'a permis de répondre efficacement aux besoins métiers. Le Projet Placement, qui visait à remplacer un fichier Google Sheets manuel par un système automatisé fondé sur les données issues du CRM Odoo, illustre parfaitement cette évolution : il a permis de centraliser les informations, de fiabiliser les indicateurs et d'améliorer la réactivité des équipes commerciales et pédagogiques.

Par ailleurs, la diversité des tickets et des analyses demandées par les équipes m'a offert une vision concrète et transverse des problématiques rencontrées sur le terrain : extraction ciblée de données, contrôles de cohérence, analyses statistiques, structuration de modèles ou mise à jour d'indicateurs. Ces missions ont renforcé ma capacité à comprendre les besoins métiers, à les reformuler précisément et à proposer un traitement adéquat, mêlant rigueur méthodologique et sens du service.

Au-delà des aspects techniques, ce stage m'a permis d'évoluer au sein d'une équipe dont l'organisation, la disponibilité des collègues et les diverses cultures constituent de véritables forces. Les échanges réguliers, les ateliers internes et l'accompagnement de ma tutrice, **Juliana CA-VALCANTE**, cheffe des projets Data, et mon collègue Data Engineer **Antony HENRION** ont grandement contribué à approfondir mes connaissances et à développer une compréhension globale des enjeux data.

Ainsi, cette expérience a confirmé mon intérêt pour les métiers de la data et renforcé ma volonté de poursuivre dans cette voie. Les responsabilités qui m'ont été confiées, les projets auxquels j'ai participé et les compétences que j'ai pu acquérir constituent un socle solide pour la suite de mon parcours professionnel.

Je quitte ce stage avec une expertise enrichie, une vision plus stratégique de la donnée et une motivation renforcée pour contribuer, à l'avenir, à des projets d'analyse, de modélisation et de valorisation des données au service des organisations.

Annexe

Extraction complète ponctuelle de l'**historique du modèle raw_odoo_survey_user_input_line** par **appel API Odoo** à l'aide d'un **script Python** :

```
[ ] import xmlrpclib
import pandas as pd
import os
import shutil
from datetime import datetime
import calendar
import csv

[ ] # environnement odoo

url = 'url vers le serveur odoo'
db = "identifiant du servuer odoo de l'entreprise de Simplon"
email = 'Mon identifiant proffetionnel'
password = 'Mon mot de passe'

common = xmlrpclib.ServerProxy(f'{url}/xmlrpc/2/common')
uid = common.authenticate(db, email, password, {})

if not uid:
    raise Exception("Échec de l'authentification Odoo, vérifie les identifiants.")

models = xmlrpclib.ServerProxy(f'{url}/xmlrpc/2/object')

[ ] # =====
# === FONCTIONS UTILITAIRES
# =====

def odoo_api_get_all(url, db, username, password, model, user_id, fields, filters=None, batch_size=1000):
    """
    Récupère toutes les données du modèle Odoo par batch, avec filtres et traduction FR.
    """
    if filters is None:
        filters = []

    models = xmlrpclib.ServerProxy(f'{url}/xmlrpc/2/object')
    offset = 0
    all_data = []

    while True:
        data = models.execute_kw(
            db, user_id, password,
            model, 'search_read',
            [filters],
            {
                'fields': fields,
                'limit': batch_size,
                'offset': offset,
                'context': {'lang': 'fr_FR'}
            }
        )
        if not data:
            break
        all_data.extend(data)
        offset += batch_size

    return all_data
```

```

[1] ⏪ def clean_relation(value):
    """
    Nettoie les relations Odoo :
    - Many2one -> conserve [id, "libellé"] sous forme de texte
    - Many2many -> liste d'IDs sous forme de chaîne séparée par des virgules
    """
    # Cas Many2one : [id, "Libellé"]
    if isinstance(value, list) and len(value) == 2 and isinstance(value[0], int):
        return str(value) # garde la liste entière comme texte

    # Cas Many2many : [id1, id2, ...]
    if isinstance(value, list) and all(isinstance(v, int) for v in value):
        return ",".join(map(str, value))

    return value

def clean_text(value):
    """
    Supprime les retours à la ligne et tabulations dans les textes """
    if isinstance(value, str):
        return value.replace("\n", " ").replace("\r", " ").replace("\t", " ")
    return value

[2] ⏪ def export_data_to_csv(path, url, db, username, password, model, user_id, fields=[], filters=None, result="display"):
    """
    Exporte les données d'un modèle Odoo en CSV ou retourne un DataFrame nettoyé.
    """
    if filters is None:
        filters = []

    data = odoo_api_get_all(url, db, username, password, model, user_id, fields, filters)

    if not data:
        print("Aucune donnée récupérée.")
        return None

    # Conversion en DataFrame
    df = pd.DataFrame(data)

    # Nettoyage global
    df = df.map(clean_relation).map(clean_text)

    if result == "display":
        return df
    elif result == "export":
        df.to_csv(path, sep=',', index=False, encoding="utf-8", quoting=csv.QUOTE_MINIMAL)
        print(f" Fichier exporté : {path}")
        return df
    else:
        return data

[3] ⏪ # =====
# === FONCTION OPTIONNELLE : SPLIT MANYZONE
# =====
def split_manyzone_columns(df, columns):
    """
    Sépare les colonnes ManyZone (ex: [id, 'nom']) en deux colonnes :
    - <col>_id
    - <col>_name
    """
    for col in columns:
        df[f"{col}_id"] = df[col].apply(lambda x: x[0] if isinstance(x, list) and len(x) == 2 else None)
        df[f"{col}_name"] = df[col].apply(lambda x: x[1] if isinstance(x, list) and len(x) == 2 else None)
    return df

[4] ⏪ # =====
# === EXEMPLE D'UTILISATION
# =====

df_hr_line = export_data_to_csv(
    'J_STG_ODOO_Line.csv',
    url,
    db,
    email,
    password,
    'survey.user_input.line', # identification du modèle technique
    user_id=uid,
    fields=[

        'id',
        'user_input_id',
        'question_id',
        'skipped',
        'display_name',
        'create_date',
    ],
    result="export"
)

# Note :
# Si on veut ensuite séparer user_input_id et question_id :
# df_hr_apprenant = split_manyzone_columns(df_hr_apprenant, ['user_input_id', 'question_id'])
# df_hr_apprenant.to_csv('J_STG_ODOO_GetSurveyUserInputLine_split.csv', index=False, encoding='utf-8')

```