

**UNIVERSITÉ DE RENNES 1 ET DE RENNES 2
MAGISTÈRE STATISTIQUE ET MODÉLISATION
ECONOMIQUE**

MAI - AOÛT



CRÉATION D'UN MODÈLE
BUDGÉTAIRE DES SALAIRES EN VBA
ET
UNIFORMISATION DE DONNÉES EN
SQL

Table des matières

Remerciements	2
Introduction	3
Présentation de l'entreprise	3
Evolution de la société	3
Les Produits.....	5
Contexte du stage	5
1. Traitement de la donnée	6
11 Extraction des données	6
13 Mise en forme des données.....	7
2 Analyse des données.....	9
21 Construction des indicateurs.....	9
211 Les valeurs aberrantes	9
212 Simulation d'une année future	11
23 Construction du tableau résumé.....	12
3 Missions de requêtage SQL.....	14
31 La requête « Article »	14
32 La requête « Pièces et sous-pièces ».....	15
Conclusion.....	16
Les aspects techniques.....	16
Les apports personnels	16
Annexe	17
Annexe 1 : Choix du dossier pour l'importation des données	17
Annexe 2 : Feuilles de données	17
Annexe 3 : Ajout de la colonne Entrées/Sorties	18
Annexe 4 : Mise à jour des ETP2 entrées	19
Annexe 5 : Calcul des salaires annuels ou des employés	19
Annexe 6 : La requête « Pièces et sous-pièces »	20
Bibliographie	23
Résumé	25

Remerciements

Je tiens à exprimer ma profonde gratitude à Philippe, mon maître de stage, pour m'avoir confié ce sujet et pour avoir toujours été disponible quand j'en avais besoin. Ses conseils avisés m'ont permis de m'améliorer et de mieux comprendre les attentes en entreprise.

Je remercie également toute l'équipe, en particulier Ginette, Mikaelle, et Victor, pour leur aide précieuse et leur soutien constant. Leur disponibilité et leur gentillesse ont grandement facilité mon apprentissage et m'ont permis de tirer le meilleur de cette expérience.

Introduction

Présentation de l'entreprise

Evolution de la société

L'entreprise MECADAQ, anciennement Mécanique d'Aquitaine, a été créée en 1971 par Monsieur Vincent DUBECQ, père de l'actuel président du groupe Julien DUBECQ. Le groupe MECADAQ est dans la fabrication et l'assemblage de pièces mécaniques de précision principalement pour le secteur aéronautique, mais aussi dans l'automobile et la domotique.

Initialement basé à Biarritz le groupe a longtemps évolué au cœur de l'industrie du sport automobile en travaillant pour la Formule 1 avec Renault et Peugeot, en complément de leur activité principale ; l'aéronautique.

MECADAQ vient à se développer et s'agrandit en acquérant une filiale à Pessac en 1988 leur permettant de se rapprocher de Dassault. C'est en 2005 que la société quitte ses locaux de Biarritz pour implanter son siège social à Tarnos avec son principal site de production.

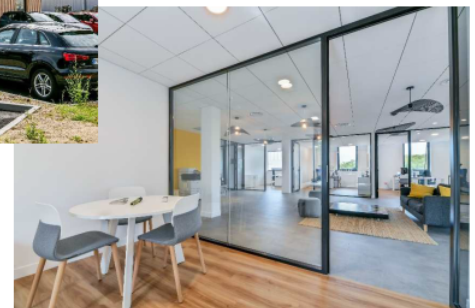


Figure 1 : Siège social de MECADAQ à Tarnos

En 2002, la décision de se recentrer principalement sur l'activité aéronautique leur permet de participer aux programmes majeurs des années 2000 (A380, A350, Falcon 7x, B787...).

La volonté de travailler avec les plus grands acteurs mondiaux les pousse à acquérir une filiale aux Etats-Unis en 2008 pour se rapprocher de Boeing. Stratégie payante car le groupe connaît une augmentation de 400% de 2010 à 2014.

Cette forte volonté de croissance les amène à travailler avec le fond d'investissement Activa Capital depuis 2015 afin de réaliser leurs projets.

Le groupe acquiert alors trois nouveaux sites entre 2016 et 2018 lui permettant de renforcer ses compétences industrielles avec un site spécialisé dans le Tournage/Taillage avec Marignier, mais aussi de consolider ses positions clients : Airbus/Stelia avec le site de Chanteloup-les-Vignes et Boeing avec le site de Hirschler à Kirkland (USA).

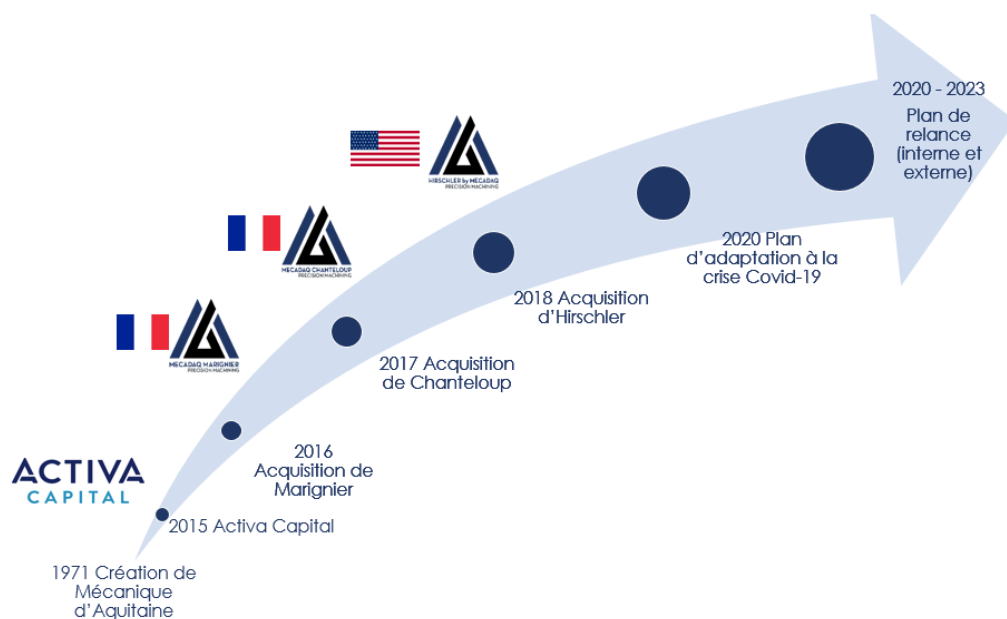


Figure 2 - Evolution de MECADAQ GROUP

Les Produits

Le Groupe MECADAQ est spécialisé dans l'usinage de pointe de pièces complexes entrant dans la composition d'ensembles techniques dans le domaine de l'aéronautique principalement, telles que :

- Des composants de portes : poignées, axes, cranks, lockers
- Des systèmes hydrauliques
- Des composants de structures de fuselages/ailes
- Des composants des nacelles (attaches des moteurs d'avion aux ailes)
- Des systèmes d'engrenages participant à la transmission d'éléments rotatifs

Contexte du stage

Pendant mon stage, j'ai intégré le service Finance-RH¹ composé de cinq personnes. Ce service gère les ressources humaines, la comptabilité et les finances de l'entreprise. J'ai également collaboré avec un Data Analyste qui m'a assisté lorsque nécessaire.

Il est crucial pour une entreprise de suivre les salaires et le nombre d'employés pour s'adapter aux variations des commandes. Ma première mission a consisté à créer un module budgétaire des salaires afin de prévoir les dépenses jusqu'à la fin de l'année 2024. Ce module permet à l'utilisateur de simuler des embauches, des licenciements ou encore des augmentations de salaire.

Le modèle budgétaire inclut les salaires bruts, les charges salariales et les ETP² (équivalents temps plein). L'objectif est de détecter des anomalies, comme un mois avec un nombre inhabituel d'employés sans augmentation de la production. Pour ce faire, les données doivent être récoltées sur une plateforme de ressources humaines (ADP), puis formatées et intégrées dans un tableau de bord récapitulatif. Toutes ces étapes ont été codées en VBA (Visual Basic for Applications) pour une accessibilité et une utilisation optimale.

Une fois ce projet terminé, j'ai travaillé sur le regroupement uniforme de données entre les différentes sociétés. Pour permettre une analyse uniforme rapide et efficace. Pour réaliser cette tâche, j'ai codé en SQL dans le système de gestion de base de données DBeaver⁵.

1. Traitement de la donnée

Les données dont nous avons besoin ne serviront pas que à la modélisation du Dashboard⁴ final, elles serviront aussi à l'équipe comptable pour bien suivre les comptes payés. Il faut donc que les variables finales aient un nom cohérent et qui parlent à tout le monde.

1.1 Extraction des données

J'ai malheureusement du travailler en langage VBA, qui ne fait pas partie des langages enseignés durant le master. Il était essentiel que les données soient accessibles et manipulables par toute personne le souhaitant, et que les résultats soient affichés sur une feuille Excel.

Après avoir vu les données brutes une première fois, j'ai remarqué qu'il y avait une énorme partie sur le traitement de ces dernières à réaliser. J'ai d'abord pensé à traiter les données directement depuis ADP³ grâce à une API. Mais à cause du fort coût pour avoir cette dernière, il n'a pas été possible de l'obtenir. J'ai d'abord pensé à utiliser le logiciel Python mais pour cela il aurait fallu que quelqu'un ait constamment Python sur son ordinateur pour ainsi pouvoir faire de la maintenance sur le code. Enfin, mon maître de stage a préféré que je code en VBA pour tout avoir sur le même classeur et ne pas à avoir à faire des mises à jour ou autre sur un logiciel qu'il ne connaissait pas.

L'importation et la mise en forme des données prenait beaucoup de temps pour être faites à la main. Il faut donc rendre cette tâche mensuelle, répétitive, automatique. Une fois les données obtenues sous format .xlsx il faut les importer et les mettre en forme dans un même classeur Excel. Nous analyserons les salaires et/ou ETP² en fonction de la nature du contrat, du site ou autre si besoin.

13 Mise en forme des données

Une fois que les données sont dans un même dossier il faut pouvoir les ramener dans notre classeur. Pour cela c'est à l'utilisateur d'aller pointer dans le dossier qu'il souhaite (*Annexe 1*). Une fois les feuilles importées, il faut les mettre en forme. Effectivement il y a une grosse partie du travail à faire là-dessus. Comme vous pouvez le voir à l'*Annexe 2* les feuilles Excel n'ont pas du tout la même présentation. Une fois les feuilles mieux ordonnées, il manque des informations pour les intérimaires ainsi que pour les employés d'Hirschler.

Pour cela il faut se servir de nouvelles feuilles qui servent de tables de correspondances. Nous sommes obligés de les utiliser car la paye des intérimaires se fait différemment de celle des salariés. Les intérimaires sont payés à l'heure et la paye change en fonction des cabinets d'intérim. Pour récupérer ces informations je me suis servi de la fonction VBA Vlookup. Cette dernière fonctionne comme la fonction Excel RECHERCHEV.

```
Application.VLookup(nom_interim, ThisWorkbook.Sheets("table th").Range("A1:K" & lastRow_tableth), 11, False)
```

Figure 3 : Fonction VLookup

En premier paramètre il y a le nom de l'intérimaire pour qui on recherche l'information, puis le tableau dans lequel les informations souhaitées se trouvent et en fin le numéro de la colonne dans laquelle se trouve ce que l'on souhaite (ici le salaire). On peut remarquer que l'espace dans lequel on va chercher les informations n'est pas fixé. Effectivement, de nouveaux intérimaires peuvent arriver à n'importe quel moment il faut donc prévoir que le code s'adapte à cette modification.

Il y a aussi eu des erreurs humaines qui ont été effectuées sur les données, des accents oubliés, des caractères spéciaux rajoutés ou enlevés (ex : « - », « ' », ...), il a donc du fallu créer une fonction qui permette de corriger toutes ces erreurs pour ne pas se tromper dans l'analyse.

```
Function RemoveAccents(str As String) As String
    Dim AccChars As String
    Dim RegChars As String
    Dim i As Long

    ' Definition des accents avec leurs correspondances
    AccChars = "ÀÁÂÃÄÅÆÇÈÉÊËÌÍÎÏÐÑÒÓÔÕÖØÙÚÛÜÝÞßàáâãäåæçèéêëìíîïðñòóôõöøùúûüýþÿçñšž"
    RegChars = "AAAAAAEEEEEEEEIIIIIOOOOOOooouuuuuyycnsZaaaaaaeeeeeeeeiiiiioooooouuuuuyycnsz "

    ' On remplace l'accent par son caractère qui n'en comporte pas
    For i = 1 To Len(AccChars)
        str = Replace(str, mid(AccChars, i, 1), mid(RegChars, i, 1))
    Next i

    RemoveAccents = str
End Function
```

Figure 4 : Nettoyage des erreurs humaines

La fonction RemoveAccents prend une chaîne de caractères en entrée et remplace les caractères accentués par leurs équivalents non accentués. Elle définit deux chaînes de caractères : AccChars, qui contient les caractères accentués, et RegChars, qui contient les caractères correspondants sans accents. Ensuite, elle parcourt chaque caractère accentué dans AccChars et utilise la fonction Replace pour remplacer ce caractère par le caractère équivalent dans RegChars. Enfin, elle renvoie la chaîne modifiée sans accents.

Une fois que toutes les données sont bien importées et mises en forme nous obtenons une colonne de plus qu'annoncé précédemment. En effet il a fallu créer une colonne pour le salaire brut en dollars. Ainsi la communication avec les Etats-Unis se fera plus simplement et clairement si besoin. Pour cela, ce sera à la comptable de transmettre, et de rentrer (dans une boîte de commande qui apparaît lors de l'importation), le taux de change chaque mois lors de l'importation pour que les comparaisons soient faites le plus fidèlement possible.

2 Analyse des données

Il y a deux points importants à analyser dans les données ; les salaires et les ETP. Pour cela, il faut construire en VBA deux tableaux croisés dynamiques (dans deux feuilles différentes) avec en ligne les noms des employés, la société à laquelle ils sont rattachés, leur contrat de travail, leur imputation et enfin une nouvelle colonne qui indique quand la personne est entrée ou sortie avec l'année de sortie (« E2024 » si la personne est entrée en 2024 par exemple) (*Annexe 3*). En colonne se trouve tous les mois de l'année et enfin dans les valeurs il y a les données, donc les salaires ou les ETP. Dans la feuille du tableau croisé dynamique des salaires il y a aussi les charges patronales qui peuvent aussi être prises en compte dans l'analyse.

21 Construction des indicateurs

211 Les valeurs aberrantes

Après avoir créé les tableaux croisés dynamiques nous devons faire des prévisions sur les mois restants de l'année 2024. Pour ce faire nous devons d'abord rechercher les salariés qui ont démissionné et les salaires qui paraissent aberrants pour que cela n'impacte pas les prévisions salariales de chaque employé.

Deux procédures ont été créées. L'une permettant de détecter les personnes qui sont encore dans la société. Pour ces personnes-là il a fallu faire un lien entre le nom sur le tableau croisé dynamique et la feuille source de données et vérifier s'il y a une date de sortie (si ce n'est pas le cas, alors la personne n'a pas pour projet de quitter la société) et que cette date soit inférieure à celle d'aujourd'hui. Pour ces personnes-là, leur ligne de données sera coloriée en rouge pour pouvoir les repérer visuellement.

La deuxième procédure permet de détecter les valeurs aberrantes de chaque personne. Un seuil a été fixé à 30%. Ce qui signifie que si le salaire du mois x de la personne est inférieur ou supérieur à 30% de son salaire moyen alors on mettra la cellule, respectivement, en bleu ou en rouge.

Il y a aussi un autre cas qu'il faut mettre en lumière pour pouvoir le repérer facilement. Effectivement il y a un 13^e mois de prime qui est réparti entre juillet et décembre. Donc lorsqu'une colonne représente le mois de juillet ou de décembre on va la colorier en jaune.

L'Etablissement	Entrées/Sorties	1_2023	2_2023	3_2023	4_2023	5_2023	6_2023	7_2023	8_2023	9_2023	10_2023	11_2023	12_2023
MECADAQ GROUP	E2011	5 017	5 017	7 017	5 017	5 017	5 017	6 517	5 017	5 017	12 425	5 017	7 517
MECADAQ GROUP	S2023					633	1 667	1 747	-				

Figure 5 : Exemple des salaires mensuels pour deux personnes en 2023

Sur l'exemple ci-dessus on peut remarquer que pour la première personne il y a deux mois en rouge, ce qui signifie qu'elle a été « anormalement » payée supérieurement à la normale. Tandis que pour celle d'en dessous c'est en bleu donc elle a été payée

inférieurement à d'habitude. On remarque aussi que la ligne est en jaune ce qui signifie que la personne est partie de la société. L'analyse se fait donc rapidement et facilement.

Tous ces indicateurs colorés sont utiles dans l'analyse visuelle des résultats mais aussi pour nos prédictions. En effet, si nous sommes en juin, alors nous avons toutes les données de paye jusqu'au mois de mai. Nous devons donc faire les prévisions sur 7 mois pour terminer l'année. Nous allons donc ajouter une colonne à chaque fois pour chacun des mois, puis pour chaque ligne mettre la dernière valeur présente. Mais cela que si la cellule est de couleur blanche pour ne pas prendre des valeurs aberrantes dans les prédictions. S'il n'y a que des valeurs aberrantes pour une personne alors on fera la moyenne des trois derniers mois de paye.

Cela permet une analyse rapide et intuitive des données, pour voir s'il y a des incohérences dans ces dernières. Si on sait à quoi elles sont dues, on sait qu'il faudra les prendre en compte lors de l'analyse. En revanche, si on ne sait pas les expliquer alors il peut s'agir d'anomalies, donc il faut voir avec l'équipe Finance pour confirmer cela. Mais tout ce travail d'analyse et d'interprétation est réalisé par le DAF donc on ne s'étendra pas sur le sujet ici.

Il faut aussi pouvoir se projeter plus loin dans le temps. Cela permet de mieux prévoir le personnel en fonction des commandes. Effectivement si le nombre de commandes augmente, alors il faudra recruter du personnel et voir ou on en serait avec le schéma d'aujourd'hui permet d'avoir un ordre d'idée.

212 Simulation d'une année future

Pour simuler l'année 2025 il faut faire attention à se baser sur les bonnes données. Grâce aux indicateurs que nous avons créé précédemment on peut jouer avec les fonds colorés des cellules pour ne pas sélectionner les valeurs aberrantes ou arranger les choses.

```
lastRow = .Cells(.Rows.count, 4).End(xlUp).Row
lastCol = .Cells(4, .Columns.count).End(xlToLeft).Column

' On rajoute une ligne a la table des etp pour que ce soit a la meme hauteur que les deux autres tables
' Puisqu'il y a la ligne de la somme des charges en plus dans les FCST
If .Name = "ETP" Then
    ws.Rows(2).Insert Shift:=xlDown, CopyOrigin:=xlFormatFromLeftOrAbove
End If

' Mois actuel et année actuelle
Mois = Month(Date)
Annee = Year(Date) + 1

' On insère 13 colonnes
For col = 13 To 1 Step -1
    .Columns(lastCol + 1).Insert
    If col = 13 Then
        .Cells(4, lastCol + 1).Value = "Total " & Annee
        .Columns(lastCol + 1).Interior.color = RGB(217, 217, 217)
    Else
        ' On les nomme
        nom_col = "P_" & col & "_" & Annee
        .Cells(5, lastCol + 1).Value = nom_col
        .Cells(5, lastCol + 1).Interior.color = RGB(255, 255, 255)
        .Cells(4, lastCol + 1).Interior.color = RGB(255, 255, 255)
        .Cells(3, lastCol + 1).Interior.color = RGB(255, 255, 255)
        If Left(nom_col, 4) = "P_1_" Then
            .Cells(4, lastCol + 1).Value = Annee
        End If
    End If
End If

' On les colorie
For ligne = 6 To lastRow
    If col <> 13 And .Cells(ligne, lastCol + 1).Interior.color <> RGB(0, 0, 0) Then
        .Cells(ligne, lastCol + 1).Interior.color = .Cells(ligne, 1).Interior.color
        .Cells(ligne, lastCol + 1).Value = .Cells(ligne, lastCol - 3).Value
    Else
        .Cells(ligne, lastCol + 1) = 0
    End If

    ' On corrige les colonnes quand y a les primes
    If .Name = "FCST" Then
        If col = 7 Or col = 12 Then
            .Columns(lastCol + 1).Interior.color = RGB(255, 255, 0)
            .Cells(ligne, lastCol + 1).Value = .Cells(ligne, lastCol - 3).Value * 1.5
        End If
    End If
Next ligne
Next col
```

Figure 6 : Année suivante

Pour créer le nombre de colonne suffisante et les remplir, j'ai ajouté les colonnes et les ai remplis au fur et à mesure. Il faut ajouter les 12 colonnes correspondantes aux chaque mois de l'année puis la colonne du total de l'année. Il faut donc faire attention lorsqu'on les nomme et à bien ajouter la prime qu'il y a pour les mois de juillet et décembre.

23 Construction du tableau résumé

Avant de regarder les anomalies en détail il faut avoir une vision générale et simple des résultats. Le Dashboard⁴ permet d'avoir une analyse efficace. Il est aussi l'interface de commande, là où on peut gérer les importations de données et les simulations.

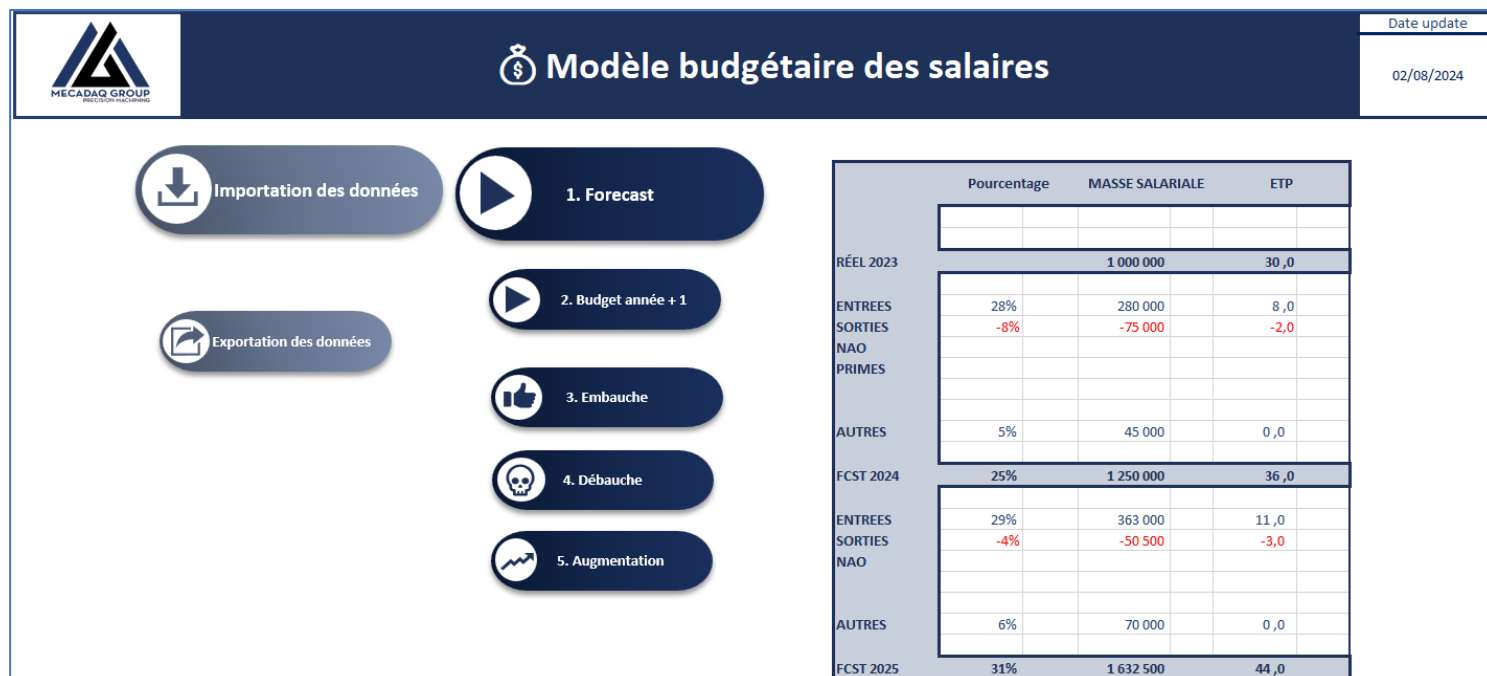


Figure 7 : Dashboard⁴ (Données factices)

On peut retrouver sur le tableau tous les boutons permettant de gérer les simulations c'est-à-dire les embauches, débauches, augmentation et la simulation pour une année supplémentaire dont nous avons de parlé précédemment. Nous allons nous intéresser au tableau qui se trouve à droite de l'image. Ce qui se trouve dans les lignes bleus sont les informations principales et le reste les explications.

Par exemple, on remarque que dans l'entreprise il y a 30 employés qui ont coûtés 1M€ en 2023. Entre les années 2023 et 2024 il y a eu 8 entrées (à 280 000€) et 2 sorties (à 75 000€). Ce qui fait que l'on prévoit en fin 2024 une masse salariale de 1,25M€ pour 36 employés. Dans la ligne « Autres » se trouve les informations « inexpliquées », elle représente la raison de pourquoi on irait voir plus précisément dans les données. Ici on remarque qu'il y a 45 000€ d'inexpliqué. Dans cet exemple, cela est dû à la prime annuelle des employés lors de l'année 2023 qui était de 5%.

Pour pouvoir avoir les informations du Dashboard, qui se mettent à jour au fur et à mesure des modifications il a fallu être précis et ruser pour contourner certains problèmes. C'est pour cela que la colonne « Entrées/Sorties » a été créée (*Annexe 4*). Grâce à cette dernière on peut mettre des filtres sur les tableaux de données, faire des sommes et récupérer les résultats. Tout d'abord il faut créer une nouvelle colonne pour chaque année avec dedans la somme des salaires ou la moyenne des ETP² pour chaque employé (*Annexe 5*). Mais il faut aussi faire la distinction entre les charges patronales et le salaire brut, lors de

```

' Boucle sur chaque colonne à partir de la 2e (puisque la 1ère colonne contient le texte)
For j = 2 To lastCol
    ' Initialiser la plage de somme pour chaque colonne
    Set sumBrut = Nothing

    ' Boucle sur chaque ligne à partir de la 6e
    For i = 6 To lastRow
        If ws.Cells(i, 1).Value = "Somme de Brut S.S." Then
            If sumBrut Is Nothing Then
                Set sumBrut = ws.Cells(i, j)
            Else
                Set sumBrut = Union(sumBrut, ws.Cells(i, j))
            End If
        End If
    Next i

    ' Si une plage a été trouvée, insérer la formule SUBTOTAL
    If Not sumBrut Is Nothing Then
        ws.Cells(1, j).Formula = "=SUBTOTAL(109," & sumBrut.Address & ")"
    End If
Next j

' Définir les valeurs et formater les nouvelles lignes
ws.Cells(1, 4).Value = "Total filtré des Brut S.S."
ws.Rows(1).Interior.color = RGB(217, 217, 217)
ws.Rows(1).Font.Bold = True ' Écriture en gras

' Calcul des totaux pour les colonnes spécifiques
For i = 1 To lastCol
    If Left(ws.Cells(4, i).Value, 5) = "Total" Then
        ws.Cells(1, i).FormulaR1C1 = "=SUM(RC[-12]:RC[-1])"
    End If
Next i

```

cette somme, qui est présent dans les données. Donc sur la feuille qui contient le tableau des salaires nous allons ajouter deux lignes en haut de la feuille qui contiendront respectivement les sommes mensuelles des salaires brut et des charges patronales. Il faut que les résultats de ces lignes soient dynamiques pour que si l'utilisateur veuille par sur les employés qui sont en CDI les résultats restent cohérent. La fonction Excel « SUBTOTAL » permettait de répondre à cette attente.

La fonction « SUBTOTAL » doit donc s'effectuer sur toute les lignes qui comportent les salaires bruts. Or, un problème est apparu. J'avais aussi utilisé cette fonction pour faire la somme annuelle des salaires pour chaque employé. Or cette fonction ne peut s'appelait elle-même. C'est pour quoi on utilise plutôt la fonction « SUM » lorsqu'une colonne contient un intitulé « Total ».

Une fois que tous ces indicateurs fonctionnent nous obtenons un modèle budgétaire des salaires interactifs. Ce modèle est simple et est flexible aux changements de données que l'utilisateur juge nécessaire. L'importation de données se fera manuellement tous les mois. Dans le classeur Excel il y a en plus un guide d'utilisation (*Annexe 6*) permettant à de nouveaux utilisateurs de le manier ou de se rappeler à quoi servent les indicateurs.

3 Missions de requêtage SQL

Durant ce stage, j'ai eu l'opportunité de travailler sur un autre projet concernant l'unification des données au sein de l'entreprise MECADAQ. MECADAQ est une entreprise composée de plusieurs sites, chacun utilisant des systèmes ERP⁶ différents. Cette disparité pose un problème de cohérence des données, rendant difficile l'exploitation d'informations pertinentes à l'échelle de l'entreprise. Mon rôle a consisté à unifier les données du site de Hirschler aux États-Unis, afin de les intégrer dans une base de données commune.

Pour identifier les informations pertinentes parmi les 200 tables de données disponibles sur ce site, j'ai collaboré étroitement avec un ingénieur de la Supply Chain⁷. Cet ingénieur possédait une connaissance approfondie des tables susceptibles de m'être utiles. Bien que les requêtes SQL nécessaires aient été relativement simples à exécuter, la véritable difficulté résidait dans la compréhension des données et de la structure des tables.

J'ai principalement réalisé deux requêtes SQL. Ces requêtes ont pour objectif de suivre les flux d'articles, de produits finis ou de matières premières dans les stocks, et de calculer le MRP⁸ (Material Requirements Planning).

31 La requête « Article »

La requête suivante permet de lister les articles en fonction de leur date de livraison prévue, de la quantité commandée, de la quantité disponible en stock, ainsi que de la quantité requise. Elle relie trois tables principales : V_ORDER_LINES, INVENTORY_MSTR, et V_ORDER_HEADER.

```
SELECT
  V_ORDER_LINES.DATE_ITEM_PROM AS "Due Date",
  V_ORDER_LINES.ORDER_NO AS "Order No",
  V_ORDER_LINES.RECORD_NO AS "Record No",
  V_ORDER_HEADER.CUSTOMER_PO AS "Customer PO",
  V_ORDER_LINES.CUSTOMER,
  V_ORDER_LINES.PART,
  V_ORDER_LINES.QTY_BO AS "Qty ordered",
  INVENTORY_MSTR.QTY_ONHAND AS "Qty On Hand",
  INVENTORY_MSTR.QTY_REQUIRED AS "Qty required"
FROM
  V_ORDER_LINES
LEFT JOIN
  INVENTORY_MSTR ON V_ORDER_LINES.PART = INVENTORY_MSTR.PART
LEFT JOIN
  V_ORDER_HEADER ON V_ORDER_LINES.ORDER_NO = V_ORDER_HEADER.ORDER_NO
WHERE
  V_ORDER_LINES.QTY_BO <> 0
ORDER BY
  V_ORDER_LINES.PART ASC,
```

Figure 9 : "Article"

V_ORDER_LINES.DATE_ITEM_PROM ASC;

Elle sélectionne des colonnes clés, telles que la date de livraison prévue, le numéro de commande, le client, la quantité commandée, la quantité en stock, et la quantité requise. La requête filtre les résultats pour ne montrer que les lignes de commande où la quantité commandée n'est pas nulle ($QTY_BO <> 0$). Les résultats sont ensuite triés par référence d'article (*PART*) et par date de livraison prévue.

32 La requête « Pièces et sous-pièces »

Cette requête est conçue pour gérer les pièces qui sont composées de sous-pièces, lesquelles peuvent également être des pièces mères d'autres sous-pièces, créant ainsi une hiérarchie complexe. Pour cela, la même table est jointe plusieurs fois afin de traiter ces relations.

(Requête en [annexe 6](#) à cause de sa taille)

La requête commence par sélectionner les informations relatives aux pièces principales (*PARENT*) et à leurs sous-pièces (*PART*). Elle récupère des données telles que la quantité commandée, la quantité disponible en stock, et la quantité requise. Elle calcule également la quantité totale disponible en stock, en commande ou en cours de traitement pour chaque sous-pièce.

La requête fait plusieurs jointures sur la même table *V_BOM_MSTR*, car certaines sous-pièces peuvent également être des pièces mères, ce qui nécessite de parcourir plusieurs niveaux de la hiérarchie. En joignant la table *V_BOM_MSTR* plusieurs fois, la requête peut ainsi explorer les différentes couches de cette celle-ci. Voici quelques éléments clés de la requête :

- Jointure sur la table *V_BOM_MSTR* : La table est jointe trois fois (*BM*, *BM2*, *BM3*) pour gérer les différents niveaux de sous-pièces.
- Agrégation des quantités : La requête utilise des fonctions d'agrégation (*AVG*, *SUM*) pour calculer la quantité totale disponible en stock ou en commande, en tenant compte de toutes les sous-pièces.
- Filtrage des résultats : La requête filtre les pièces dont la quantité commandée n'est pas nulle, puis classe les résultats par référence de sous-pièce (*HW*) et par date de besoin.

Cette requête est essentielle pour gérer les inventaires de manière précise, surtout dans un environnement où les pièces sont organisées en structures complexes avec plusieurs niveaux de sous-ensembles.

Conclusion

Les aspects techniques

Durant ce stage, j'ai eu l'opportunité de développer et de perfectionner mes compétences techniques, particulièrement en VBA et SQL, des outils essentiels pour automatiser et optimiser les processus financiers et RH au sein de l'entreprise. L'une des réalisations majeures a été la conception d'un module budgétaire des salaires, capable de simuler divers scénarios de gestion du personnel, tels que les embauches, les licenciements, et les augmentations de salaire. Ce module permet à l'entreprise de prévoir avec précision ses dépenses jusqu'à la fin de l'année 2024, offrant ainsi une meilleure gestion de ses ressources. En parallèle, j'ai contribué à la simplification et à l'automatisation de tâches répétitives, garantissant une fiabilité accrue des données et une réduction significative des erreurs manuelles. Ces projets m'ont permis d'acquérir une compréhension approfondie des outils de gestion de données et de leur application dans un contexte réel, renforçant ainsi mes compétences techniques et ma capacité à résoudre des problèmes complexes.

Les apports personnels

Sur le plan personnel, ce stage a été une expérience enrichissante qui m'a permis de développer des qualités essentielles telles que l'adaptabilité, la rigueur, et la gestion du temps. Travailler avec des outils techniques non enseignés dans mon cursus, comme le VBA, m'a appris à sortir de ma zone de confort et à apprendre de manière autonome, renforçant ainsi ma capacité à m'adapter à de nouveaux environnements professionnels. De plus, j'ai gagné en confiance en moi, en particulier dans ma capacité à mener des projets complexes et à collaborer efficacement avec différents services de l'entreprise. Ce stage m'a également permis de comprendre l'importance de la communication claire et précise, notamment lorsqu'il s'agit de présenter des données techniques à un public non spécialisé. Enfin, cette expérience a affiné mon sens de l'organisation, des compétences qui me seront indéniablement utiles dans ma future carrière.

Annexe 1 : Choix du dossier pour l'importation des données

Annexe 2 : Feuilles de données

[illegible]17

Annexe 3 : Ajout de la colonne Entrées/Sorties

```
Function AjoutColonneEntreesSorties(sheetName As String)
    ' Déclarer les variables pour la dernière ligne et la dernière colonne
    Dim lastRow As Long, lastRowListing As Long
    Dim ws, wsListing As Worksheet
    Dim i As Long, j As Long
    Dim nom, contrat As String
    Dim dateEntree As Date, dateSortie As Date
    Dim CurrentYear, NextYear As Long

    ' Référencer les feuilles spécifiées
    Set ws = ThisWorkbook.Sheets(sheetName)
    Set wsListing = ThisWorkbook.Sheets("Listing filtré")

    ' Insérer une colonne pour "Entrées/Sorties" dans la feuille "FCST"
    ws.Columns("G:G").Insert Shift:=xlToRight
    If ws.Name <> "ETP" Then
        ws.Cells(3, "G").Value = "Entrées/Sorties"
        ligne_begin = 4
    Else
        ws.Cells(4, "G").Value = "Entrées/Sorties"
        ligne_begin = 5
    End If

    ' Obtenir la dernière ligne utilisée dans les deux feuilles
    lastRow = ws.Cells(ws.Rows.count, "D").End(xlUp).Row
    lastRowListing = wsListing.Cells(wsListing.Rows.count, "A").End(xlUp).Row

    ' Parcourir chaque ligne de "FCST" à partir de la ligne_begin
    For i = ligne_begin To lastRow
        nom = ws.Cells(i, "D").Value
        contrat = ws.Cells(i, "C").Value
        ws.Cells(i, "G").Value = "" ' Initialiser la cellule avec une valeur vide

        ' Parcourir chaque ligne de "Listing filtré" pour trouver le nom correspondant
        For j = 2 To lastRowListing

            If wsListing.Cells(j, "A").Value = nom And wsListing.Cells(j, "L").Value = contrat Then

                dateEntree = wsListing.Cells(j, 6).Value
                dateSortie = wsListing.Cells(j, 7).Value

                ' Vérifier si l'année de la date de sortie est l'année courante
                If Year(dateSortie) > 2020 Then ' Il n'y a pas de date de sortie supérieur a 2020, et on dirait qu'il y a des "fauss
                    ws.Cells(i, "G").Value = "S" & Right(dateSortie, 4)
                    Exit For ' Sortir de la boucle si une sortie est trouvée
                ElseIf "E" & Right(dateEntree, 4) = "E0:00" Then ' Quand la cellule est "vide" y'a ca... donc on remplace par du vide
                    ws.Cells(i, "G").Value = ""
                Else
                    ws.Cells(i, "G").Value = "E" & Right(dateEntree, 4)
                    Exit For ' Sortir de la boucle si une entrée est trouvée
                End If
            End If
        Next j
    Next i
End Function
```

Annexe 4 : Mise à jour des ETP² entrées

La procédure est bien plus longue mais répétitive, il faut procéder de la même manière pour les sorties et la masse salariale.

```
' Fonction permettant de calculer la masse salariale
Sub EntreeSortie()

    ' Déclaration des variables
    Dim ws As Worksheet
    Dim lastRow As Long
    Dim i, ligne As Long, sommeSorties As Double, sommeEntrees As Double
    Dim col As Long
    Dim PastYear, CurrentYear, NextYear As Variant
    Dim rng As Range

    CurrentYear = Year(Date)
    PastYear = CurrentYear - 1
    NextYear = CurrentYear + 1

    ' Définir la feuille de calcul sur laquelle vous travaillez
    Set ws = ThisWorkbook.Sheets("ETP")

    ' Parcourir les colonnes de la troisième ligne
    lastRow = ws.Cells(ws.Rows.count, "A").End(xlUp).Row

    ligne = 3
    ' Trouver la colonne contenant l'année
    col = -1
    For i = 1 To ws.Cells(3, ws.Columns.count).End(xlToLeft).Column
        If ws.Cells(3, i).Value = "Total " & CurrentYear Then
            col = i
            Exit For
        End If
    Next i

    'ETP ENTREEE
    ' Appliquer le filtre pour les années actuelles et passées
    Sheets("ETP").Activate
    tabName = "Tableau_Forecast_ETP"
    ActiveSheet.ListObjects(tabName).Range.AutoFilter Field:=7, _
        Criteria1:="E" & CurrentYear, Operator:=xlOr, Criteria2:="=E" & PastYear

    ' Calculer la différence et copier la valeur
    Debug.Print col
    Range("A2").Value = ws.Cells(1, col).Value - ws.Cells(1, col - 13).Value + (ws.Cells(2, col).Value - ws.Cells(2, col - 13).Value)
    Range("A2").Select
    Selection.Copy
    Sheets("Dashboard").Select
    Range("T15").Select
    Selection.PasteSpecial Paste:=xlPasteValues, Operation:=xlNone, SkipBlanks:=False, Transpose:=False
```

Annexe 5 : Calcul des salaires annuels ou des employés

```
' Mets à jour les totaux des années et NON des mois
Function MJTotaux(sheetName As String)
    Dim ws As Worksheet
    Dim lastRow As Long
    Dim lastCol As Long
    Dim i As Long

    ' Définir la feuille de travail basée sur le nom passé en paramètre
    Set ws = ThisWorkbook.Sheets(sheetName)

    With ws
        ' Déterminer la dernière colonne non vide dans la quatrième ligne
        lastCol = .Cells(4, .Columns.count).End(xlToLeft).Column

        ' Déterminer la dernière ligne non vide dans la colonne correspondant à lastCol et ajouter 1 pour la ligne de total
        lastRow = .Cells(.Rows.count, lastCol).End(xlUp).Row + 1

        ' Recalculer la colonne avec le total général
        ' Parcourir chaque ligne pour calculer la somme des cellules
        For i = 4 To lastRow
            .Cells(i, "S").FormulaR1C1 = "=SUBTOTAL(109,RC[-12]:RC[-1])"
            .Cells(i, "AF").FormulaR1C1 = "=SUBTOTAL(109,RC[-12]:RC[-1])"
        Next i
    End With
End Function
```

Annexe 6 : La requête « Pièces et sous-pièces »

```
-- 2_ORDER_LINES (2)
SELECT BM."Due date",
       BM."Order No",
       BM."Record No",
       BM."PARENT" AS "PART",
       BM."Qty ordered",
       BM."Qty on hand",
       BM."Qty required",
       BM."Category",
       BM.QUANTITY,
       BM.PART AS "HW",
       AVG(BM."Qty on order / in process (HW)") + SUM(BM."Qty on hand") + SUM(BM."QTY_SUB") AS
"Qty on hand / on order / in process (HW)",
       BM."Qty ordered (HW)",
       BM."Date needed"
FROM (
  SELECT DISTINCT OL.PART AS "PARENT",
         BM.PART,
         BM.QUANTITY,
         OL.QTY_BO AS "Qty ordered",
         OL.QTY_BO * BM.QUANTITY AS "Qty ordered (HW)",
         DATE_ITEM_PROM AS "Due date",
         ORDER_NO AS "Order No",
         DATEADD(day, -IM.TIME_MATERIAL_LEAD, DATE_ITEM_PROM) AS "Date needed",
         RECORD_NO AS "Record No",
         BM.CATEGORY AS "Category",
         COALESCE(PL.QTY_REQUIRED, 0) + COALESCE(JO.QUANTITY_ACTUAL, 0) AS "Qty on order / in
process (HW)",
         IM.QTY_ONHAND * BM.QUANTITY AS "Qty on hand",
         (COALESCE((SELECT QTY_ONHAND FROM V_INVENTORY_MSTR WHERE PART = BM.PART), 0)
          + COALESCE((SELECT QTY_ONHAND FROM V_INVENTORY_MSTR WHERE PART = BM2.PART2),
0)
          + COALESCE((SELECT QTY_ONHAND FROM V_INVENTORY_MSTR WHERE PART = BM3.PART3),
0)) AS "QTY_SUB",
         IM.QTY_REQUIRED AS "Qty required"
  FROM V_ORDER_LINES OL
  LEFT OUTER JOIN V_INVENTORY_MSTR IM ON OL.PART = IM.PART
  LEFT OUTER JOIN (
    SELECT PART, QUANTITY, CATEGORY, PARENT
    FROM V_BOM_MSTR
    WHERE QUANTITY <> 0
      AND PART IS NOT NULL
      AND PART <> '          '
      AND PARENT <> PART
    ) BM ON OL.PART = BM.PARENT
```

-- On joint la table plusieurs fois, car certaines pieces filles sont aussi des pieces meres, où leurs filles sont aussi des pieces meres, etc...

```
LEFT OUTER JOIN (
  SELECT PART AS PART2, QUANTITY, CATEGORY, PARENT AS PARENT2
  FROM V_BOM_MSTR
  WHERE QUANTITY <> 0
    AND PART IS NOT NULL
    AND PART <> '      '
    AND PARENT <> PART
)BM2 ON BM.PART = BM2.PARENT2
LEFT OUTER JOIN (
  SELECT PART AS PART3, QUANTITY, CATEGORY, PARENT AS PARENT3
  FROM V_BOM_MSTR
  WHERE QUANTITY <> 0
    AND PART IS NOT NULL
    AND PART <> '      '
    AND PARENT <> PART
)BM3 ON BM2.PART2 = BM3.PARENT3
LEFT OUTER JOIN (
  SELECT PART, UM_PURCHASING,
    SUM(CASE
      WHEN UM_PURCHASING = 'FT' THEN (QTY_ORDER - QTY_RECEIVED) * 12
      ELSE QTY_ORDER - QTY_RECEIVED
    END) AS QTY_REQUIRED
  FROM V_PO_LINES
  WHERE FLAG_RECV_CLOSE <> 'Y'
    AND QTY_ORDER - QTY_RECEIVED NOT IN (0, 1)
  GROUP BY PART, UM_PURCHASING
  ORDER BY PART
) AS PL ON BM.PART = PL.PART
LEFT OUTER JOIN (
  SELECT JO.PART,
    JO.UM,
    SUM(JO.HOURS_ACTUAL) AS QUANTITY_ACTUAL
  FROM (
    SELECT PART, UM, HOURS_ACTUAL, JOB, SUFFIX, FLAG_CLOSED
    FROM V_JOB_OPERATIONS
    WHERE UM NOT IN (' ', 'HR')
      AND HOURS_ACTUAL <> 0
      AND PART <> 'O/P'
    ) AS JO
  LEFT OUTER JOIN (
    SELECT JOB, SUFFIX, QTY_ORDER
    FROM JOB_HEADER
    WHERE JOB NOT IN ('IND21', 'IND23', 'INDBAL', 'ZBAL', 'ZIND')
      AND DATE_CLOSED = '000000'
    ) AS JH ON JO.JOB = JH.JOB AND JO.SUFFIX = JH.SUFFIX
  WHERE JH.QTY_ORDER IS NOT NULL
```

```

        AND JO.FLAG_CLOSED = 'Y'
    GROUP BY JO.PART, JO.UM
    ORDER BY JO.PART
) AS JO ON BM.PART = JO.PART
WHERE BM.PART IS NOT NULL
) AS BM
WHERE BM."Qty ordered" <> 0
GROUP BY BM.PART,
    BM."Qty ordered (HW)",
    BM."Due date",
    BM."Date needed",
    BM."Record No",
    BM."Category",
    BM."Order No",
    "PART",
    BM."Qty ordered",
    BM."Qty on hand",
    BM."Qty required",
    BM.QUANTITY,
    BM."QTY_SUB"
ORDER BY "HW", BM."Due date";

```

Bibliographie

1. Finance-RH (Finance – Ressources humaines) : Le service compte une RH, une comptable, une stagiaire, un alternante et enfin le directeur administratif financier (le maitre de stage). Plus précisément, lors de ce stage je me suis occupé de la partie finance des ressources humaines.
2. ETP (équivalent temps plein) : Si une personne arrive en août 2024 égale à 0,33. Mais son ETP sera égale à 1 pour l'année 2025 si la personne reste bien les 12 mois.
3. ADP : est un fournisseur mondial complet de solutions de gestion du capital humain.
4. Dashboard : est un ensemble d'indicateurs de performance (KPIs) qui permettent de suivre une activité, si possible en temps réel, afin de réagir en conséquence le plus rapidement possible.
5. DBeaver : est un logiciel permettant l'administration et le requêtage de base de données. Pour les bases de données relationnelles.
6. ERP : est un type de logiciel que les entreprises utilisent pour gérer leurs activités quotidiennes telles que la comptabilité, les achats, la gestion de projets, la gestion des risques et la conformité, ainsi que les opérations de Supply Chain.
7. Ingénieur Supply Chain : assure le pilotage stratégique et opérationnel des approvisionnements de l'entreprise.
8. MRP (Material Requirements Planning) : signifie "planification des besoins en matières" en anglais. Il s'agit d'un système de planification de la production utilisé pour gérer les besoins en matières premières, les composants et les produits finis dans une entreprise.

Résumé

Ce rapport de stage présente mon expérience au sein du service Finance-RH de l'entreprise MECADAQ, spécialisée dans l'aéronautique. Durant cette période, j'ai été chargé de développer un module budgétaire des salaires visant à améliorer la gestion prévisionnelle des ressources humaines. Ce projet m'a permis de maîtriser des outils techniques tels que VBA et SQL, tout en contribuant à l'automatisation et à l'optimisation des processus financiers. Le stage m'a également apporté une compréhension approfondie des enjeux financiers et organisationnels dans un contexte industriel. Il a renforcé ma capacité à m'adapter à des technologies nouvelles et à gérer des projets complexes, tout en développant des compétences personnelles essentielles comme l'autonomie, la rigueur et la communication.