MISSIONS 2EME ANNEE -RH TRANSPORT

Durant ma deuxième année d'apprentissage j'ai travaillé en mode **projet** sur une application créée depuis **Windev**, un atelier de génie logiciel permettant la création d'application en **Wlangage** mais aussi accessoirement en Java, python, visual Basic et C. L'application de mon entreprise nommé **Simulation RH** depuis sa création utilise le langage propre de **Windev**: le **Wlangage**.

I. Contexte

Simulation RH a été créée pour simplifier le calcul des **DFS**, Déduction forfaitaire spécifique (pour frais professionnels). Réduction d'une partie de l'assiette des cotisations et contributions de Sécurité sociale, considérée comme représentative de frais professionnels, accordée à certaines professions, pour certaines catégories de salariés, dans la limite d'un certain montant.

Lors de mon arrivée, j'ai travaillé à partir d'une version existante, on m'a demandé de reprendre le projet et de le retravailler avec mon maitre d'apprentissage. L'objectif est de terminé le **projet** pour **Aout 2024.** Durant notre travail, nous avons utilisé les outils de projet que propose **Windev** nommé « **Centre de suivi de projet** » afin de se départager les tâches plus facilement et de répertorier toute notre avancée.

÷	Affectée à 🔎	¢ Projet د	° Titre	ر Durée estimée ر	* Temps passé _p *	Date dispo 🔎 🕨
1	fiona.fauvelle	Simulation RH DFS	Raffraichir Tableau TABLE_BULLCOTISATION_DFS	30 minutes	15 minutes	07/09/2023
1	fiona.fauvelle	Simulation RH DFS	Ancrage - Table - Colonne	60 minutes	43 minutes	08/09/2023
1	fiona.fauvelle	Simulation RH DFS	Modification Requête REQ_MILIEUBULL	180 minutes	11 heures	15/09/2023
1	fiona.fauvelle	Simulation RH DFS	Création d'un état récapitulatif	120 minutes	6 heures	20/09/2023
1	fiona.fauvelle	Simulation RH DFS	Créer Bouton Suppression Salarié	60 minutes	4 heures et 3 mi	28/09/2023
1	fiona.fauvelle	Simulation RH DFS	Bouton suppression d'un client	60 minutes	0 jour	28/09/2023
1	fiona.fauvelle	Simulation RH DFS	Ancrage des tables	60 minutes	1 heures et 41 m	29/09/2023
1	fiona.fauvelle	Simulation RH DFS	Créer bouton modifier Client	60 minutes	9 heures	04/10/2023
/	fiona.fauvelle	Simulation RH DFS	Créer bouton modifier Salarié	60 minutes	41 minutes	04/10/2023
1	fiona.fauvelle	Simulation RH DFS	Fusionner "Recalcul" et "Recalcul manuel" en 1 seul bouton	120 minutes	30 minutes	04/10/2023
1	fiona.fauvelle	Simulation RH DFS	Harmoniser PLan 1 - Boutons	60 minutes	12 minutes	04/10/2023
/	fiona.fauvelle	Simulation RH DFS	Affecter couleur si modification "TABLE BULL_COTISATION"	64 minutes	3 heures et 34 m	04/10/2023
/	fiona.fauvelle	Simulation RH DFS	Interdire création Bulletin avec plus d'un mois d'écart	180 minutes	8 heures	06/10/2023
1	fiona.fauvelle	Simulation RH DFS	Enregistrement du cumul des heures quand on enregistre une simulation	240 minutes	2 heures et 27 m	06/10/2023
1	fiona.fauvelle	Simulation RH DFS	Création d'un GOTO	1 heure	59 minutes	25/10/2023
	fiona.fauvelle	Simulation RH DFS	Creation d'un GOTO	1 heure	59 minutes otal durée restante :	25/10/2023 1 jour. 7 heure

J'ai principalement travaillé sur **Windev23** (Ancienne version de Windev, actuellement nous sommes à la version **29 – Windev2024**) ainsi que sur la **base de données** et création/Modification de **requête SQL.**

2023/2024

Fiona FAUVELLE

Premier Semestre П.

A. Création d'un CLOUD pour le projet

Durant le premier semestre, j'ai dû effectuer des recherches sur l'installation d'un serveur afin d'avoir le projet en ligne et non en local comme il était initialement. Après mes recherches, nous avons entamé les démarchés pour créer notre GDS, gestionnaire de source à l'aide de Windev : https://doc.pcsoft.fr/?2038001 qui proposait un GDS.



Après validation, nous avons crées des utilisateurs personnels qui avaient accès au projet. Le principe est de modifier le projet où nous pouvons extraire une partie (cela peut être un onglet, une requête SQL, etc...) du projet en local, puis dès que la modification est terminée le réintégrer afin que le projet se met à jour. Il y a une sécurité qui interdit que deux personnes aient intégrer le même bout de projet afin d'éviter d'éventuels problèmes de sauvegarde.



autre utilisateur. Je devais soit attendre qu'il finisse, soit je pouvais également extraire la fenêtre pour

Page **2** sur **25**

2023/2024

effectuer des tests, mais les modifications que j'apportais n'étaient pas enregistré. A chaque connexion, nous devions nous connecter sur notre compte utilisateur ainsi que sur le **cloud** de l'entreprise, et sélectionné le projet que nous souhaitons modifier durant notre session :



B. Etude de la base de données

J'ai par la suite dû étudier la base de données :



Ceci est la **première** version de la BDD, elle a été modifiée par la suite. Cette partie de la **base de données** regroupe essentiellement les Salariés et Client, les créations de bulletins de salaires et des paramètres de cotisations.

Fiona FA	AUVELLE
	a création d'une table sur windou est très simple d'utilisation :

	DElementsBulletin Drore Gypticable CodeElementBulletin ContElementBulletin CotisationPatronale CotisationSaariale AvriableCalcul	Nom	Type Talle Id. automatique ● Numérique 4 Booléen 1 Texte 50 Booléen 1
(+) @ _M	O Général	Orancé ≧ Etats et Requêtes Rubrique non reliée à un Métatype	Le fonctionnement de cette partie de la base de données était fonctionnel, je n'ai pas eu besoin de la modifier. J'ai simplement appris les fonctionnalités de windev sur la BBD. J'ai cependant pu créer/modifier des requêtes SQL sur ses tables durant mon premier semestre.
	Sous type : Clé Non clé Clé uniqu Clé avec Paramètres o Sensible à Sensible à Tableau	Identifiant automatique (8 octets) Sens de parcours Ascendant Descendant doublons de l'index et de la recherche pour clé texte — à la casse à l'accentuation aux espaces, ponctuation et car. spéciaux Dimension 0 Taille réelle	C'est lors de mon deuxième semestre, où j'ai créé la suite de la base de données que nous verrons plus tard.

2023/2024

Page **4** sur **25**

C. Fenêtre Accueil – Premières lignes de code

1. Prise en main sur la fenêtre et ses fonctionnalités

La **première** version de **Simulation RH** contenait qu'une seule fenêtre avec plusieurs plans, la fenêtre **accueil :**





Les **fenêtres** peuvent avoir plusieurs plans (illimité), pour cette **fenêtre** elle a un total de **11 plans** (Certaines sont vides par suite de changements faits au **second semestre**). Nous pouvons modifier l'ordre et utiliser un plan de référence : c'est là où nous mettons le menu de la fenêtre pour qu'elle s'affiche sur tout les plans et futurs plans de la **fenêtre**.



2023/2024

Cette version a été modifié depuis. J'ai effectué les modifications pour que l'application soit plus **à l'image de l'entreprise** (de couleur rouge et noir comme le logo RH Transport) et **ergonomique** pour la présentation de la première version faite en novembre 2023 lors d'une réunion.

Durant les modifications, j'ai eu plusieurs missions :

- Créer un bouton « modifier un salarié » ;
- Créer un bouton « modifier un client »;
- Créer un bouton « supprimer un salarié » ;
- Créer un bouton « supprimer un client ».

Clic sur BIN_MODIFIERClient1 Si Erreur : par programme Quand Exception : par programme											
gnDirectionbouton=2											
sNomduSalarie, sPrenomduSalarie, sTypeduSalaire, sDateduSalarie, sCoeffduSalarie sont une chaîne											
DEG SELECT MODIESALADIE Idealanda - TADLE DEG LISTEALADIE COL IDEAlanda											
KCQ_SELECI_MODIFSALARLE.IGSALAFIE = IABLE_KCQ_LISISALARLE.COL_IDSALAFIE											
gintigneencours = REV_SELECT_MOUTSALAALE_TOSALATE											
H2S PAS HEXECUTEREQUETE(REQ_SELECI_MODIFSALARIE, hRequeteDefaut) ALORS											
Erreur("Erreur d'initialisation de la requête"+RC+HErreurInfo())											
RETOUR											
SINON											
日											
HLitPremier(REQ_SELECT_MODIFSALARIE)											
sNomduSalarie = REQ_SELECT_MODIFSALARIE.Nom											
sPrenomduSalarie = REQ_SELECT_MODIFSALARIE.Prénom											
sTypeduSalaire = REQ_SELECT_MODIFSALARIE.TYPE											
sDateduSalarie = REQ_SELECT_MODIFSALARIE.DateEntree											
sCoeffduSalarie = REQ_SELECT_MODIFSALARIE.Coeff											
LFIN											
OuvrePopup(FEN_PopupSalarie, gnDirectionbouton, gnLigneencours, Null, sNomduSalarie, sPrenomduSalarie, sTypeduSalarie, sDateduSalarie, sCoeffduSalarie)											

Chaque fonctionnalité utilisait des **requêtes SQL** que je créé en amont, il est possible d'utilisé les fonctionnalités de **windev** afin que le logiciel nous fasse la requête automatiquement. Mais pour mon apprentissage, les requêtes devaient être écrit et je ne devais pas utiliser les outils de **windev**.

```
Code SQL de REQ_SELECT_MODIFSALARIE

Exemple des requêtes SQL.

Exemple des requêtes SQL.
```

```
2023/2024
```

Code SQL de **REQ_UPDATE_SALARIE**

SET

```
Salarie.NomSalarie = {NomSalarie},
Salarie.PrenomSalarie = {PrenomSalarie},
Salarie.TypeConducteur = {TypeSalarie},
Salarie.DateEntree = {DateEntree},
Salarie.Coeff = {CoeffSalarie},
Salarie.ContratHeure = {HeureSalarie}
```

WHERE

```
Salarie.IDSalarie = {IDSalarie}
```

Code SQL de REQ_DELETE_SALARIE

.....

DELETE FROM Salarie

WHERE Salarie.IDSalarie={IdSalarie}

```
Code SQL de REQ_SELECT_MODIFCLIENT
```

Client.NomClient, Client.Cp, Client.Ville, Client.Syndicat, Client.IDEffectif

FROM

Client

WHERE

```
Client.IDClient = {IDClient}
```

```
Code SQL de REQ_UPDATE_CLIENT
UPDATE Client
```

```
SET
Client.NomClient = {NomClient},
Client.Cp = {Cp},
Client.Ville = {Ville},
Client.Syndicat = {Syndicat},
Client.IDEffectif = {Effectif}
```

WHERE

Client.IDClient = {idClient}

Page **7** sur **25**

Code SQL de REQ_DELETE_CLIENT

DELETE FROM Client

WHERE Client.IDClient={idClient}



2023/2024

Fiona FAUVELLE

2. Création de conditions

La première condition était pour la création de bulletin. Je devais créer plusieurs conditions qui bloque la création de bulletin si :

- Le bulletin existe déjà sous la date entrée ;
- Le bulletin à un mois d'écart avec celui d'avant ;
- Le bulletin est inférieur à la date d'entrée du Salarié.

Pour cela on m'a demandé de noter l'algorithme que j'ai imaginé, voici la première version :

```
SI (CONDITION DOUBLON) = VRAI
       Info("")
       RETOUR
SINON
       SI DATEENTREE_SALARIE = {ParamDateBulletin} ALORS
       Crée le bulletin
       SINON
                SI DATEENTREE_SALARIE > {ParamDateBulletin} ALORS
                info("")
                RETOUR
                SINON
               SI REQUETE DU MOIS PRECEDENT = Oui ALORS
                Crée le bulletin
                STNON
                       SI DATEENTREE SALARIE > 01 Janvier ALORS
                       Crée le bulletin
                       SINON
                       SI DATEBULLETIN = 01(janvier) ALORS
                       Crée le bulletin
                       SINON
                       Info("")
                       RETOUR
                       FTN
                       FIN
(Je rentre le 1er Fevrier, et je rentre le bulletin le 1er Mars. il existe un bulletin au 1er Février.)
(Je rentre le 1er Fevrier et je saisi mon bulletin du 1er Février)
(Je suis rentrée le 1février et je saisi mon bulletin du premier janvier)
(Je suis rentrée le 1er Janvier, et je fais un bulletin au 1er Janvier)
(Je suis rentrée le 1er Décembre et je fais un bulletin au 1er Janvier)
       Et voici le résultat final :
     CITC 30 DIM_OKCREATIONSIN
 SelectIDSalarié est entier = FEN_Accueil.TABLE_REQ_LISTSALARIE.COL_IDSalarie
 Selectdate est une chaîne = COMBO_Annee..ValeurAffichée+ COMBO_MOIS..ValeurAffichée + "01"
 sSelectJanvier est une chaîne = COMBO_Annee..ValeurAffichée+ "0101"
 gdSelectjanvier=sSelectJanvier
 grContratheure=FEN_Accueil.TABLE_REQ_LISTSALARIE.COL_HeureSalarie
 MoisPrecedent est un entier = Val(COMBO_MOIS..ValeurAffichée)
 MoisPrecedent = MoisPrecedent - 1
□ SI MoisPrecedent = 0 ALORS
     MoisPrecedent = 12
FIN
 ResultatMoisPrecedent est une chaîne = NumériqueVersChaîne(MoisPrecedent, "02d")
 sSelectDateAvant est une chaîne = COMBO_Annee..ValeurAffichée+ResultatMoisPrecedent+"01"
 nAnEnCours est un entier = COMBO Annee..ValeurAffichée
 gsAnbulencours=nAnEnCours
```

Texte en vert : Variable local (seulement sur ce code) Texte en bleu : Variable global (Touche l'entièreté de la fenêtre.

2023/2024

```
SI PAS HExécuteRequête(REQ_SELECT_DATEBULLETIN, hRequêteDéfaut, SelectIDSalarié, Selectdate) ALORS
      Erreur("Erreur d'initialisation de la requête"+RC+HErreurInfo())
曱
      RETOUR
 SINON
      HLitPremier(REQ_SELECT_DATEBULLETIN)
Ę
      SI REQ_SELECT_DATEBULLETIN.DateBulletin > 0 ALORS
₽
曱
          Info("Une fiche avec cette date existe déjà pour ce salarié : Création impossible !")
          RETOUR
 Ĭ
      SINON
Ę
Ð
          SI PAS HExécuteRequête(REQ_SELECT_DATEDENTREE, hRequêteDéfaut, SelectIDSalarié) ALORS
曱
              Erreur("Erreur d'initialisation de la requête"+RC+HErreurInfo())
              RETOUR
 Ī
          SINON
Ę
              HLitPremier(REQ_SELECT_DATEDENTREE)
曱
              SI REQ_SELECT_DATEDENTREE.DateEntree > Selectdate ALORS
                  Info("Ce Salarié ne peut pas insérer de Bulletin avant sa date d'entrée : Création impossible ! "+SelectIDSalarié)
曱
                  RETOUR
              SINON
Ξ
                  SI REQ_SELECT_DATEDENTREE.DateEntree <= Selectdate OU Selectdate = sSelectJanvier ALORS</pre>
曱
                      grSalairebase=COMBO_Salaire_de_Base..ValeurAffichée
                      grTauxhoraire=SAI Tauxhoraire
                      grHnuit=SAI_HNUIT
                      grMontantFRais=SAI_FRAIS
                      FEN_Accueil.SAI_FRAIS=grMontantFRais
                      grEmbauche150M=SAI_150M
                      FEN_Accueil.SAI_150M=grEmbauche150M
                      gdAtebulletin=COMBO Annee..ValeurAffichée+COMBO MOIS..ValeurAffichée+"01"
曱
                      SI COMBO_CAISSECP..ValeurAffichée="OUI" ALORS
٦
ا
                          gbCaissecp=Vrai
                      SINON
ļ
                          gbCaissecp=Faux
                      FIN
```

Par la suite, je devais également créer une condition pour la suppression du bulletin.

Les conditions sont fonctionnelles.

Fiona	a FAUVELLE 2023/20)24
	Clic sur BTN_SUPPRIMESIMULATION Si Erreur : reprendre la saisie Quand Excep	
	JR TOUTE LIGNE SELECTIONNEE DE TABLE_CumulSimulationMois	
	REQ_DELETE_SIMULATION.iDSimulation = TABLE_CumulSimulationMois. COL_IDSimulation REQ_DELETE_CUMULSIMULATIONMOIS.IDSimulation = TABLE_CumulSimulationMois. COL_IDSimulation	
	<pre>REQ_SELECT_SUPRESSIONBULL.idSalarie = FEN_Accueil.TABLE_REQ_LISTSALARIE.COL_IDSalarie REQ_SELECT_SUPRESSIONBULL.DateAprès = TABLE_CumulSimulationMois.COL_DateBulletin</pre>	
	<pre>SI PAS HExécuteRequête(REQ_SELECT_SUPRESSIONBULL) ALORS Erreur("Erreur d'initialisation de la requête"+RC+HErreurInfo()) RETOUR SINON</pre>	
	HLitPremier(REQ_SELECT_SUPRESSIONBULL) SI REQ_SELECT_SUPRESSIONBULL.DateBulletin > 0 ALORS Info("Suppression Impossible lors des bulletins postérieur sont crées.") RETOUR SINON	
₽ ₽ ₽ 	<pre>SI PAS HExécuteRequête(REQ_DELETE_SIMULATION) ALORS Erreur("Erreur d'initialisation de la requête"+RC+HErreurInfo()) RETOUR SINON</pre>	
- 무 	SI PAS HExécuteRequête(REQ_DELETE_CUMULSIMULATIONMOIS) ALORS Erreur("Erreur d'initialisation de la requête"+RC+HErreurInfo()) RETOUR SINON	
	Ouvre(FEN_PopupConfirmer_Simulation) ExécuteTraitement(BTN_Valider,trtCLic) FIN FIN FIN FIN	

3. Nouvelles fonctionnalités

Après avoir ajouter des petites fonctionnalités sur la fenêtre accueil, on m'a confié de créer une fenêtre de connexion. Au départ on accédait directement à l'application et toutes ses données, alors on m'a donné de créer seule les fonctionnalités qui permettra d'accéder à l'application.

	Login	J'ai pu créer une table se nommant « login » où sont répertorier les informations des utilisateurs et des administrateurs de l'application.				
	IDLogin Abc Prénom Abc Nom IDatedébut Datedébut	CONNECTEZ-VOUS				
l,n (Abc login Abc Mdp 123 LogAdmin	Nom d'utilisateur Mot de passe : • •				
	123 Comptelnactif	Valider Annuler				

Fiona FAUVELLE Code SQL de REQ_SELECT_AUTHENTIFICATION ELLECT Login.login AS IDLogin, Login.logAdmin AS Login, Login.logAdmin AS Admin, Login.logAdmin AS Admin, Login.CompteInactif AS Inactif FROM Login WHERE Login.login = {Paramlogin} AND Login.Mdp = {ParamMotdePasse}

Il y a certaines conditions qui ont été rajouté **après** la création de la fenêtre et de la table à la demande de mes supérieurs, comme : créer la possibilité de mettre un compte actif (= 0) ou inactif (= 1) avec une date de fin. Donc **si la date de fin est supérieure à la date actuelle, l'accès à l'application lui ait refusé.**

Prendre la rebrique "Date Fin" dans la table pour effectuer cette mission. - Ajouter le champ "Inactif" dans la table

Mettre à jour les requêtes qui affiche la date des logins (Ajouter date début et date de fin)
 Dans la création d'un utilisateur, ajouter la possibilité de mettre une date de fin / PAREIL POUR LA MODIFICATION

Dès que celà fonctionne, revoir la fenêtre LOGIN - Eviter connexion.

SI DATEFIN > DATESYS ALORS REFUSER LA CONNEXION FIN 2023/2024

Fior	The FAUVELLE 2023/2024 Clic sur BIN_ValiderConnexion Si Erreur : par programme Quand Exception : par programme
dD)ateAujourdhui est une Date = DateSvs()
hH	leureAujourdhui est une Heure = HeureSvs()
RF	O SELECT AUTHENTIETCATION Paramlogin = SAT NOMULTI ISATEUR
RF	O SELECT_AUTHENTETCATION_ParamMotdePasse = SAI_MOTDEPASSE
₽SI	SAI NOMUTILISATEUR = "" ALORS
ġ	Erreur("Vous devez saisir un nom d'utilisateur"⊠)
L	RepriseSaisie(SAI_NOMUTILISATEUR)
51	NON
Ģ SI	SAI_MOTDEPASSE = "" ALORS
Ģ	Erreur("Le mot de passe est incorrect")
1	RepriseSaisie(SAI_MOTDEPASSE)
L	
51	NON
Ð	SI PAS HExécuteRequête(REQ_SELECT_AUTHENTIFICATION, hRequêteDéfaut, SAI_NOMUTILISATEUR, SAI_MOTDEPASSE) ALORS
ę	Erreur("Erreur d'initialisation de la requête"+RC+HErreurInfo())
Ļ	RETOUR
	SINON
Ð	HLitPremier(REQ_SELECT_AUTHENTIFICATION)
Ę	SI PAS HTrouve ALORS
Ę	<pre>Info("Utilisateur ou mot de passe incorrect.")</pre>
L	RETOUR
51	NON
Ę	SI REQ_SELECT_AUTHENTIFICATION.Expiration >= dDateAujourdhui ALORS
P	<pre>Info("Compte devenu Inactif, veuillez contacter le support.")</pre>
L	RETOUR
	SINON
Ę	SI REQ_SELECT_AUTHENTIFICATION.Inactif = 1 ALORS
曱	<pre>Info("Compte devenu Inactif, veuillez contacter le support.")</pre>
Ļ	RETOUR
	SINON
Ę	grIdlogin=REQ_SELECT_AUTHENTIFICATION.Idlogin
Ð	SI REQ_SELECT_AUTHENTIFICATION.Admin = 1 ALORS
曱	
	REQ_INSERT_DateDuJour.IDLogin = REQ_SELECT_AUTHENTIFICATION.Idlogin
	REQ_INSERT_DateDuJour.DateDeConnexion = dDateAujourdhui
	<pre>REQ_INSERT_DateDuJour.HeureDeConnexion = hHeureAujourdhui</pre>
Ð	SI PAS HExécuteRequête(REQ_INSERT_DateDuJour, hRequêteDéfaut) ALORS
曱	Erreur("Erreur d'initialisation de la requête"+RC+HErreurInfo())
Ļ	RETOUR
L	FIN

Par la suite, dès que l'utilisateur se connectait il arrivait sur la fenêtre accueil. Par la suite j'ai dû créer la fenêtre **administrateur,** en faisant en sorte que si l'utilisateur est admin (=1) **alors** il devait se connecter sur cette fenêtre :

iona FAUVEL	LE	2023/2024
I grIdlogin=	REO SELECT AUTHENTIFICATION. Idlogin	
두 SI REQ_SEL	ECT_AUTHENTIFICATION.Admin = 1 ALORS	
早 	REQ_INSERT_DateDulour_IDLogin = REQ_SELECT_AUTHENTIFICATION.Idlogin	
	REQ_INSERT_DateDulour_DateDeConnexion = dDateAujourdhui	
	REQ_INSERT_DateDuJour.HeureDeConnexion = hHeureAujourdhui	
」 早 日	<pre>SI PAS HExécuteRequête(REQ_INSERT_DateDuJour,hRequêteDéfaut) ALORS Erreur("Erreur d'initialisation de la requête"+RC+HErreurInfo()) RETOUR</pre>	
Ĺ	FIN Ouvre(FEN_BackOffice, 1)	
SINON		
₽//	REQ_INSERT_DateDuJour.DateDeConnexion = DateSys()	
11	REQ_INSERT_DateDuJour.HeureDeConnexion = HeureSys()	
	REQ_INSERT_DateDuJour.IDLogin = REQ_SELECT_AUTHENTIFICATION.Idlogin	
	REQ_INSERT_DateDuJour.DateDeConnexion = dDateAujourdhui	
·····	REQ_INSERT_DateDuJour.HeureDeConnexion = hHeureAujourdhui	
「 早	SI PAS HExécuteRequête(REQ_INSERT_DateDuJour, hRequêteDéfaut) ALORS	
P.	<pre>Erreur("Erreur d'initialisation de la requête"+RC+HErreurInfo())</pre>	
L	RETOUR	
00	Vre(FEN_ACCUELL,5)	

La fenêtre nommé « **Back Office** » a été crée pour log les connexions et les déconnexions des utilisateurs avec les heures.

cueil Gesti	on Administrateur	Gestio	ns des cotisations	Para	mètres			
								Actualiser
echercher								6)
	Login	Ŧ	DateConnexion	¢	HeureConnexion	¢ Q	HeureDeconnexion	p +
								6)
					2			

Page **14** sur **25**

2023/2024

Il y a bien sûr d'autres fonctionnalités propres à la fenêtre, tel que la gestion des cotisations (créer/modifier/supprimer) et la **journalisation :**

Rechercher									
÷ login	Ŧ	Date	Q	÷ Heure	Q	÷ Actions 🔻	÷ Catégorie 🔻	÷ Détails	Þ
								•••	÷.
								0.0.0	
								0.0.0	
								0.0.0	
								0.0.0	
								0.0.0	
								0.0.0	
								0.0.0	↔
								0.0.0	
								0.0.0	
								0.0.0	
								0 0 0	
						\$		0.0.0	~

La journalisation nous permet de voir les créations/modifications/suppression des Taux et des Cotisations (pour l'instant) et de les récupérer si erreur. Pour cela j'ai créé la suite de la base de données :



Les tables « Recup.. » récupère les données des tables du même nom et les enregistre dans une autre table. Afin de pouvoir les retrouver.

Rechercher					
÷ login 🔻	° Date ,Ω	Heure D	¢ Actions ▼	÷ Catégorie 🕎	Détails
tete	01/12/2023	16:09:00	Suppression	Taux	•••
tete	01/12/2023	16:11:00	Suppression	Taux	•••
tete	01/12/2023	16:27:00	Suppression	Taux	•••
tete	06/12/2023	11:12:00	Création	Cotisations	•••
tete	06/12/2023	11:13:00	Création	Cotisations	•••
tete	06/12/2023	11:17:00	Suppression	Cotisations	•••
tete	06/12/2023	12:45:00	Modification	Taux	•••
tete	06/12/2023	14:37:00	Modification	Taux	•••
tete	06/12/2023	15:16:00	Modification	Taux	•••
tete	06/12/2023	15:19:00	Modification	Taux	•••
tete	06/12/2023	15:21:00	Modification	Taux	•••
tete	06/12/2023	15:23:00	Modification	Taux	•••
tete	06/12/2023	15:30:00	Modification	Taux	•••
tete	06/12/2023	15:31:00	Modification	Taux	•••
tete	06/12/2023	15:37:00	Modification	Taux	•••
tete	06/12/2023	15:38:00	Suppression	Taux	•••
tete	06/12/2023	16:54:00	Suppression	Cotisations	•••
tete	06/12/2023	16:54:00	Création	Cotisations	•••

Exemple du tableau de journalisation lors des tests de l'application.

2023/2024

Pour se faire, j'ai dû repasser sur les fonctionnalités créer/modifier/supprimer des taux et cotisations pour ajouter ce bout de programmation :

```
REQ_SELECT_IDLogs.Id = grIdlogin
Ę
         SI PAS HExécuteRequête(REQ_SELECT_IDLogs, hRequêteDéfaut) ALORS
Ð
              Erreur("Erreur d'initialisation de la requête"+RC+HErreurInfo())
              RETOUR
          FIN
          REQ_INSERT_Journalisation.IDLogs = REQ_SELECT_IDLogs.LastID
          REQ_INSERT_Journalisation.HeureManip = hHeureAujourdhui
          REQ_INSERT_Journalisation.Actions = "Création"
          REQ_INSERT_Journalisation.Categorie = "Cotisations"
曱
         SI PAS HExécuteRequête(REQ_INSERT_Journalisation) ALORS
曱
              Erreur("Erreur d'initialisation de la requête"+RC+HErreurInfo())
              RETOUR
          SINON
             FIN
          REQ_SELECT_IDJournalisation.id = REQ_SELECT_IDLogs.LastID
曱
          SI PAS HExécuteRequête(REQ_SELECT_IDJournalisation) ALORS
Ę
              Erreur("Erreur d'initialisation de la requête"+RC+HErreurInfo())
              RETOUR
             FIN
          REQ_INSERT_RECUPCOTI.IDJournalisation = REQ_SELECT_IDJournalisation.LastIDJourna
          REQ_INSERT_RECUPCOTI.Ordre = SAI_COTISATION_ORDRE
          REQ_INSERT_RECUPCOTI.Applicable = CHOIX_COT[1]
          REQ_INSERT_RECUPCOTI.CodeElementBulletin = SAI_COTISATION_CODESILAE
          REQ_INSERT_RECUPCOTI.NomElementBulletin = SAI_COTISATION_LIBELLE
          REQ_INSERT_RECUPCOTI.CotisationSalariale = CHOIX_COT[2]
          REQ_INSERT_RECUPCOTI.CotisationPatronale = CHOIX_COT[3]
曱
          SI PAS HExécuteRequête(REQ_INSERT_RECUPCOTI) ALORS
Ę
              Erreur("Erreur d'initialisation de la requête"+RC+HErreurInfo())
L
              RETOUR
     SINON
          FIN
```

L'objectif par la suite était de journaliser également la création/modification/suppression des salariés et clients mais cela était mit de coté pour effectuer d'autres missions plus urgentes pour la préparation de la **deuxième version** de l'application.

III. Second semestre

A. « ETAT » Cumul

1. Première version de l'ETAT CUMUL

Dès le commencement du second semestre, on me donne une nouvelle mission qui dure encore à l'heure actuelle : Création d'ETAT.

Qu'est-ce qu'un ETAT ?

Selon Windev, un état permet d'obtenir une vue personnalisée d'informations : données saisies dans la base de données, données présentes dans un fichier texte, données provenant d'un champ Table, ...

Quels sont les éléments d'un ETAT ?

Un état est constitué de :

- Blocs délimitant les différentes parties de l'état (pour plus de détails, consultez Blocs d'un état).
- Champs dans lesquels les différentes données de l'état sont affichées (pour plus de détails, consultez Champs d'un état).

Un état offre la possibilité de résumer et de synthétiser les données. Il est possible de :

- Regrouper les données,
- Trier les données selon n'importe quel critère,
- Réaliser des calculs, des moyennes ou des statistiques, ou même de dessiner des graphes,
- Afficher les données sur plusieurs colonnes.

Il suffit d'imaginer un état, l'éditeur d'états le réalisera.

Il existait déjà un ETAT existant créé par mon maitre d'apprentissage, après réflexion il était préférable que je recommence à zéro. L'objectif du premier ETAT Cumul était de calculer la somme des données des bulletins sans et avec DFS d'une société.



Voici l'ETAT Cumul, SANS, AVEC et ECART de chaque donnée représentée ici.

Pour afficher des données, l'ETAT a besoin d'imprimer des données venant d'une table créée sur une fenêtre ou même d'une requête SQL... Pour les ETATS il a été décidé de ré-utiliser les tables.

	Les données	; imprimées viennent de : 🏢 un champ table					
Champ : FEN_Accueil.TABLE_CumulSimulationMois							
	L'état imprime :						
	 Toutes les lignes du champ 						
	Seulement les lignes sélectionnées						

De ce fait, cet ETAT affichera les données de la table nommé « **CumulSimulationMois** » de la fenêtre Accueil.

Fiona FAUVELLE 2023/2024 Un tableau est alors créé sur l'ETAT, avec la possibilité de choisir Colonne 1 le nom, la liaison le type et le calcul que nous pouvons faire. F Nom FEN_Accueil.TABLE_CumulSimulationMois.COL_Nom F Texte 4 <Aucun> 4 < mono-numer ou variable Rubrique ou variable reliée : <Aucun> FEN_Accueil.TABLE_CumulSimulationMois COL_IDSimulation COL_NomSalarié 📳 COL_PrenomSalarié COL_DateBulletin COL_SalaireBrut COL_Dfs_SalaireBrut COL_AbattementConstaté COL_SalaireNet COL_dfs_SalaireNet COL_ChargesPatronales COL_ChargesPatronalesDFS COL_CoutTotalEmployeur COL_dfs_CoutTotalEmployeur COL_EcartSalaireBrut COL_EcartSalaireBrutApresAbattement COL_EcartChargesPatronales COL_GainSalaireNetClientDFS COL_GainCoutTotalEmployeurClientDFS COL_IDSALARIE Service Services

Dans cet ETAT, il existe donc des blocs ayant chacun un rôle. On avait quelques possibilités de les modifier, les supprimer également les rendre invisible **si** le résultat = 0.

Page 19 sur 25

na FAUVELLE	αι												:	2023/202
Début de	documen	Posi t	ition — AVANT le bl APRES le bl	loc "Entê loc "Entê	te de page" te de page"			Ajo	outer un pprimer l Editer le l	bloc 🔻 le bloc bloc				
Entête de	page													
Début de	rupture n	°1												
Corps														
Fin de rup	ture n°1													
Pied de pa	age	Posi	ition —— En bas de Collé aux b	page xlocs pré	cédents									
Fin de doo	cument	Post	ition — AVANT le bl APRES le bl Collé au bl A LA PLACE	loc "Pied loc "Pied oc "Pied E du bloc	de page" de page" de page" "Pied de pag	ge"								
Salaire Cha	ANS DFS	Coût	Taux	Salaire	BRUT Après	AVEC	Net à	Coût	Salaire	BRUT après	ECAR	T Net à	Coût	
Brut Patro	nales payer	Global	Abattement	Brut	abattement	Patronales	payer	Global	Brut	abattement	Patronales	payer	Global	Commentaire
•									= 9 999 9]				

écap	itualitif d	les gain <u>s</u> (a	vant - apr	ès DFS)						
dentifi	NomSalarie	PrenomSalarie	DateBulletin	salairebrut	dfs_salairebru	abattemen	Gain Salaire	GainTotalCou		
								I		
999 9			JJ/MM/AAAA	99 999,99	99 999,99	99 999,99	999 999 999			
				= 99 999,99	= 99 999,99	= 99 999,99	= 999 999 999	= 99 999,99		
										[NUMPAGE]/[NBPAGE
				= 99 999,99	= 99 999,99	= 99 999,99	= 999 999 999	= 99 999,99		
Nombre	de lignes : =	999_999								

Page **20** sur **25**

RH Trans	port							Simul	ation de l	'impact de	e la Đéduc 14/05	tion Fofa	itaire Spé U 14/05	cifique 5/1983	
		SANS DES					AVEC DES					FCART			
Nom Prenom	SALAIRE Che Brut patro	arges ronales Net à payer	Coût Global	Taux Abattement	SALAIRE Brut	BRUT Après Abattement	Charges patronales	Net à payer	Coût Global	SALAIRE Brut	SALAIRE BRUT Abattement	Charges	Net à payer	Coût Global	Nombre de mais
x	= 999 999	000 000 = 999 999	= 999 999	- 00 004	- 000 000	= 999 999	- 000 000	= 999 999	= 999 999	= 999 999	= 999 999	= 999 999	= 999 999	= 999 999	- 0
Total	= 999 999 - 7	000 000 = 999 999	- 999 999		- 000 000	- 999 999	- 000 000	= 999 999	- 999 999	- 999 999	- 999 999	- 999 999	- 999 999	- 999 999	
X Total	= 999 999	000 000 = 999 999 000 000 = 999 999	= 999 999 = 999 999	- 02 00%	- 000 000	= 999 999 = 999 999	- 000 000	= 999 999 = 999 999	= 999 999 = 999 999	= 999 999 = 999 999	= 999 999 = 999 999				

2023/2024

Version finale.

Fiona FAUVELLE

Après la création de la version finale, il m'a été demandé de créer le même ETAT de trois manières différentes.

- 1. Récapitulatif de tous les clients ;
- 2. Récapitulatif de tous les salariés d'un client ;
- 3. Récapitulatif d'un salarié.
 - 2 Deuxième version de l'ETAT CUMUL

En avril, on m'a demandé de revenir sur ses ETATS pour recréer une autre version tout en gardant la **version précédente**, pour cette version, il fallait que le tableau soit modifié et donc plus petit.

J'ai dû revoir certains codes de mon ETAT et de refaire tout l'alignement de mes colonnes. Comme cela faisait beaucoup d'impression possible, mon maitre d'apprentissage m'a demandé de créer un bouton avec des sous menu, pouvant choisi quel type de Récapitulatif l'utilisateur souhaite imprimer.



Fiona FAUVELLE B. ETAT Bulletin

À la suite des ETAT Cumul, on m'a demandé de continuer sur cette voie afin de créer cette fois-ci des ETAT « Bulletin de Salaire » de **trois types :** Sans DFS, avec DFS, les ECART. Je devais reproduire un bulletin de salaire similaire au bulletin de salaire de l'entreprise.

Matricule : Emploi : Statut Professionne Groupe : Coefficient ; Entrée : Ancienneté : Convention Collec	el : tive T i	ransports routiers et	activités auxilia iei	res du transpoi	t		Pé	BUL eriode :	LE ⁻ mai 15	ΓΙΝ Ε 183	DE S	ALA	
⊟éments de paie					Base	Та	ux	A dédu	ire	A Payer	Charg	es Patro	nales
alaire de base	aire de base				151,67		11,79	2	19	1 788, 19	151,6	7	
								= 99 99	9,9 =	999 999,		= '	9 999,99
alaire Net à paye	er ava	ant impôts		2					=	99 999,9			
Heure	S	Heures suppl.	Brut	Plafond S.S	S Netimp	oosable	[∞] Ch.pati	ronales	Coût G	obal Tot	tal versé 🛛		
ensuel	0,00	0,00	0,00	0,	00	0,00	<u>a</u>	0,00		0,00	0,00		0,00
nnuel	0,00	0,00	0,00	0, ∞	00 <u>×</u>	0,00	8	0,00		0,00	0,00		0,00
Quis Congés I	0,00 [∞]	Congés N 0,00	⁸ 0,00	∞ 0,	00	0,00	2	0,00					
is													

Exemple du bulletin de salaire SANS DFS.

Matricule	e :						ΒL	JLL	ET	IN	DE	SAL		RE	DFS
Emploi :								Pé	ériode :	mai	1983				
Statut Pr	rofessionnel :														
Groupe :	:														
Coefficie	ent :														
Entrée :															
Anc ienne	eté :														
Conventi	ion Collective : T	ransports routiers et	activités auxilia	ières du tra	ansport										
	EI	éments de paie			Ba	ase	Ta	aux	A déo	luire	A Paye	er 🛛	Charges	Patror	ales
alaire de bas	e					151,67		11,79			1 78	8,19	151,67		
									= 9 99	99 99	= 99 9	99,		= 99	999,99
oattement Co	onstaté								= 9	99,99%					
alaire Net	à payer ava	int impôts									= 99 9	99,			
14 1	Heures	Heures suppl.	Brut	Plafon	d S.S	Net impo	sable	Ch. pat	ronales	Coût	Global	Total ve	ersé		
ensuel		2		0		1									
nuel		8		0		4									
3	Congés N-1	Congés N		-a											
:quis is				2											

Exemple du bulletin de salaire AVEC DFS.

2023/2024

	Pé	ériode : mai	1983	
transport				
Base	Taux	A déduire	A Payer	Charges Patronales
			= 9 999 99	
0,00	0,00	0,00	0,00	0,00 0,00
		= 9 999 99	= 99 999,	= 99 999,99
	0,00	= 99,99 %	0,00	
			= 99 999,	
ond S.S Net impo	Ch. pat	ronales Coû	t Global Tota	Il versé
	transport Base 0,00 0,00 0,00 0 0 0 0 0 0 0 0 0 0 0 0	transport Base Taux Base 0,00 0,00 0,00 0,00 0,00 0,00 0,00 0 0,00 <	transport Base Taux A déduire 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 999999 0.00 0.00 = 9 999 99 0.00 0.00 = 99,99% 0.00 0.00 = 0.00 0.00 0.00 = 0.00 0.00 0.00 = 0.00 0.00 0.00 = 0.00 0.00 0.00 = 0.00 0.00 0.00 = 0.00 0.00 0.00 = 0.00 0.00 0.00 = 0.00 0.00 0.00 = 0.00 0.00 0.00 = 0.00 0.00 0.00 = 0.00 0.00 0.00 = 0.00 0.00 0.00 = 0.00 0.00 0.00 = 0.00 0.00 0.00 = 0.00 0.00 0.00 = 0.00 0.00 0.00 = 0.00 0.00 0.00 = 0.00 0.00 0.00 0.00 </td <td>transport Base Taux A déduire A Payer 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0 0.00 0.00 0.00 0 0.00 0.00 0.00 0 0.00 0.00 99.999 99.999, 0 0.00 = 99.999.99 = 99.999, 0.00 0 0.00 = 99.999, 0.00 = 99.999, 0.00 0 0.00 E. Coût Global Total 0 0 E. E. E. 0 0</td>	transport Base Taux A déduire A Payer 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0 0.00 0.00 0.00 0 0.00 0.00 0.00 0 0.00 0.00 99.999 99.999, 0 0.00 = 99.999.99 = 99.999, 0.00 0 0.00 = 99.999, 0.00 = 99.999, 0.00 0 0.00 E. Coût Global Total 0 0 E. E. E. 0 0

Exemple du bulletin de salaire ECART (Toujours en cours).

Code SQL de REQ_SELECT_LECTURE_ECARTBULLETIN SELECT DFS_BulletinClient.Libelle AS Libelle, (DFS_BulletinClient.Base - BulletinClient.Base) AS EcartBase, (DFS_BulletinClient.TauxSalarial - BulletinClient.TauxSalarial) AS EcartTauxSalarial, (DFS_BulletinClient.A_deduire_salarial - BulletinClient.A_deduire_salarial) AS EcartA_deduire_salarial, (DFS_BulletinClient.A_deduire_salarial - BulletinClient.A_deduire_salarial) AS EcartA_deduire_salarial, (DFS_BulletinClient.TauxPatronal - BulletinClient.A_payer) AS EcartA_payer, (DFS_BulletinClient.TauxPatronal - BulletinClient.TauxPatronal) AS EcartA_deduire_patronal, (DFS_BulletinClient.A_deduire_patronal - BulletinClient.A_deduire_patronal) AS EcartA_deduire_patronal, BulletinClient.Type AS Type, BulletinClient.IDBulletinClient AS IDBulletinClient FROM BulletinClient, DFS_BulletinClient WHERE BulletinClient.IDBulletinClient = DFS_BulletinClient.IDBulletinClient AND (BulletinClient.IDSimulation = {ParamIDSimulation} AND (((DFS_BulletinClient.Base - BulletinClient.Base) <> 0 AND BulletinClient.Type = 'H' / OR BulletinClient.Type = 'M' OR BulletinClient.Type = 'X')) ORDER BY Type ASC, IDBulletinClient ASC

Code SQL de mon Bulletin Ecart

Par la suite j'ai pu relier ses ETAT Bulletin dans des boutons (Modifié depuis avril) afin que l'utilisateur puisse les imprimer.

Dans mon Bulletin, j'ai ajouté les informations personnelles	s du Salarié, par exemple l'ancienneté :
	DUL
Emploi :	
Statut Professionnel :	
Groupe :	
Coefficient :	
Entrée :	
Ancienneté :	
Convention Collective : Transports routiers et activités auxiliaières du transport	
lcul de la différence en années et mois es est un entier est un entier	
<pre>sDifference = nJoursDifference - (nAnnees * 365) // Soustraction des jours lcul du nombre de mois</pre>	s correspondants aux années
= (nJoursDifference / 30) + 1 // Conversion des jours restants en mois	
fichage du résultat nnees = 0 ALORS nciennetéCode = nMois + " mois" TMON	
fichage du résultat nnees = 0 ALORS nciennetéCode = nMois + " mois" INON nnees > 1 ALORS	
fichage du résultat nnees = 0 ALORS nciennetéCode = nMois + " mois" INON nnees > 1 ALORS nciennetéCode = nAnnees + " ans" + " et " + nMois + " mois"	
<pre>fichage du résultat nnees = 0 ALORS nciennetéCode = nMois + " mois" INON nnees > 1 ALORS nciennetéCode = nAnnees + " ans" + " et " + nMois + " mois" nciennetéCode = nAnnees + " an" + " et " + nMois + " mois"</pre>	
<pre>fichage du résultat nnees = 0 ALORS nciennetéCode = nMois + " mois" INON nnees > 1 ALORS nciennetéCode = nAnnees + " ans" + " et " + nMois + " mois" nciennetéCode = nAnnees + " an" + " et " + nMois + " mois"</pre>	

Code source du libellé « AnciennetéCode »

Le code derrière mon sous-menu est assez simple, car j'utilise la fonctionnalité de **Windev.** J'appelle seulement mon ETAT et je lui dis que l'utilisateur aura la possibilité d'imprimer par la suite.



Fiona FA	UVELLE		Général		2023/2024 Edi
Ξ	🖫 Sélecti	on du menu de	_Menu.OPT	Fichier.OPT	_Imprimer.OP
	iAperçu(i iImprimeE	100, ETAT_ECAR	TSIMULATION	N_SALARIE2, Vr _SALARIE2)	rai)

Voici les nombreuses missions dont j'ai faite depuis septembre 2023, le **projet** est toujours en cours à l'heure actuelle. J'ai essentiellement touché du **SQL** durant mes missions, et j'ai pu **approfondir** mes connaissances sur **Windev**, la **version 2** sera possiblement terminé pour **juin 2024.**

Page **25** sur **25**