

Université de Caen Basse-Normandie
Institut Universitaire de Technologie de Caen
Département Informatique

Diplôme Universitaire de Technologie Informatique

Base de données Revivre

Mémoire de conduite de projet



Valentin DURAND
Alexandre FRERET
Pierre FRIBOULET
Jérémie LE BAS

Tuteurs :
Fabienne JORT
Samir LOUDNI

Commanditaires :
Lysiane DREAN
Gilles LE DUC

Rapport Projet Revivre

Base de données Revivre

Remerciements

Merci à Mme Jort, M. Doucet et M. Loudni pour avoir été nos tuteurs durant ce projet et de nous avoir aidé dans la conduite de celui-ci.

Merci à M. Leduc, Mme Dréan ainsi que tout le personnel de l'association Revivre. Leur soutien, leur sympathie et leur compréhension nous ont permis de travailler pendant ces 2 ans dans un cadre agréable favorisant de bonnes relations humaines et une bonne communication.

Merci aux étudiants de 1ère année Corentin Cosseron, Florian Duval, Paul Jourdain et Benjamin Lemaître d'avoir repris ce projet que nous ne voulions pas voir s'arrêter après tant d'efforts. Nous leur souhaitons également une bonne continuation.

1. Présentation du projet	6
1.1. Contexte	6
1.2. Enjeux	6
1.3. Objectifs	6
1.4. Jalons	7
1.5. Evolution du projet	7
2. Analyse	8
2.1. Modèle logique des Données	8
2.1.1. Extrait du MLD actuel	8
2.1.2. Détail de la table Personne	9
2.1.3. Détail du modèle Chantier	9
2.1.4. Détail du modèle Insertion	9
2.1.5. Evolution de la base au cours des 2 ans	10
2.2. Dictionnaire des données	12
2.2.1. Extrait du DDD	12
2.3. Diagramme de cas d'utilisations	13
2.4. Répartition des tâches	14
2.5. Diagramme de Gantt	15
2.6. Planning des tâches	16
3. Réalisation	17
3.1. Langages de programmation	17
3.1.1. Gestion de la base de données	17
3.1.2. Interface	17
3.1.3. Manipulation des données	19
3.2. Schéma Navigationnel d'Interaction	20
3.3. Ancienne interface	23
3.4. Nouvelle interface	24
4. Réponse à la demande	27
4.1. Fonctionnalités	27
4.1.1. Personnes	27
4.1.2. Chantiers	27
5. Conclusion	28

1. Présentation du projet

1.1. Contexte

L'association Revivre est chargée d'accueillir et d'orienter les personnes en difficulté sociale. Elle propose différentes activités pour réinsérer ses adhérents : chantiers en bâtiment, prestations diverses (nettoyage, conditionnement, assemblage, etc). Elle met aussi à leur disposition un système de navette, de location de véhicule ainsi que de restauration.

Pour pouvoir gérer ces activités mises en place sous forme de chantiers, l'association utilise une base de données permettant de suivre leur avancement, que ce soit au niveau dépenses ou temps de travail. La base gère aussi l'allocation d'encadrants à ces chantiers, ainsi que le suivi des ressources qui leur sont allouées.

Cette base de données a été créée sous Access et est disponible via un Intranet qui est géré par une machine sous Windows Server 2008 R2. Cette base de données a été réalisée par un autodidacte (M. Gilles Leduc). L'organisation de la base de données n'est pas optimisée et ne permet pas à un utilisateur non familier avec le système de pouvoir l'utiliser correctement.

1.2. Enjeux

Le principal enjeu de ce projet est d'accroître la facilité d'utilisation de la base en permettant à d'autres membres de l'association de consulter et d'ajouter des informations très simplement.

Une restructuration de la base de données afin d'obtenir un nouveau schéma cohérent et optimisé est aussi nécessaire, l'ancien schéma de la base n'étant pas utilisable en l'état.

Nous devons aussi permettre à différents utilisateurs de se connecter en même temps sur la base de données, il faut donc gérer le multi-sessions. Un autre enjeu est de convertir la base vers une version libre et gratuite.

1.3. Objectifs

Les objectifs fixés pour ce projet sont :

- gérer les chantiers avec la visualisation et l'impression des devis
- gérer les achats ainsi que les fournisseurs

- gérer les temps de travail
- gérer les salariés et membres de l'association
- gérer le suivi des chantiers
- gérer l'export Excel

1.4. Jalons

13 / 01 / 2014 : Choix du projet

06 / 02 / 2014 : Première prise de contact

24 / 02 / 2014 : Réalisation de la structure papier de la base de données

27 / 03 / 2014 : Présentation de la structure de gestion des chantiers

24 / 04 / 2014 : Présentation de la base au complet

12 / 06 / 2014 : Installation de la partie chantier, premier lot

17 / 11 / 2014 : Installation de la partie chantier, deuxième lot

05 / 01 / 2015 : Installation de la partie salariés en insertion, restructuration de la base

23 / 03 / 2015 : Installation de l'export Excel

1.5. Evolution du projet

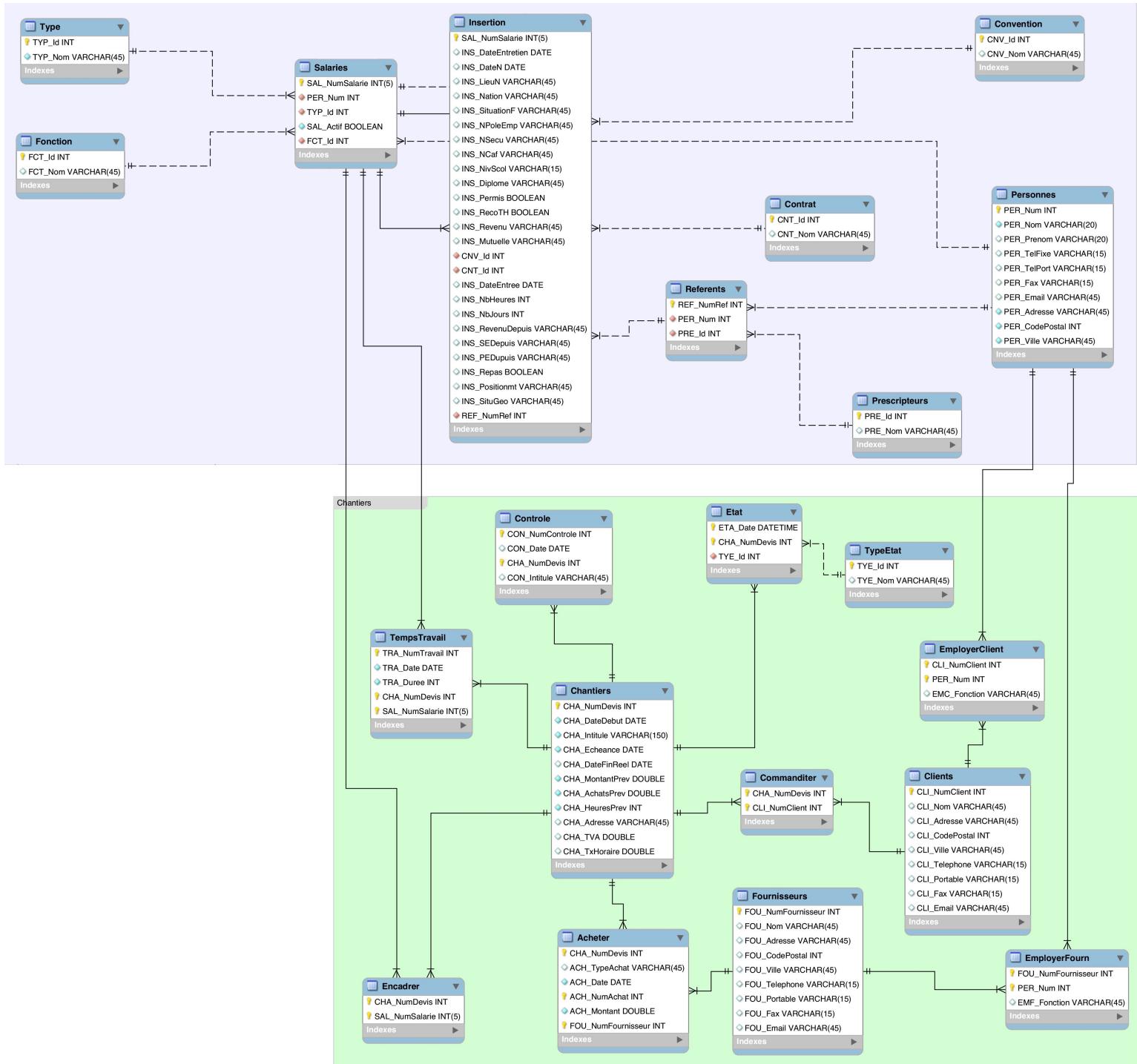
Il est important de noter que les objectifs ainsi que les jalons du projet ont été modifiés à maintes reprises. En effet, les attentes relatives aux fonctionnalités de l'application ont beaucoup évoluées au fur et à mesure de l'avancement du projet. Nous nous sommes donc focalisé avec l'accord du commanditaire sur trois points importants: le répertoire, les chantiers et les salariés en insertion, en laissant de côté les autres activités de l'association. C'est pourquoi ce projet devant être terminé initialement en un cycle normal de 2 ans a été repris par d'autres étudiants et qu'il a été nécessaire de restructurer la base de données sur le jalon du 05/01/2015. Ne sera abordé dans ce rapport uniquement ce que nous avons fait et non ce que nous aurions dû faire si le projet n'avait pas été modifié.

2. Analyse

2.1. Modèle logique des Données

cf. Annexe 1.

2.1.1. Extrait du MLD actuel



2.1.2. Détail de la table Personne

La table personne sert de point central à la base de données. Comme son nom l'indique, toutes les personnes y sont rentrées quelque soit leur statut, on y retrouvera donc les employés, des fournisseurs, les salariés de l'association, les bénévoles, les employés des entreprises clientes ainsi que les particuliers clients.

Les entreprises client et fournisseur ont leur propre table à laquelle sont liés les employés par une contrainte d'intégrité multiple (CIM).

2.1.3. Détail du modèle Chantier

Les clients commanditent un chantier. Le chantier est donc créé avec un état, premièrement « En attente », puis « Validé », « En cours » et « Terminé » ou « Refusé » si le chantier est annulé. Ces états sont stockés dans la table Etat qui est reliée à la table chantier par une simple contrainte d'intégrité fonctionnelle (CIF).

Pendant le déroulement du chantier, les salariés vont travailler sur celui-ci, il faut donc pouvoir rentrer les temps de travail ainsi que les dates pour pouvoir contrôler l'avancement du chantier. Il existe ainsi une CIM entre les chantiers et les salariés représentée par la table TempsTravail. Pour être en mesure de rentrer ces heures de travail, un encadrant doit être attribué au chantier.

Enfin, la gestion des achats doit aussi être prise en compte avec date et montant. Chaque achat doit être relié à un fournisseur préalablement rentré dans la base. La table Acheter sert donc de CIM entre les fournisseurs et les chantiers.

2.1.4. Détail du modèle Insertion

Les salariés ont un type qui les positionne globalement dans la structure de l'association Revivre (stagiaire, salarié, ...) et une fonction qui détaille leurs activités au sein de l'association (secrétaire, encadrant, ...). Si la personne possède le type « Stagiaire », « Salarié en Insertion », ou « Atelier Occupationnel » elle est alors suivie avec une fiche de renseignement détaillée. Cette personne sera aussi encadrée par un référent qui appartient à une structure qui ici est appelée « Prescripteur ».

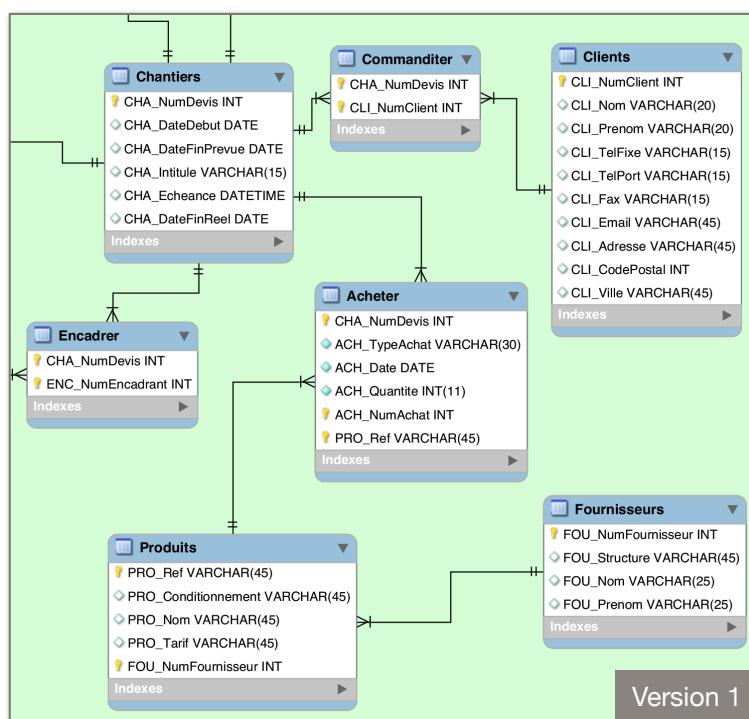
La table Insertion sert donc de complément à la table Salarié et Personne afin de stocker toutes les informations de suivi quand le type du salarié le nécessite. Un référent est lié à cette table par une CIF.

2.1.5. Evolution de la base au cours des 2 ans

En première année nous avions établi une structure de base relative à notre compréhension de l'association et aux attentes du client à ce moment mais celles-ci ayant évolué considérablement durant le déroulement du projet, nous avons été amenés à modifier successivement certains points de l'application.

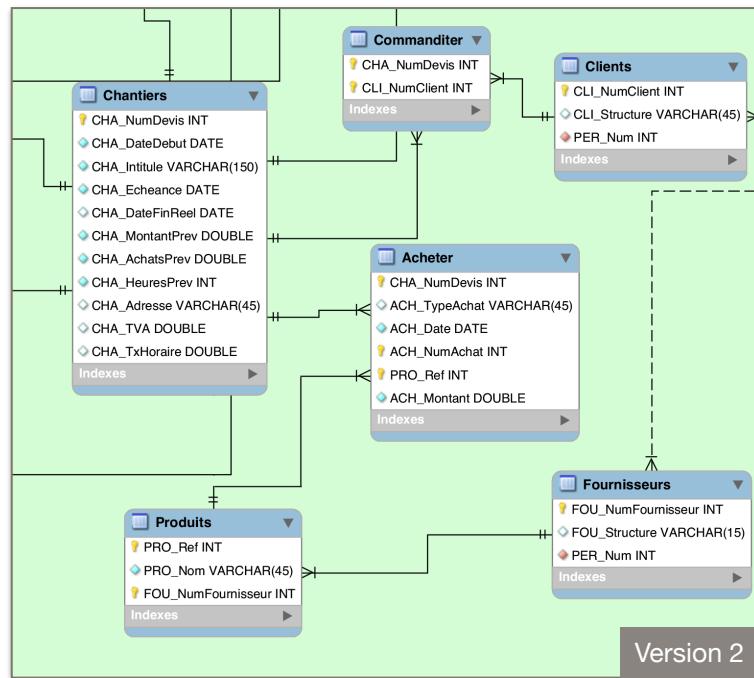
- Gestion des achats dans les chantiers :

Le premier système d'achat rendu à la rentrée de deuxième année (début de S3) consistait en une table Produits comportant le prix unitaire, le conditionnement, le nom, etc. Cette table était liée à un fournisseur par une CIF et l'achat se faisait par une CIM entre la table Chantiers et Produits, la table Acheter représentait cette CIM et comprenait la quantité achetée de tel produit. Le coût total de la commande était ainsi calculé en multipliant la quantité achetée par le prix unitaire du produit. Après utilisation, il s'avérait difficile de positionner certains produits avec des conditionnements différents. Par exemple, des pots de peinture d'1L et 2L nécessitaient deux entrées différentes dans la table Produits.

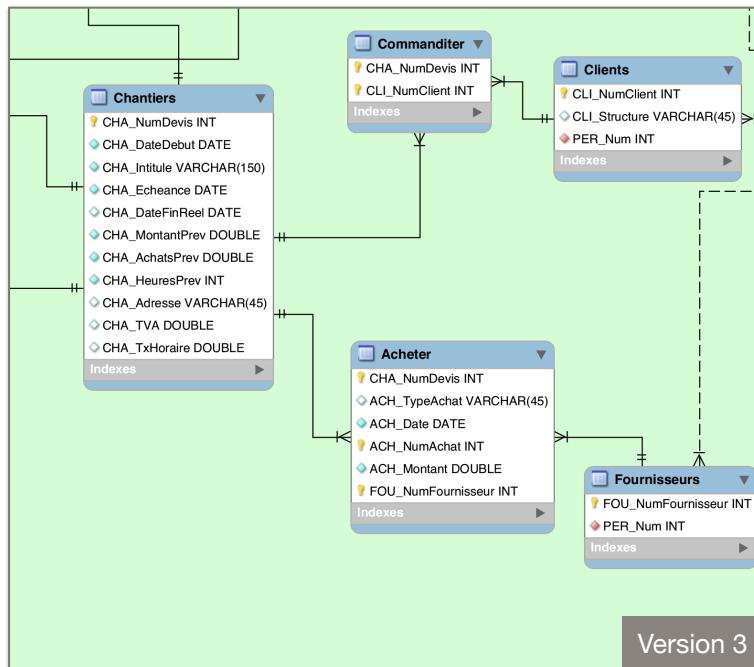


Pour simplifier la gestion des produits, le commanditaire a souhaité pouvoir spécifier directement le prix total sans se soucier de la quantité. La table Acheter incluait donc directement le montant de l'achat et la table Produits listait les produits disponibles pour importer la quantité, identifié par leur référence et nom. Comme dans la version précédente, la clé primaire du fournisseur était incluse dans le produit pour assurer la CIF. Cette méthode imposait encore de nombreuses entrées dans la base et un temps de traitement important pour les utilisateurs, qui devaient sélectionner individuellement chaque produit à acheter et l'ajouter s'il n'existe pas.

Rapport Projet Revivre



Dans la version actuelle de la base, la table Produits a été supprimée pour favoriser une gestion des achats par commande. La table Acheter comporte toutes les informations relatives à une commande chez un fournisseur, peu importe les produits qu'elle contient. Elle est reliée à Fournisseurs par une CIF et l'utilisateur peut attribuer un libellé à cette commande pour l'identifier facilement.



Pour chaque modification de la base, il était nécessaire de mettre à jour l'interface et le traitement pour satisfaire la nouvelle structure.

- Gestion du temps de travail dans les chantiers:

En début de projet, le temps de travail était géré à la minute près, mais le commanditaire nous a fait remarquer que bien que théoriquement optimal, dans la pratique cela s'avérait inapplicable. Nous nous sommes donc orienté vers un modèle par heure complète. Après utilisation les utilisateurs ont préféré revenir vers un système plus précis, par tranche de cinq minutes.

- Organisation des fournisseurs et clients:

Il faut distinguer deux types de clients de l'association, les clients structures (entreprise, association, etc...) et les clients particuliers (individu). La toute première version de la base ne gérait pas cette distinction, nous avons vite corrigé cela en intégrant la table Personnes détaillée précédemment, qui relie aux tables Fournisseurs et Clients permettant de joindre une identité (nom, prénom, adresse, etc...) aux fournisseurs et clients, qu'ils soient structures ou particuliers. Au mois de Décembre, il est remonté lors d'une réunion qu'il serait intéressant de pouvoir associer des employés aux structures. Nous avons donc remodelé massivement la partie Client/Fournisseur de la base, en intégrant une table intermédiaire entre les tables Fournisseurs/Clients et la table Personnes. Cette table sert de CIM afin de lier des employés à chaque structure, et les données des structures ne se trouvent plus dans la table personne mais de nouveau dans les tables Fournisseurs/Clients, comme dans la première version.

2.2. Dictionnaire des données

cf. Annexe 2.

2.2.1. Extrait du DDD

Chantiers

Donnée	Type	Commentaires
CHA_NumDevis	int	Clé primaire
CHA_DateDebut	date	
CHA_Intitule	varchar(200)	
CHA_Echeance	date	
CHA_DateFinReel	date	
CHA_MontantPrev	double	
CHA_AchatsPrev	double	
CHA_HeuresPrev	int	
CHA_Adresse	varchar(150)	
CHA_TVA	double	
CHA_TxHoraire	double	

Salaries

Donnée	Type	Commentaires
SAL_NumSalarie	int(5)	Clé primaire
PER_Num	int(11)	Jointure personne
TYP_Id	int(11)	Jointure type de salarié
SAL_Actif	tinyint(1)	0 : inactif 1 : actif
FCT_Id	int(11)	Jointure fonction

TempsTravail

Donnée	Type	Commentaires
TRA_NumTravail	int(11)	Clé primaire
TRA_Date	date	
TRA_Duree	time	
CHA_NumDevis	int(11)	Jointure chantier
SAL_NumSalarie	int(5)	Jointure salarié

Comme on peut le voir dans cet extrait du dictionnaire des données, l'ajout des temps de travail se fait par la table TempsTravail qui lie les salariés aux chantiers. Les clés primaires de ces deux tables se retrouvent donc dans la table TempsTravail, auxquelles on ajoute la date et la duré du travail. La table salarié est jointe à la table Personnes par la clé primaire de celle-ci (PER_Num) afin de détailler l'identité du salarié. Lui est aussi associé une fonction et un type, tout deux possédant leur table spécifique afin de pouvoir les sélectionner avec une liste déroulant lors de l'ajout.

2.3. Diagramme de cas d'utilisations

Le diagramme de cas d'utilisations représente toutes les actions que pourront faire les différents utilisateurs de l'application que nous avons développé. Cela permet ainsi de définir des schémas d'utilisations et quels sont les différents profils utilisateurs mis en place.

cf. Annexe 3.

2.4. Répartition des tâches

Tâches	Etudiants
Dictionnaire des Données	Le Bas
MLD	Durand, Freret, Friboulet, Le Bas
MCD	Durand, Freret, Friboulet, Le Bas
Cas d'utilisation	Durand, Freret, Friboulet, Le Bas
Diagramme de cas d'utilisation	Le Bas
MySQL	Durand, Friboulet
Vues et Requêtes	Durand, Friboulet
Home	Freret
Bandeau et Pied de page	Freret
CSS	Freret
Menu déroulant	Durand, Freret
JavaScript	Durand
Liste Personnes	Freret, Le Bas
Liste Chantiers	Freret, Le Bas
Tri Personnes	Freret, Le Bas
Détail personne	Freret, Le Bas
Détail chantier	Freret, Le Bas
Ajout personne	Durand, Friboulet
Ajout travail	Durand, Friboulet
Ajout responsable	Durand, Friboulet
Ajout achat	Durand, Le Bas
Ajout produit	Durand, Le Bas
Modification personne	Freret, Friboulet
Modification chantier	Freret, Friboulet
Modification état	Freret, Friboulet
Graphique	Durand
Barres de progression	Durand
Rapport d'anomalies ou erreurs	Durand

Tâches	Etudiants
Maintenance	Durand, Friboulet
Importation des Données	Durand, Friboulet
Modification achat	Le Bas
Adaptation à la nouvelle base (Jalon du 05/01/2015)	Durand
Suivi des chantiers par encadrant	Durand
Impression des données chantier	Freret
Ajout salarié en insertion	Friboulet
Modification salarié en insertion	Friboulet
Modification temps de travail	Friboulet
Exportation Excel	Freret
Ajout référent	Le Bas
Modification référent	Le Bas

2.5. Diagramme de Gantt

Cf. Annexe 4.

2.6. Planning des tâches

N°	RÉCIT UTILISATEUR (OU USER STORY)	PRIORITE	ITERATION
1	L'utilisateur se connecte sur l'adresse internet	1	1
2	Au lancement l'utilisateur peut entrer ses identifiants	2	1
3	L'utilisateur peut visualiser le répertoire (contacts)	3	1
4	L'utilisateur peut ajouter un contact dans le repertoire	4	1
5	L'utilisateur peut modifier un contact	5	1
6	L'utilisateur peut visualiser les chantiers existants	6	2
7	L'utilisateur peut créer un nouveau chantier	7	2
8	L'utilisateur peut attribuer des salariés au chantier	8	2
9	L'utilisateur peut visualiser et ajouter les temps de travail	9	2
10	L'utilisateur peut attribuer des encadrants au chantier	10	2
11	Jauge de progression	11	2
12	L'utilisateur peut modifier l'état du chantier	12	2
13	L'utilisateur peut filtrer les chantiers en fonction du client ou du responsable	13	3
14	L'utilisateur peut ajouter des achats sur un chantier	14	3
15	L'utilisateur peut ajouter des produits	15	3
16	Graphiques des heures de travail et achats	16	3
17	L'utilisateur peut envoyer des rapports d'anomalies ou erreurs	17	3
18	Importation des anciennes données	18	3
19	L'utilisateur peut ajouter des salariés en insertion	19	4
20	L'utilisateur peut visualiser et modifier ces salariés	20	4
21	L'utilisateur peut suivre l'activité d'un encadrant	21	4
22	L'utilisateur peut exporter des données sous Excel	22	4
23	L'utilisateur peut supprimer des temps de travail	23	4
24	L'utilisateur peut supprimer des achats	24	4
25	L'utilisateur peut faire des remises aux fournisseurs	25	4

3. Réalisation

3.1. Langages de programmation

Notre commanditaire souhaitant s'éloigner d'Access, nous nous sommes orientés vers une application Web. Notre choix s'étant fait en première année, nos connaissances de Java / Swing et du modèle objet ne nous semblaient pas assez poussées pour satisfaire les demandes du commanditaire. Les membres de l'association n'accédant à la base que depuis les postes présents sur le site de Colombelles, une application mobile paraissait inutile. L'application web semblait une parfaite solution pour mettre en place rapidement une interface claire et simple afin de pouvoir se focaliser sur les fonctionnalités demandées.

3.1.1. Gestion de la base de données

L'ampleur des données de l'association n'étant pas assez importante pour faire le choix d'une base de données NoSQL tel que MongoDB, nous nous sommes naturellement tournés vers le modèle relationnel. En effet, nous avons étudié cela de façon très approfondie en cours.

Concernant les choix du gestionnaire de base de données, plusieurs alternatives s'offraient à nous. Après avoir étudié différents SGBD, nous avons finalement choisi MySQL pour notre base.

Parmi les autres SGBD que nous aurions pu choisir, on peut citer par exemple : PostgreSQL, Oracle et Access.

Oracle et Access ont la particularité d'être payants or notre commanditaire nous a spécifié qu'il voulait une application libre et gratuite pour ne pas avoir à payer de licences, ce qui serait une grosse contrainte pour l'association qu'il représente.

MySQL a été préféré à Postgre pour sa meilleure intégration Web. De plus MySQL nous a été présenté en cours dès la première année, et s'est avéré très utilisé en deuxième année. Nous avons également trouvé son panneau d'administration (PHPMyAdmin) très simple d'utilisation, ce qui permet à une personne non professionnelle de pouvoir interagir avec la base de données de manière relativement efficace.

3.1.2. Interface

Pour la conception de l'interface, puisque nous avons fait le choix d'une application web, l'HTML / CSS se sont imposés à nous. Ces choix sont presque obligatoires car ce sont les langages les plus répandus, les plus utilisés et les plus efficaces dans la conception de pages Web. L'HTML permet de définir la structure de la page tandis que le CSS permet de créer une mise en page agréable avec un système de balises en cascade ; chaque élément

d'une même catégorie se voit attribuer la même mise en page sans avoir besoin de les éditer individuellement et ceci même après une modification du code HTML.

La design de l'application est basé sur le thème Wordpress utilisé par le site officiel de l'association (<http://revivre-asso.org/>), réalisé par un étudiant d'Info-Com. Le projet ne consistant pas à faire du graphisme, nous nous sommes appuyés sur ce CSS déjà prêt afin de pouvoir nous concentrer rapidement sur l'exécution des jalons.

Nous avons aussi décidé de ne pas utiliser de CMS ni de Framework puisque débutant en PHP lors du commencement du projet. En effet, nous aurions passé davantage de temps à comprendre le fonctionnement du Framework qu'à mettre en place les fonctionnalités requises dans le projet. En codant tout nous-même, nous avons pu grandement nous améliorer en PHP, malgré le fait que nous n'ayons pas utilisé le PHP objet, lacune que nous regrettons maintenant.

Nous avons aussi intégré du JavaScript dans notre application pour permettre des interactions dynamiques, telles que la réorganisation en temps réels des tables de données (tri alphabétique, par type de donnée, ...) ou pour modifier l'affichage des formulaires en fonction du type de contact sélectionné.

Nous avons aussi utilisé les bibliothèques morris.js et raphael.js afin de tracer des graphiques de l'évolution des achats et des temps de travail pour chaque chantier, ainsi que jQuery pour afficher des barres de progression de ces évolutions.

Exemple de code JavaScript:

```
// Barre de progression des achats
$(function () {
    // initialisation de la barre au niveau de l'id progressbar2
    var progressbar = $("#progressbar2"),
        progressLabel = $(".progress-label2");
    // récupération php du pourcentage d'avancement des achats par rapport au montant maximum fixé
    progressbar.progressbar({
        value: <?php if(!empty($MontantMax)) echo round($totAchat*100/$MontantMax, 2); else echo "100"; ?>
    });
    // cas particulier où les achats réalisés sont plus élevés que ceux prévus
    <?php if ($totAchat*100/$MontantMax > 100) {?>
        progressLabel.text("<?php echo round($totAchat*100/$MontantMax, 2) ; ?> %");
    // cas où le montant maximum n'est pas fixé (division par zéro)
    <?php } else if (empty($MontantMax)&&!empty($totAchat)) { $surplusM = 1; ?>
        progressLabel.text("EXCES");
    // cas normal
    <?php } else { ?>
        progressLabel.text(progressbar.progressbar("value") + "%");
    <?php } ?>
});
});
```

3.1.3. Manipulation des données

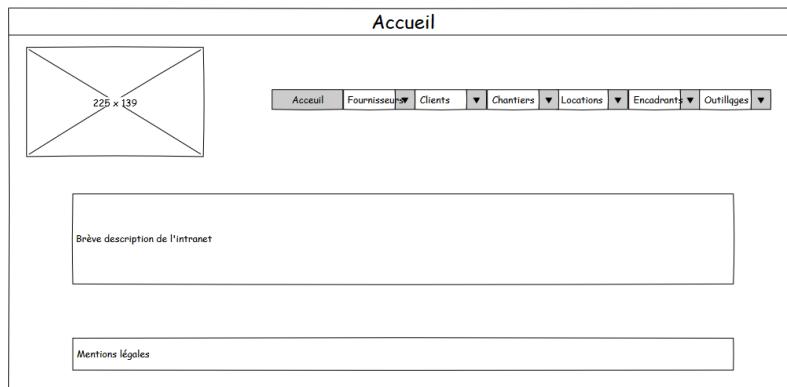
En ce qui concerne le traitement des données (envoi et réception entre la page internet et le serveur), nous avons fait le choix d'utiliser PHP pour traiter ces requêtes, à défaut d'alternative comme ASP ou JSP. En effet, le PHP faisant parti du cursus du DUT, nous avons vu ce projet comme un outil d'apprentissage nous permettant d'assimiler de nombreuses connaissances, et nous aidant par extension dans l'acquisition de notre DUT. Les interactions avec la base ce font grâce aux fonctions mysqli de PHP, sans modèle objet. La plupart du code reste donc très simple mais très redondant. Avec le recule nous aurions souhaité introduire les concepts objets dès le début du projet afin de pouvoir alléger grandement le code et ne pas se retrouver dans notre situation où chaque changement du MLD entraîne une nécessité de mettre à jour d'importante quantité de code.

Exemple de code php:

```

18 $partic=addslashes($_POST["Particulier"]);
19 $fonct=addslashes(mysqli_real_escape_string($db, $_POST["Fonction"]));
20 $newfct = addslashes(mysqli_real_escape_string($db, formatLOW($_POST["NewFct"])));
21 $presc = addslashes(mysqli_real_escape_string($db, $_POST["Prescript"]));
22 $newpresc = addslashes(mysqli_real_escape_string($db, formatLOW($_POST["NewPresc"])));
23
24 $queryPerMax = mysqli_query($db, "SELECT MAX(PER_Num) as maxi FROM Personnes");
25 $resultPerMax = mysqli_fetch_assoc($queryPerMax);
26 $perMax = $resultPerMax['maxi']+1;
27
28 // Ajout d'un Client
29 if($type==0){
30     $queryCliMax = mysqli_query($db, "SELECT MAX(CLINumClient) as maxi FROM Clients");
31     $resultCliMax = mysqli_fetch_assoc($queryCliMax);
32     $cliMax = $resultCliMax['maxi']+1;
33     // Insertion directement dans la table Clients si le client est une structure
34     if($partic == "Structure"){
35         $insertClient = "INSERT INTO Clients VALUES ($cliMax, '$nom', '$add', '$cp', '$ville', '$tel', '$port', '$fax', '$email')";
36         $sql = mysqli_query($db, $insertClient);
37         $errr = mysqli_error($db);
38     }
39     // Si c'est un client on l'insert d'abord dans Personnes puis on le joint en tant que seul employé d'une structure
40     else if($partic == "Particulier"){
41         $insertPersonne = "INSERT INTO Personnes VALUES ($perMax, '$nom', '$prenom', '$tel', '$port', '$fax', '$email', '$add', '$cp', '$ville')";
42         $sql3 = mysqli_query($db, $insertPersonne);
43         $errr3 = mysqli_error($db);
44
45         if($sql3){
46             $insertClient = "INSERT INTO Clients (CLINumClient, CLINom, CLI_Adresse, CLI_CodePostal, CLI_Ville, CLI_Telephone, CLI_Portable, C
47             $sql2 = mysqli_query($db, $insertClient);
48             $errr2 = mysqli_error($db);
49
50             if($sql2){
51                 $insertEmployerClient = "INSERT INTO EmployerClient (CLINumClient, PER_Num, EMC_Fonction) VALUES ($cliMax, $perMax, NULL)";
52                 $sql = mysqli_query($db, $insertEmployerClient);
53                 $errr = mysqli_error($db);
54             }
55         }
56     }
57 }
58 }
```

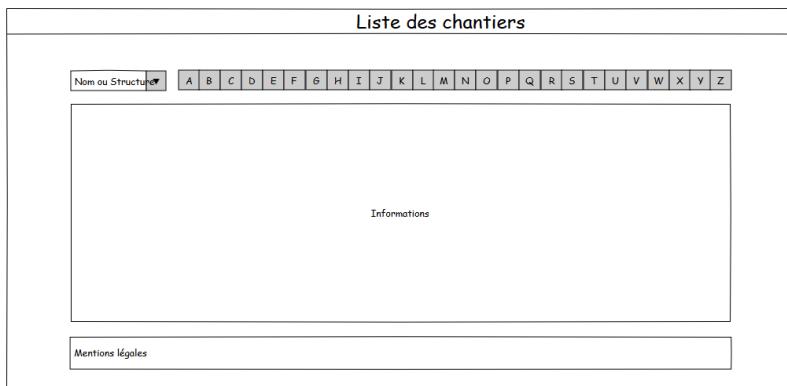
3.2. Schéma Navigationnel d'Interaction



Ce schéma représente la page d'accueil avec :

- le logo de l'association « Revivre » en haut à gauche
- Un résumé central récapitulant les dernières modifications
- Les différents menus déroulants en haut à gauche permettant de naviguer rapidement entre les différentes pages de l'intranet.

Notons que la disposition des menus déroulants et du logo de l'association est commune à toutes les pages de l'intranet.



Ce schéma représente l'affichage des données. Le menu déroulant en haut à gauche permet de trier les données affichées selon des critères précis tandis que les boutons de A à Z permettent de trier alphabétiquement.

Rapport Projet Revivre

Ce schéma sera réutilisé pour toutes données de la base (chantiers, personnes, ...).

Le schéma montre une interface web intitulée "Entrer contact". À la tête de la page, il y a un champ déroulant "Catégorie" avec une option "Catégories" et deux options radio : "Structure" et "Particulier". Ensuite, il y a cinq paires de champs pour les informations de contact : Nom, Prénom, Tel Fixe, Portable et Fax dans la colonne de gauche, et Email, Adresse, Code Postal et Ville dans la colonne de droite. En bas de la page, il y a deux boutons : "Submit" et "Reset", et un encadré pour les "Mentions légales".

Ce schéma représente l'insertion d'un contact dans la base de données. On peut y voir tous les champs à compléter ainsi que le bouton d'envoi du formulaire (« Submit ») et le bouton de remise à zéro du formulaire (« Reset »).

La liste déroulante « Catégorie » permet de choisir parmi tous les types de personne pour sélectionner le type de contact à ajouter. Si le contact est un client, on demande alors si c'est un particulier ou une structure.

Le schéma montre une interface web intitulée "Détail Fournisseur". Il y a deux groupes de champs : un sur la gauche pour Nom, Téléphone Fixe, Téléphone Portable et Fax, et un sur la droite pour Email, Adresse, Ville et Code Postal. En bas de ces groupes, il y a deux boutons : "Modifier" et "Ajouter employé". En dessous, il y a deux tableaux : "Liste des Contacts" (affichant # | Nom | Prénom | Téléphone | Portable | Email) et "Liste des Achats" (affichant Achat | Date | Chantier | Montant). Enfin, il y a un encadré pour les "Mentions légales".

Ce schéma représente le détail d'un fournisseur (ou d'un client) de type catégorisé comme une structure. On peut consulter toutes ses informations. De plus, si des contacts sont rattachés à cette structure, un tableau s'affiche, listant toutes ces personnes. Enfin si on est sur le détail d'un fournisseur, la liste des achats effectués auprès de ce fournisseur s'affiche.

Rapport Projet Revivre

Détail Chantier

Nom du Chantier Adresse Date début Echéance Fin Etat du devis Montant Achats Prévus Heures Prévues Responsable	Client Tél Fixe Adresse Ville TVA Taux Horaire Achat Heures en cours Heures restantes Achats Restants
<input type="button" value="Modifier"/> <input type="button" value="Travail"/> <input type="button" value="Etat"/> <input type="button" value="Achat"/>	
Liste des heures de travail Courbe de progression des heures	
Liste des Achats Courbe de progression des Achats	
<input type="button" value="Mentions légales"/>	

Ce schéma représente la page de suivi d'un chantier où toutes ses informations sont consultables. A remarquer que les données « Achat » et « Heures en cours » sont des barres de progression dynamiques.

En dessous de ce tableau, nous trouvons les boutons permettant de modifier, ajouter un temps de travail, modifier l'état du chantier et ajouter un achat.

Enfin, deux tableaux viennent compléter cette page, l'un représentant les heures de travail et l'autre les achats. Chacun de ces tableaux est accompagné de sa courbe de progression dynamique.

Suivi des chantiers par encadrant

Encadrant	Année	Mois	impression
<input type="button" value="Intitulé"/> <input type="button" value="Montant"/> <input type="button" value="Achats Prévus"/> <input type="button" value="Achats Totaux"/> <input type="button" value="Ecart Achat"/> <input type="button" value="Heures prévues"/> <input type="button" value="Heures Totales"/> <input type="button" value="Ecart MO"/> <input type="button" value="NB Salariés"/>			
Total			
<input type="button" value="Mentions légales"/>			

Ce schéma représente le suivi des chantiers par encadrant. Il permet d'obtenir rapidement un résumé de tous les chantiers effectués dans une période donnée ainsi que par encadrants.

3.3. Ancienne interface

Page d'ajout d'un contact dans Access

Prénom	<input type="text"/>	Catégorie	<input type="text" value="CLIENT"/>
Nom	<input type="text"/>	Service:	<input type="text"/>
Structure	<input type="text"/>	E-mail	<input type="text"/>
Fonction	<input type="text"/>	Web Page	<input type="text"/>
+ / - Images			
Telephone	TEL Standard TEL Bureau TEL Portable Fax		
Adresse	<input type="text"/> Rue Ville Pays CP Country/Region		
	<input type="text"/>		

Page d'ajout d'un chantier

N° Devis	<input type="text" value="0040"/>	Fin réelle	<input type="text" value="20/03/2012"/>
Affecté a	<input type="text" value="HARTEL Jean-M."/>	NB Salarié	<input type="text" value="6"/>
Client	<input type="text" value="176"/>	Durée	<input type="text" value="5"/>
Chantier	Déménagement du tremplin grentheville vers le templin calvados		
Montant	<input type="text" value="3 005,00 €"/>	échéance	<input type="text"/>
TVA	<input type="text" value="19,600%"/>	État	<input type="text" value="Livré"/>
Achats prévus	<input type="text" value="50,00 €"/>	Achat total	<input type="text" value="25,20 €"/>
Heures prévues	<input type="text" value="196"/>	Ecart achat	<input type="text" value="24,80 €"/>
Priorité	<input type="text" value="(2) Normale"/>	Heure total	<input type="text" value="176,5"/>
État	<input type="text" value="Livré"/>	Ecart MO	<input type="text" value="19,5"/>
Date de début	<input type="text" value="15/03/2012"/>		
Fin prévue	<input type="text" value="20/03/2012"/>		

3.4. Nouvelle interface

- La page d'accueil de l'application, offrant une vue rapide des derniers ajout dans la base. Un bandeau au sommet permet à l'aide d'un menu déroulant de naviguer entre les différents menus.

The screenshot shows the Revivre application's main interface. At the top, there is a blue header bar with the logo "REVIVRE" and the tagline "Vivons solidaires !". Below the header, a navigation menu includes "Accueil", "Répertoire", "Chantiers", "Véhicules", "N@vette", "Logements", "Outils", "Plannings", and a search bar. A small "X" icon is in the top right corner of the header. The main content area has a light blue header titled "Base de Données de l'Association Revivre". Below this, there are three sections of tables:

- Derniers Chantiers Ajoutés :** A table with columns: #, Chantier, Client, Responsable, Début, Fin. It lists three entries: 10 (Electricité), 9 (Manutention), and 8 (Espaces Verts).
- Derniers Membres de l'association Ajoutés :** A table with columns: Nom, Prénom, Tél Fixe, Tél Portable, Email, Adresse, Type. It lists three entries: DUPONT (Jean), HANSON (Melinda), and MARKS (Jade).
- Derniers Clients Ajoutés :** A table with columns: Structure, Nom, Prénom, Tél Fixe, Tél Portable, Email, Adresse. It lists three entries: JACOBS, ODIO INC, and ADIPISCING ELIT.

- Le répertoire liste les contacts de l'association ainsi que leurs membres. Des pages de sous-répertoires spécifiques aux Clients, Fournisseurs et Membres ont été créées pour simplifier la navigation. Ces tableaux peuvent être filtrés alphabétiquement ou triés suivant les différentes colonnes.

The screenshot shows a "Liste des Contacts" page. At the top, there is a search bar with fields for "nom, prenom, structure" and a "Rechercher" button. Below the search bar, there is a navigation bar with letters A through Z and a dropdown menu for "Nom". The main content area is a table with the following columns: Structure, Nom, Prénom, Ville, Téléphone, and Type. The table lists various clients, such as ADIPISCING ELIT ETIAM COMPANY, Rowland, Forrest, etc., with their respective details. The "Type" column indicates that all listed entities are "Client".

Rapport Projet Revivre

- **La page d'ajout d'un salarié** est un formulaire complexe permettant l'ajout et le suivi détaillé de ce dernier.

Ajouter un Salarié en insertion

Catégorie	Stagiaire
Date de l'entretien :	<input type="text" value="dd/mm/yyyy"/>
Nom :	<input type="text"/>
Date de naissance :	<input type="text" value="dd/mm/yyyy"/>
Nationalité :	<input type="text"/>
Adresse :	<input type="text"/>
Ville :	<input type="text"/>
Téléphone Fixe :	<input type="text"/>
Téléphone Portable :	<input type="text"/>
Fax :	<input type="text"/>
Code Postal :	<input type="text"/>
Email :	<input type="text"/>
N° Pôle Emploi :	<input type="text"/>
N° Sécurité sociale :	<input type="text"/>
N° CAF :	<input type="text"/>
Référent identifié :	<input type="text" value="Maison Relais"/>
Date d'entrée :	<input type="text" value="dd/mm/yyyy"/>
Convention :	<input type="text" value="CG"/>
Type de contrat :	<input type="text" value="AVA"/>
Jours travaillés :	<input type="text" value="0"/>
Nombre d'heures :	<input type="text" value="0"/>
Niveau scolaire :	<input type="text" value="Non scolarisé"/>
Diplôme :	<input type="text" value="Sans"/>
Reconnaissance TH:	<input type="radio"/> Oui <input checked="" type="radio"/> Non
Permis :	<input type="radio"/> Oui <input checked="" type="radio"/> Non
Revenus :	<input type="text" value="RSA"/>
Depuis :	<input type="text" value="Moins de 6 mois"/>
Inscription Pôle Emploi depuis :	<input type="text" value="Non inscrit"/>
Sans emploi depuis :	<input type="text" value="Moins de 6 mois"/>
Positionnement sur l'atelier du CAP :	<input type="text" value="GOB (ACI)"/>
Mutuelle :	<input type="text" value="CMU"/>
Situation géographique :	<input type="radio"/> ZUS <input type="radio"/> CUCS <input checked="" type="radio"/> Autre
Repas :	<input type="radio"/> Oui <input type="radio"/> Non <input checked="" type="radio"/> Indécis
<input type="button" value="Ajouter"/> <input type="button" value="Annuler"/>	

- **La page de suivi des encadrants** est constituée d'un tableau dynamique listant les chantiers attribués à un encadrant, ainsi que les totaux en fonction du mois et de l'année.

Encadrant	PATEL Jerome	Année	2015	Mois	Janvier	Imprimer
Intitulé	Montant	Achats Prévus	Achats Totaux	Ecart Achat	Heures prévues	Heures Totales
Aménagement	1437 €	887 €	124.00 €	763.00 €	6	16:00
Espaces Verts	1817 €	691 €	2,154.00 €	-1463.00 €	21	22:00
Propreté	1259 €	624 €	121.00 €	503.00 €	48	08:00
TOTAL	4513 €	2202 €	2399 €	-197 €	75	46
						29
						11

Rapport Projet Revivre

- **Le tableau de bord d'un chantier** réunit l'ensemble des données d'un chantier, les heures de travail restantes y sont calculées ainsi que les achats restants et sont mis en valeur grâce à des barres de progression. Un affichage conditionnel en couleur permet de connaître rapidement l'état du chantier.

Deux tableaux contiennent les différents achats et temps de travail appliqués à ce chantier. Depuis cette page, il est possible de modifier les données du chantier, d'ajouter des achats et temps de travail, ainsi que les éditer ou supprimer. On peut aussi changer l'état du chantier et lui allouer un responsable si cela n'a pas été fait lors de la création du chantier.

Un bouton imprimer permet de générer un pdf imprimable contenant les données relatives au chantier, couramment appelé bordereau de livraison.

A l'origine, en plus des tableaux, des graphiques représentant l'évolution des achats et temps de travail venaient compléter ce tableau de bord. Le commanditaire à souhaité ensuite les retirer pour simplifier la mise en page du tableau de bord.

Detail du Chantier n°9		Imprimer	
Nom du Chantier:	Manutention	Client:	ALIQUAM ASSOCIATES
Adresse:	228-1237 In Ave	Tél Fixe :	0876060334
Date de Début:	28 / 01 / 2015	Adresse :	Ap 460-5676 Integer St
Echéance:	25 / 12 / 2015	Ville :	Guwahati 6341
Fin:	--	TVA :	20 %
Etat du Devis:	En Cours	Taux Horaire :	8 € /h
Montant:	1206 €	Achats :	100%
Achats Prévus:	541 €	Heures en cours :	100%
Heures Prévues:	22h	Heures restantes :	00h 00min
Responsable:	DOTSON Hedda	Achat Restant :	0 €
Modifier Travail Etat Achat			
Liste des Heures de Travail			
Membre	Date	Durée	
MARKS Jade	03 / 04 / 2014	09:00	
RUSH Michelle	25 / 05 / 2014	09:00	
ROBLES Bo	16 / 10 / 2014	01:00	
HAYDEN George	17 / 12 / 2014	01:00	
ELLIOTT Ina	01 / 01 / 2015	02:00	
	Heures Totales :	22:00	
<u>Evolution des heures de travail :</u>			
<p>2015-01-01 Nombre d'heures: 22</p>			

4. Réponse à la demande

4.1. Fonctionnalités

4.1.1. Personnes

Les personnes peuvent être ajoutées et modifiées. Si la personne est une structure il y alors possibilité de rattacher des contacts à cette structure, permettant de connaître rapidement toutes les personnes travaillant dans une certaine structure.

Si la personne est un fournisseur, une liste des achats faits auprès de ce fournisseur est disponible.

Si la personne est un client, une liste des chantiers effectués pour ce client est disponible.

La gestion des salariés en insertion, qui est une gestion poussée, est aussi opérationnelle et permet de faire des fiches de suivi relatives à leur insertion dans le monde professionnel.

Fonctionnalités :

- gestion globale des personnes
- gestion des contacts d'une structure
- gestion des fournisseurs, liste des achats effectués
- gestion des salariés en insertion
- gestion client, historique des chantiers
- gestion des membres de l'association Revivre

4.1.2. Chantiers

L'application gère les informations relatives aux chantiers. Ainsi, grâce à la gestion des achats, on peut savoir quel fut le coût du chantier et si le chantier déficitaire par rapport au budget initialement prévu. Avec la gestion des temps de travail, on peut savoir si l'estimation du temps nécessaire est respectée et en tirer des conclusions pour de futurs chantiers similaires. Enfin, ces informations permettent de fabriquer un bordereau de livraison pour le client.

Fonctionnalités :

- gestion globale du chantier
- gestion des encadrants
- gestion des temps de travail, salariés travaillant sur le chantier

- gestion des achats, fournisseurs du chantier
- impression des devis
- gestion de l'état du chantier (en attente, en cours, terminé, refusé)

5. Conclusion

Dû aux nombreux retours sur utilisation de notre commanditaire, les fonctionnalités de l'application se sont grandement développés depuis le début du projet. Celui-ci s'en est donc trouvé très rallongé et nous avons donc dû recentrer le projet, mais cela nous a permis de réellement comprendre la gestion d'un projet et la relation avec un client. Nous avons ainsi pu d'apprendre à travailler en équipe et à être réactif lors des phases de correction d'anomalies.

Au niveau technique, grâce à ce projet nous savons maintenant créer et maintenir une base de donnée et nous avons aussi eu l'opportunité d'expérimenter avec le JavaScript et d'étendre nos connaissances en PHP.

L'association Revivre gérant aussi des services de location de voiture, de restauration, de location d'outillage, ... Tous ces services peuvent être intégrés à l'application selon le souhait de M. Leduc, il reste donc un travail important à fournir afin de combler toutes les attentes de l'association. La reconduction du projet s'avère alors tout à fait judicieuse, tant elle devrait permettre aux nouveaux étudiants d'approfondir grandement leurs connaissances. De plus, quand on prend en considération les nombreux retours sur l'application, le développement s'en trouve ralenti, mais au profit d'une diversification des savoir-faire emmagasinés.

Glossaire

Access (ou **Microsoft Access**) : SGBD relationnel édité par Microsoft et faisant partie de la suite Microsoft Office.

Application libre (ou **Logiciel libre**) : un logiciel libre est un logiciel dont l'utilisation, l'étude, la modification et la duplication en vue de sa diffusion sont permises, techniquement et légalement.

Application Web : application manipulable grâce à un navigateur Web.

ASP (ou **Active Server Pages**) : ensemble de logiciels développés par Microsoft et utilisés dans la programmation Web. C'est une suite de logiciels destinée à créer des sites web dynamiques.

Barre de progression : composant de base des interfaces graphiques qui permettent d'indiquer à l'utilisateur l'état d'avancement d'un travail qu'est en train d'effectuer l'ordinateur. Au début la barre est complètement vide, puis elle se remplit au fur et à mesure de l'avancement de la tâche pour finir complètement remplie lorsque le travail est terminé.

Base de données : une base de données est un outil permettant de stocker et de retrouver l'intégralité de données brutes ou d'informations en rapport avec un thème ou une activité.

Cahier des charges : document qui doit être respecté lors de la réalisation d'un projet, contenant les objectifs à réaliser clairement établis de ce dernier.

Cas d'utilisation : en génie logiciel et en ingénierie des systèmes, un cas d'utilisation définit une manière d'utiliser le système et permet d'en décrire les exigences fonctionnelles.

Clé primaire : dans une base de données relationnelle, une clé primaire est une contrainte d'unicité qui permet d'identifier de manière unique un enregistrement dans une table.

CSS (ou **Cascading Style Sheets**) : **feuilles de style en cascade** en français, langage informatique qui décrit la présentation des documents HTML et XML.

Contrainte d'Intégrité Fonctionnelle (ou **CIF**) : association entre deux entités en Merise. Les CIF ont pour particularité d'être binaires et d'avoir une cardinalité min à 0 ou 1 et une cardinalité max à 1 ou n, de plus elles ne sont pas porteuses de propriétés.

Contrainte d'Intégrité Multiple (ou **CIM**) : association entre deux entités en Merise (méthode d'analyse, de conception et de gestion de projet informatique). Les CIM sont n-aires et ont toutes leurs cardinalités max à n, de plus elles peuvent être porteuses de propriétés.

Diagramme de cas d'utilisations : diagramme UML utilisé pour donner une vision globale du comportement fonctionnel d'un système logiciel.

Diagramme de Gantt : outil utilisé en ordonnancement et en gestion de projet et permettant de visualiser dans le temps les diverses tâches composant un projet. Il permet de représenter graphiquement l'avancement du projet.

Dictionnaire des données (ou **DDD**) : collection de métadonnées ou de données de référence nécessaire à la conception d'une base de données relationnelle.

Excel (ou **Microsoft Excel**) : logiciel tableur de la suite bureautique Microsoft Office, développée et distribuée par l'éditeur Microsoft.

HTML : l'Hypertext Markup Language, généralement abrégé HTML, est le format de données conçu pour représenter les pages web. C'est un langage de balisage permettant d'écrire de l'hypertexte, d'où son nom. HTML permet également de structurer sémantiquement et de mettre en forme le contenu des pages, d'inclure des ressources multimédias dont des images, des formulaires de saisie, et des programmes informatiques.

IHM (ou **Interactions Homme-Machine**) : les interactions homme-machine (IHM) définissent les moyens et outils mis en œuvre afin qu'un humain puisse contrôler et communiquer avec une machine.

Interaction dynamique : possibilité de changer intuitivement et facilement des variables, par opposition à une interaction statique.

Intranet : réseau informatique interne à une entreprise (ou toute autre entité organisationnelle), mettant à la disposition du personnel les technologies d'échange d'informations issues d'Internet (courrier électronique, visioconférence, pages web, etc.), et avec l'utilisation des mêmes protocoles (TCP, IP, HTTP, SMTP, IMAP, etc...).

Itération : courte période de temps, de durée fixe, durant laquelle vont se dérouler une série d'activités (analyse, conception, codage, test...) et qui se termine par une livraison.

Jalon : dans le cadre de la gestion de projet, un jalon est la fin d'une étape, la fin d'un travail. Il est ainsi intéressant de fixer des échéances dans le respect des méthodes agiles.

Java : langage de programmation orienté objet.

JavaScript : langage de programmation de scripts.

Jointure : en informatique et plus particulièrement dans les bases de données relationnelles, la jointure est l'opération permettant d'associer plusieurs tables ou vues de la base par le biais d'un lien logique de données entre les différentes tables ou vues, le lien étant vérifié par le biais d'un prédictat. Le résultat de l'opération est une nouvelle table.

JSP (ou **JavaServer Pages**) : technique basée sur Java qui permet aux développeurs de créer dynamiquement du code HTML, XML ou tout autre type de page web.

Modèle Conceptuel de Données (ou **MCD**) : le modèle conceptuel des données a pour but d'écrire de façon formelle les données qui seront utilisées par le système d'information. Il s'agit donc d'une représentation des données, facilement compréhensible, permettant de décrire le système d'information à l'aide d'entités.

Modèle Logique des Données (ou **MLD**) : le modèle logique des données consiste à décrire la structure de données utilisée sans faire référence à un langage de programmation. Il s'agit donc de préciser le type de données utilisées lors des traitements. Ainsi, le modèle logique est dépendant du type de base de données utilisé.

Modèle objet ou Programmation orientée objet : type de programmation qui consiste en la mise en relation d'objets, c'est-à-dire d'éléments de programmation, avec un langage spécifique qui permet aux objets de communiquer entre eux.

Modèle relationnel : manière de modéliser les informations contenues dans une base de données. Il permet de représenter la base de données comme un ensemble de tables, sans préjuger de la façon dont les informations sont stockées dans la machine. Les tables constituent donc la structure logique du modèle relationnel.

MongoDB : système de gestion de base de données orientée document, répartissable sur un nombre quelconque d'ordinateurs et ne nécessitant pas de schéma prédéfini des données.

MySQL : système de gestion de bases de données relationnelles (SGBDR).

NoSQL : catégorie de systèmes de gestion de base de données (SGBD) qui n'est plus fondée sur l'architecture classique des bases relationnelles. L'unité logique n'y est plus la table, et les données ne sont en général pas manipulées avec SQL.

Oracle ou (**Oracle Database**) : système de gestion de base de données relationnel (SGBDR) qui depuis l'introduction du support du modèle objet dans sa version 8 peut être aussi qualifié de système de gestion de base de données relationnel-objet (SGBDRO).

PDF (ou Portable Document Format) : langage de description de pages créé par la société Adobe Systems en 1993.

PHP : langage de programmation, qui est interprété par un serveur web.

phpMyAdmin : application Web (permettant notamment l'administration) pour les systèmes de gestion de base de données MySQL réalisée en PHP et distribuée sous licence GNU GPL.

Pied de page : élément écrit ou graphique qui se situe en bas d'un document (nom, prénom, numéro de page, etc.).

Planning des tâches (ou Backlog) : liste de fonctionnalités ou de tâches, jugées nécessaires et suffisantes pour la réalisation d'un projet.

PostgreSQL : système de gestion de base de données relationnelle et objet.

Rapport d'anomalie : rapport par lequel un programme signale une anomalie. Ce rapport explique cette dernière en même temps que l'environnement de son occurrence.

Récit utilisateur (ou User Story) : phrase simple dans le langage courant permettant de décrire avec suffisamment de précision le contenu d'une fonctionnalité à développer.

Requête : en traitement de bases de données, une interrogation de la base. Le client reçoit comme réponse des éléments de cette base correspondant aux critères définis dans la requête. Dans le cas particulier des bases de données relationnelles, le format standard de requête est le SQL.

Schéma Navigationnel d'Interactions (ou SNI) : modélisation du contenu d'une IHM.

SQL (ou Structured Query Language) : langage de requête structurée en français, langage informatique normalisé servant à exploiter des bases de données relationnelles.

Swing : bibliothèque graphique pour le langage de programmation Java, permettant de générer des interfaces.

Système de Gestion de Base de Données (ou **SGBD**) : logiciel système destiné à stocker et à partager des informations dans une base de données, en garantissant la qualité, la pérennité et la confidentialité des informations, tout en cachant la complexité des opérations.

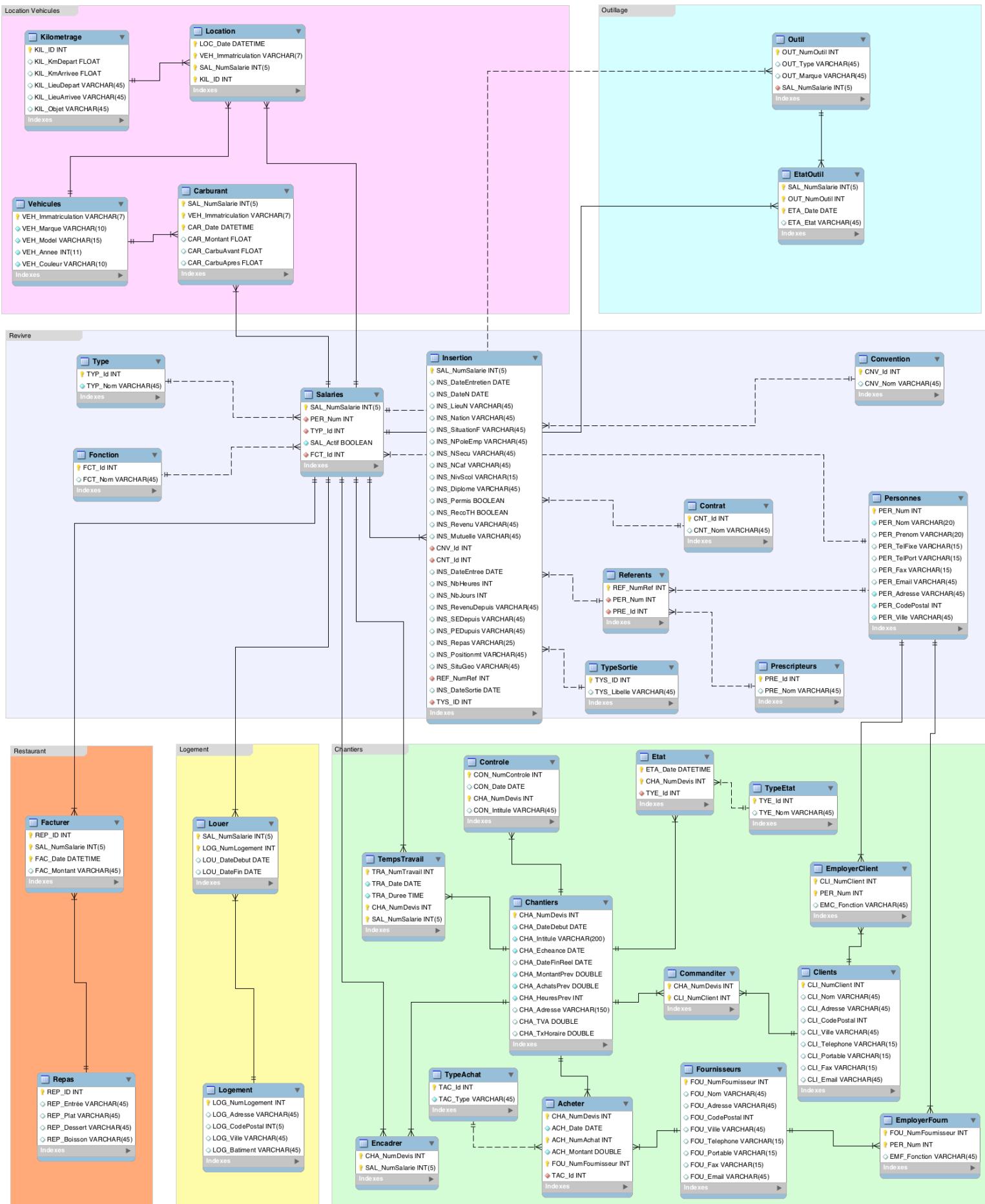
Table : dans les bases de données relationnelles, une table est un ensemble de données organisées sous forme d'un tableau où les colonnes correspondent à des catégories d'information (une colonne peut stocker des numéros de téléphone, une autre des noms...) et les lignes à des enregistrements, également appelés entrées.

UML (ou **Unified Modeling Language**) : **langage de modélisation unifié** en français, langage de modélisation graphique à base de pictogrammes. Il est utilisé en développement logiciel, et en conception orientée objet. UML est couramment utilisé dans les projets logiciels.

Vue : dans une base de données, une vue est une synthèse d'une requête d'interrogation de la base. On peut la voir comme une table virtuelle, définie par une requête.

Windows Server 2008 R2 : il s'agit de la version « serveur » du système d'exploitation Windows 7 (qui constitue quant à lui une version « client »). Ils partagent tous deux un noyau Windows NT 6.1.

Annexe 1: MLD



Annexe 2: Dictionnaire des données

Acheter

Donnée	Type
CHA_NumDevis	int(11)
ACH_Date	date
ACH_NumAchat	int(11)
ACH_Montant	double
FOU_NumFournisseur	int(11)
TAC_Id	int(11)

Chantier

Donnée	Type
CHA_NumDevis	int(11)
CHA_DateDebut	date
CHA_Intitule	varchar(200)
CHA_Echeance	date
CHA_DateFinReel	date
CHA_MontantPrev	double
CHA_AchatsPrev	double
CHA_HeuresPrev	int(11)
CHA_Adresse	varchar(150)
CHA_TVA	double
CHA_TxHoraire	double

Clients

Donnée	Type
CLI_NumClient	int(11)
CLI_Nom	varchar(45)
CLI_Adresse	varchar(45)
CLI_CodePostal	int(11)
CLI_Ville	varchar(45)
CLI_Telephone	varchar(15)
CLI_Portable	varchar(15)
CLI_Fax	varchar(15)
CLI_Email	varchar(45)

EmployerClient

Donnée	Type
CLI_NumClient	int(11)
PER_Num	int(11)
EMC_Fonction	varchar(45)

EmployerFourn

Donnée	Type
FOU_NumFournisseur	int(11)
PER_Num	int(11)
EMF_Fonction	varchar(45)

Etat

Donnée	Type
ETA_Date	datetime
CHA_NumDevis	int(11)
TYE_Id	int(11)

Fournisseurs

Donnée	Type
FOU_NumFournisseur	int(11)
FOU_Nom	varchar(45)
FOU_Adresse	varchar(45)
FOU_CodePostal	int(11)
FOU_Ville	varchar(45)
FOU_Telephone	varchar(15)
FOU_Portable	varchar(15)
FOU_Fax	varchar(15)
FOU_Email	varchar(45)

Table salaries

Donnée	Type
SAL_NumSalarie	int(5)
PER_Num	int(11)
TYP_Id	int(11)
SAL_Actif	tinyint(1)
FCT_Id	int(11)

Insertion

Donnée	Type
SAL_NumSalarie	int(5)
INS_DateEntretien	date
INS_DateN	date
INS_LieuN	varchar(45)
INS_Nation	varchar(45)
INS_SituationF	varchar(45)
INS_NPoleEmp	varchar(45)
INS_NSecu	varchar(45)
INS_NCaf	varchar(45)
INS_NivScol	varchar(15)
INS_Diplome	varchar(45)
INS_Permis	tinyint(1)
INS_RecoTH	tinyint(1)
INS_Revenu	varchar(45)
INS_Mutuelle	varchar(45)
CNV_Id	int(11)
CNT_Id	int(11)
INS_DateEntree	date
INS_NbHeures	int(11)
INS_NbJours	int(11)
INS_RevenuDepuis	varchar(45)
INS_SEDepuis	varchar(45)
INS_PEDupuis	varchar(45)
INS_Repas	varchar(25)
INS_Positionmt	varchar(45)
INS_SituGeo	varchar(45)
REF_NumRef	int(11)
INS_DateSortie	date
TYS_ID	int(11)

Personnes

Donnée	Type
PER_Num	int(11)
PER_Nom	varchar(20)
PER_Prenom	varchar(20)
PER_TelFixe	varchar(15)
PER_TelPort	varchar(15)
PER_Fax	varchar(15)
PER_Email	varchar(45)
PER_Adresse	varchar(45)
PER_CodePostal	int(11)
PER_Ville	varchar(45)

Table TempsTravail

Donnée	Type
TRA_NumTravail	int(11)
TRA_Date	date
TRA_Duree	time
CHA_NumDevis	int(11)
SAL_NumSalarie	int(5)

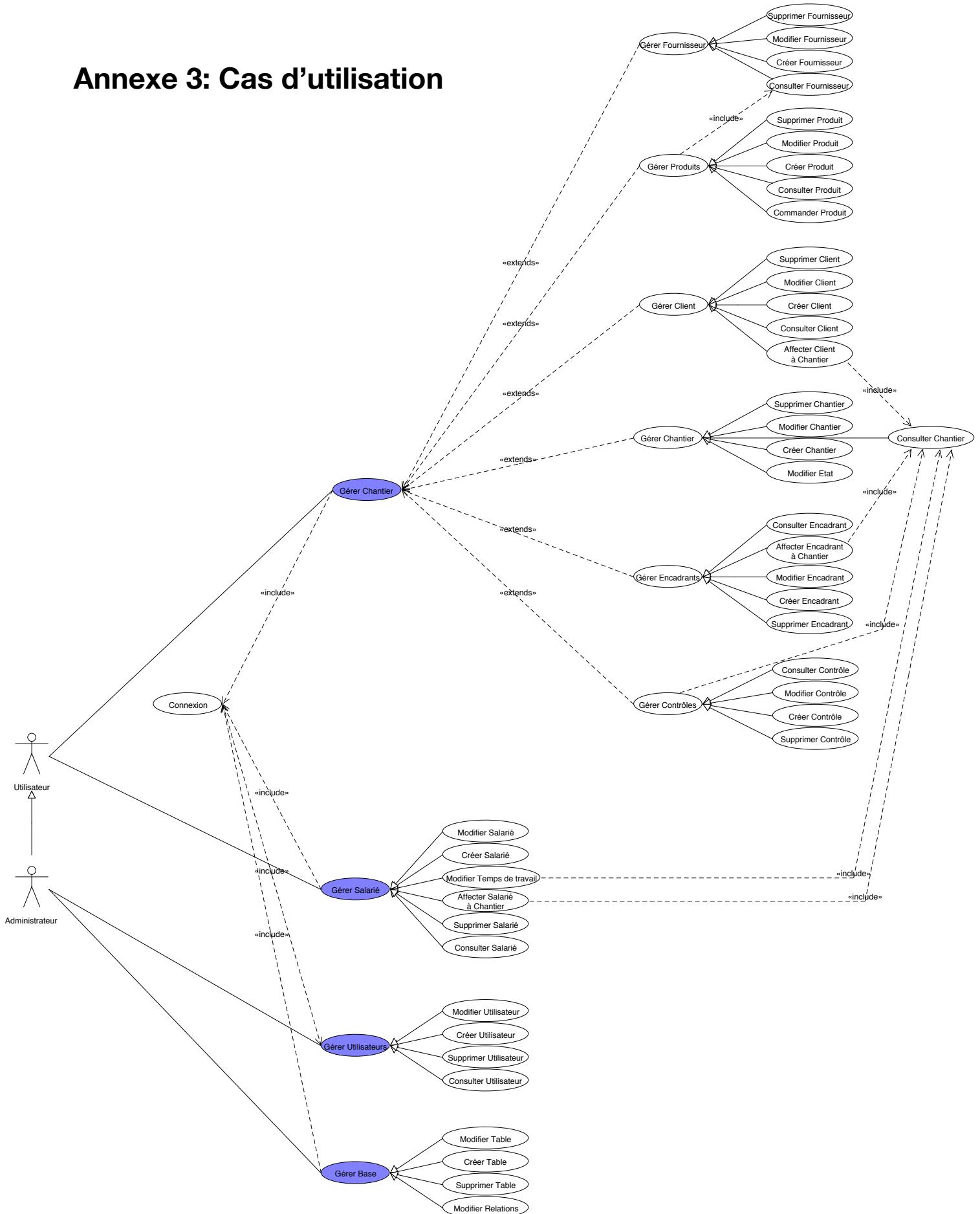
Referents

Donnée	Type
REF_NumRef	int(11)
PER_Num	int(11)
PRE_Id	int(11)

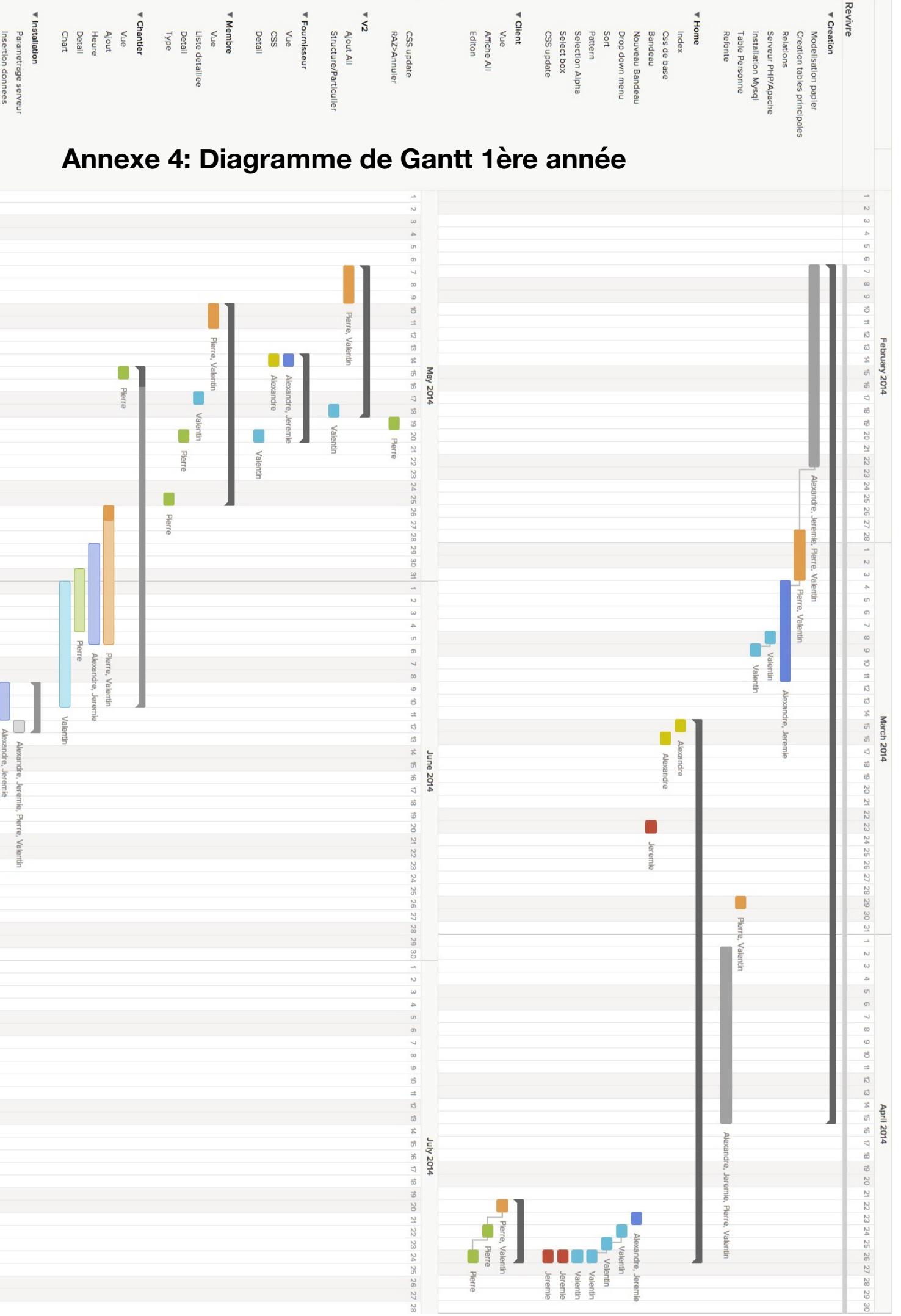
Prescripteurs

Donnée	Type
PRE_Id	int(11)
PRE_Nom	varchar(45)

Annexe 3: Cas d'utilisation



Annexe 4: Diagramme de Gantt 1ère année



Annexe 4:

Diagramme de Gantt

2ème année

