



# Rapport de Stage

---

Développeur Informatique  
Année 2007/2008

Simona DUMITRU

## TABLE DES MATIERES

<b>TABLE DES MATIERES</b>	<b>2</b>
<b>PRESENTATION DE L'ENTREPRISE</b>	<b>3</b>
<b>PROJET MAWARI</b>	<b>5</b>
<b>I THEME DU PROJET ET CAHIER DES CHARGES</b>	<b>6</b>
<b>I.1 CONTEXTE ET OBJECTIFS</b>	<b>6</b>
<b>I.2 DESCRIPTIF DU PROJET</b>	<b>6</b>
<b>I.3 CAHIER DES CHARGES</b>	<b>7</b>
<b>I.4 LA BASE DE DONNEES</b>	<b>9</b>
I.4.1 MODELE CONCEPTUEL DES DONNEES	10
I.4.2 MODELE PHYSIQUE DES DONNEES	11
<b>I.5 CHOIX TECHNIQUE</b>	<b>12</b>
<b>II DEVELOPPEMENT</b>	<b>13</b>
<b>II.1 PRESENTATION DE L'INTERFACE</b>	<b>13</b>
<b>II.2 LE CODE</b>	<b>29</b>
<b>PROJET TABLEAU DE BORD</b>	<b>32</b>
<b>I THEME DU PROJET ET CAHIER DE CHARGE</b>	<b>33</b>
<b>I.1 DESCRIPTIF DU PROJET</b>	<b>33</b>
<b>I.2 CHOIX TECHNIQUE</b>	<b>33</b>
<b>I.3 CAHIER DES CHARGES</b>	<b>34</b>
<b>I.4 L'ARBORESCENCE DU SITE</b>	<b>34</b>
<b>I.5 LA BASE DE DONNEES</b>	<b>35</b>
I.5.1 MODELE CONCEPTUEL DES DONNEES	36
I.5.2 MODELE PHYSIQUE DES DONNEES	37
<b>II DEVELOPPEMENT</b>	<b>38</b>
<b>II.1 PRESENTATION DE L'INTERFACE</b>	<b>38</b>
<b>II.2 SQL</b>	<b>44</b>
<b>CONCLUSION</b>	<b>46</b>
<b>REMERCIEMENTS</b>	<b>46</b>

## PRESENTATION DE L'ENTREPRISE

Indigo Global Services est une SSII en plein développement créée en octobre 2004 autour de 3 pôles d'activités :

- la gestion de ressources humaines
- l'infogérance
- le développement

Les principes capitaux, qui font office de valeurs de la société sont : objectif, méthode, communication, engagement, implication, confiance, innovation.

### LE RECRUTEMENT

- Le recrutement direct : mode de recrutement par lequel Indigo Global Services agit tel un cabinet de recrutement ;
- La pré-embauche : équivaut, de manière approximative, à une période d'essai sécurisée. Durant sa période d'essai, le salarié est employé par Indigo Global Services.
- La régie : le collaborateur est employé par Indigo Global Services. Il est positionné chez des clients finaux pour des missions à contenu et à durée variables.

### L'INFOGERANCE



Intitulé VIPC (Very Important Personal Computer) en interne, l'infogérance proposée des prestations comme :

Installation (pc, logiciel, adsl, wifi, petit réseau), assistance utilisateurs/mise en réseau wifi, câblage, formation (utilisateur), intégration (mise en réseau systèmes/de solution), sécurité (solutions), solutions réseaux (administration/supervision), messagerie, hébergement, administration et supervision des serveurs, revente de matériel, revente de logiciel.

## LE DEVELOPPEMENT



Appelé en interne VIPDev (Very Important Personal Développement), à pour but de créer des applications garantissant accompagnement et expertise tout au long du cycle projet et des différentes étapes qui le constituent :

- Analyse de l'existant
- Etude des besoins et faisabilité
- Développement d'applications
- Intégration
- Formation et transfert de compétences
- Assistance
- Maintenance

# PROJET MAWARI

# I THEME DU PROJET ET CAHIER DES CHARGES

## **I.1 CONTEXTE ET OBJECTIFS**

Dans le cadre de ma formation à l'AFPA, j'ai effectué un stage de trois mois dans le service Développement de l'entreprise Indigo Global Services.

Ma principale mission pendant ce stage a été d'ajouter de fonctionnalités et d'améliorer une plate-forme web d'échange entre différentes sociétés (un groupware).

## **I.2 DESCRIPTIF DU PROJET**

Le CIFEG *Centre International pour la Formation et les Echanges en Géosciences* a pour objet de favoriser les échanges et la coopération dans le domaine des Sciences de la Terre (ressources minérales, ressources en eau, géologie des risques naturels) entre la France, les pays industriels et les pays en développement.

Dans le cadre de sa coopération dans le projet MAWARI (Gestion durable de la ressource en eau dans le système du Rift est africain), le CIFEG a mis en place une solution de type Groupware.

L'application permet le renforcement et l'optimisation de la collaboration entre plusieurs pays : l'Ethiopie, Djibouti, le Kenya, le Sénégal et la France.

### **I.3 CAHIER DES CHARGES**

#### **Panneau d'administration**

Fonctionnement actuel de l'application :

- Quatre type de compte (Admin, Manager, Member, User)
- Seul les « Admin » ont la possibilité de créer/supprimer/modifier les données des comptes de la plateforme.
- Lors de la création d'un groupe d'utilisateur, un répertoire de travail est automatiquement créé dans lequel est créé un répertoire « Budget ».

Fonctionnement souhaité :

- Les « Manager » peuvent créer/supprimer/modifier des comptes pour leur groupes de travail.
- Tous les comptes peuvent modifier leurs données personnelles.
- Crypter les mots de passes en MD5.
- Lors de la création d'un groupe d'utilisateur, permettre le choix d'avoir ou non un répertoire « Budget ».

#### **Calendrier**

Fonctionnement actuel :

- Affichage des tâches dans une couleur unique.
- Affichage du détail d'une tâche comprenant les informations basiques relatives à la tâche.

Fonctionnement souhaité :

- Les tâches doivent avoir une couleur différente qui correspondait au projet auquel elles sont rattachées.
- L'affichage du détail d'une tâche doit renvoyer sur le gestionnaire de projet pour afficher la totalité des informations se rapportant à cette tâche.

## **Gestionnaire de projet**

Fonctionnement actuel :

- Les projets qui ont été définis en tant que projets protégés sont accessibles par tout le monde sauf pour les répertoires de travail. Aucune modification ne peut-être apportée au projet par des personnes extérieures au projet.

Fonctionnement souhaité :

- Les projets protégés ne doivent être visible que par les membres de ce projet.

## **Accueil**

Fonctionnement actuel :

- Page statique expliquant le fonctionnement des divers modules.

Fonctionnement souhaité :

- Possibilité d'éditer la page afin de pouvoir la mettre à jour avec diverses informations.

## **Général**

Ajouter des messages pour avertir lorsqu'une action s'est correctement déroulée ou non.



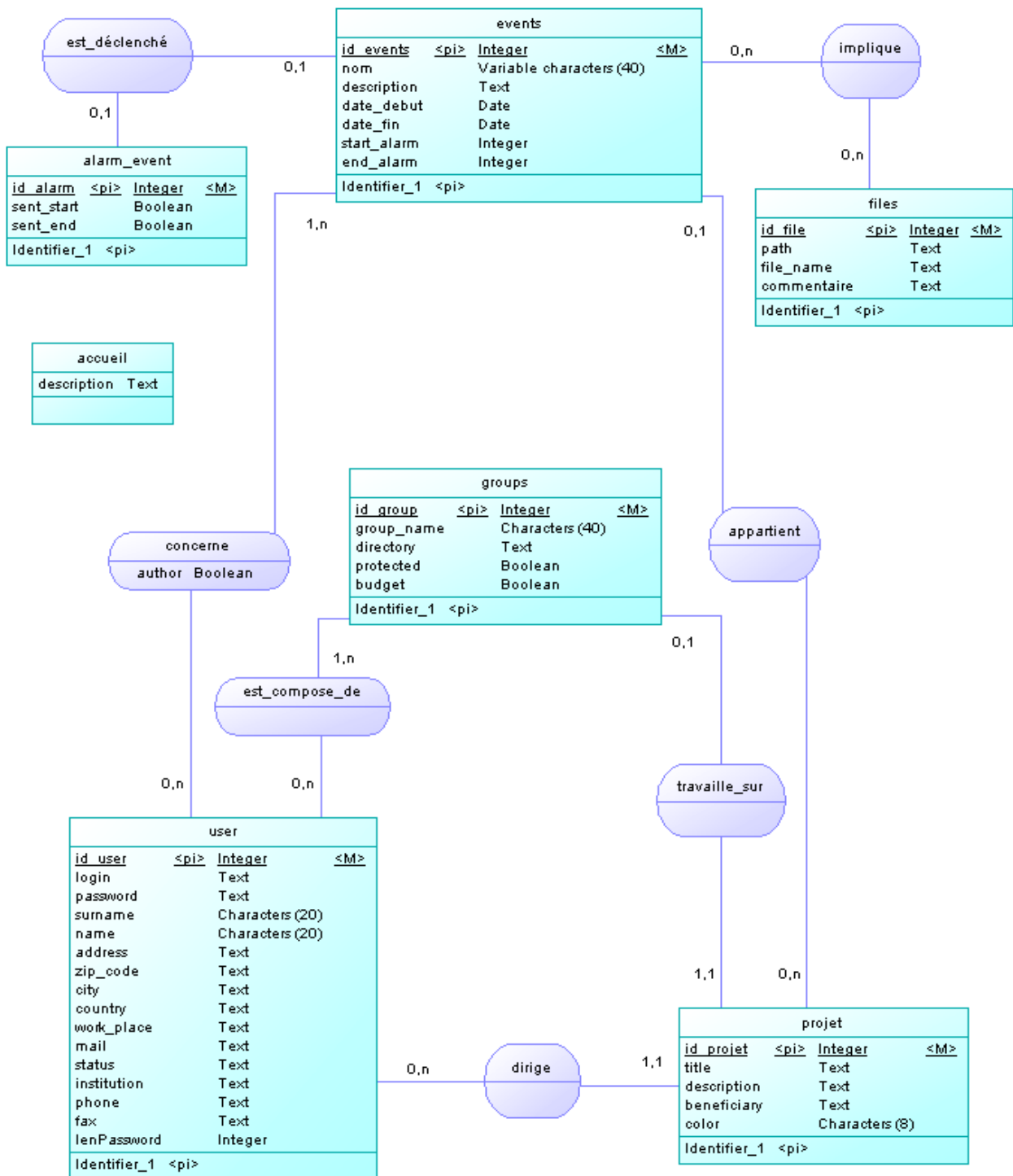
## **I.4 LA BASE DE DONNEES**

### *Les modifications de la base de données*

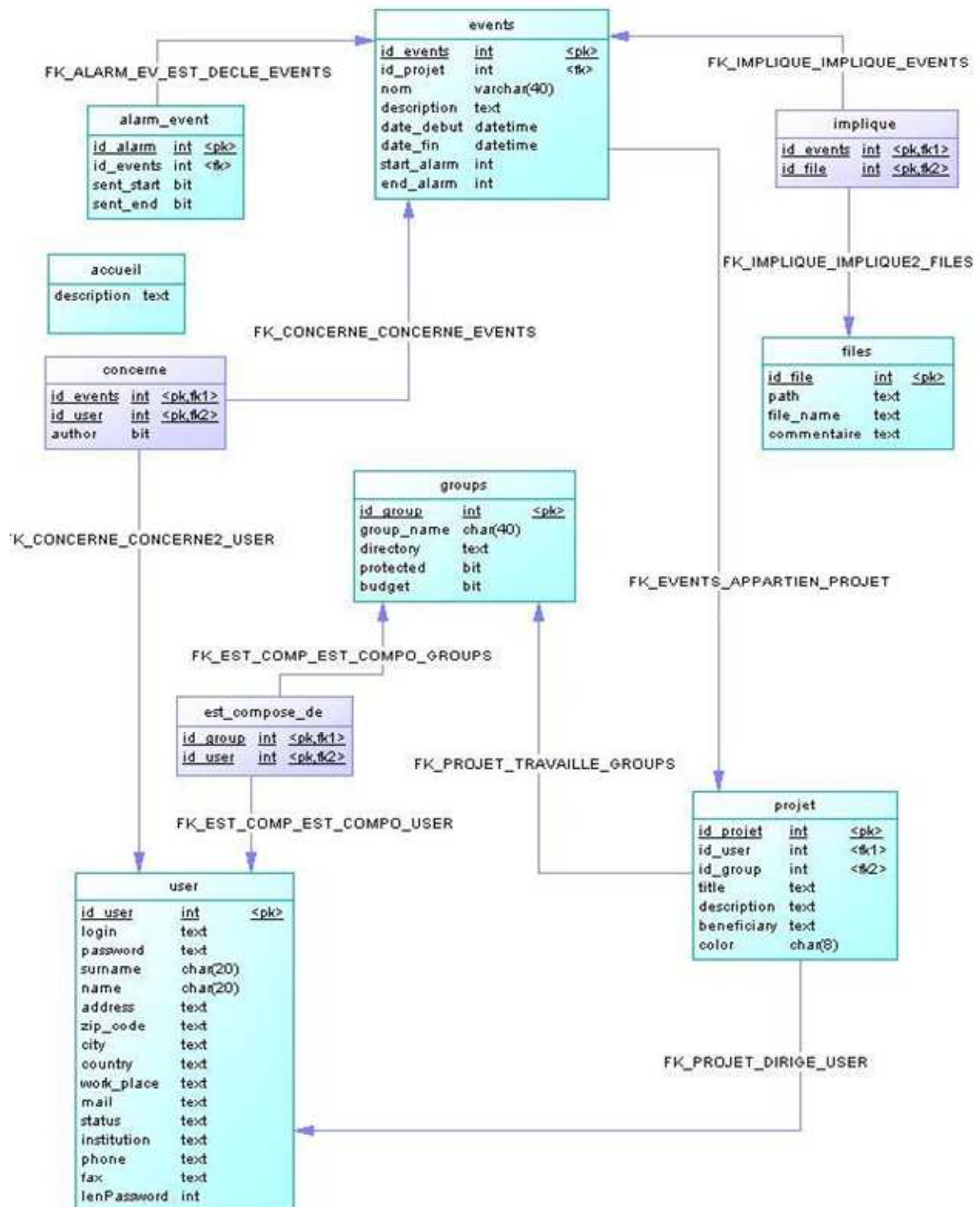
Après l'étude du cahier des charges et de la base de données existante j'ai trouvé nécessaire d'apporter quelques modifications sur la base de données.

- Dans l'application en production, tous les groupes ont un répertoire « budget ». Le client veut donner la possibilité aux groupes de choisir la création de ce répertoire.  
Pour enregistrer cette information j'ai ajouté le champ « budget » dans la table « groups ».
- Pour répondre à la demande faite par le client de pouvoir choisir différentes couleurs pour les projets, j'ai ajouté un champ « color » dans la table « projet »
- Dans la version initiale du projet, les mots de passe des utilisateurs était enregistrés en clair. Pour éliminer ce problème de sécurité nous avons choisi d'enregistrer dans la base de données les mots de passe crypté en utilisant le cryptage MD5.  
Au moment de l'affichage d'un compte d'utilisateur on doit afficher dans le champ « password » un nombre de « \* » correspondant à la longueur de mot de passe. Pour enregistrer dans la base de données la longueur des mots de passe j'ai ajouté un champ « lenPassword » dans la table « user ».
- La page d'accueil de l'application en production est une page statique. Le client souhaite pouvoir éditer cette page pour la mettre à jour avec diverses informations. Pour ça j'ai créé la table « accueil » qui contient le champ « description » où sera enregistré le code HTML de la page d'accueil.

### 1.4.1 MODELE CONCEPTUEL DES DONNEES



### I.4.2 MODELE PHYSIQUE DES DONNEES



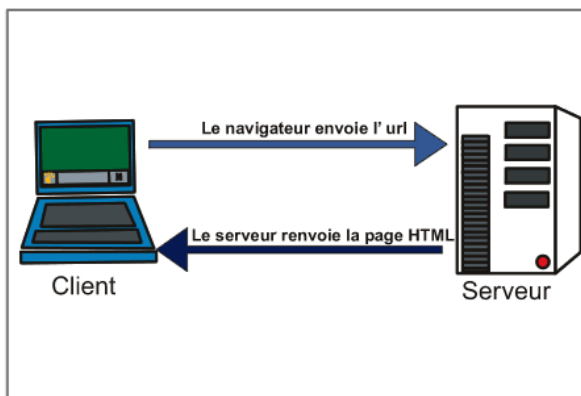
## I.5 CHOIX TECHNIQUE

Pour la conception de l'interface j'ai utilisé les langages PHP, JavaScript et HTML et comme système de gestion de base de données MySQL.

HTML (Hyper Text Markup Language) est un langage de balisage conçu pour représenter les pages web. Il est constitué de marqueurs que le navigateur comprend, analyse, et affiche.

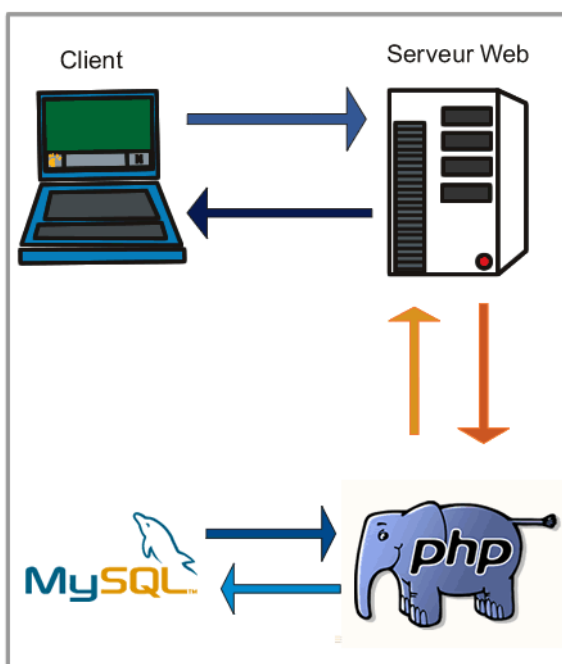
PHP est un langage de programmation destiné à être intégré dans des pages HTML. PHP est principalement dédié à la production de pages HTML générées dynamiquement.

### Communication entre un client et un server sans PHP



*Lorsqu'un client renvoie une requête HTTP pour consulter une page web, le serveur répond en renvoyant le contenu de la page.*

### Communication entre un client et un server avec PHP



*Quand un fichier avec une extension `.php` est demandé au serveur web, ce dernier le charge en mémoire et y cherche tous les scripts PHP qu'il transmet à l'interpréteur. L'interpréteur PHP exécute le script, en échangeant éventuellement des informations avec un système de base de données comme MySQL, ce qui a pour effet de produire du code HTML qui vient remplacer le script PHP dans le document finalement fourni au navigateur.*

*Ce dernier reçoit donc du HTML et ne voit jamais les instructions PHP.*

## II DEVELOPPEMENT

### **II.1 PRESENTATION DE L'INTERFACE**

Toutes les pages de l'application sont divisées en 3 parties : HEADER, MIDDLE et FOOTER.

Le HEADER est subdivisé en deux parties, une cellule contient le LOGO de l'application, et l'autre contient:

- pour le portail des images pour faire rappel sur les images de projet MAWARI.
- pour l'application, un tableau contenant les différents icônes du menu de l'application permettant de naviguer à travers les divers modules demandés par le client.

Le MIDDLE, c'est la partie utilisée pour afficher les modules de l'application.

Le FOOTER contient le logo de la société créatrice de la plate-forme (INDIGO GLOBAL SERVICES) et les logos des sociétés partenaires du projet (CIFEG, coordinateur du projet et Ministère des Affaires Etrangères, sponsor du projet).

## LE PORTAIL



Le « MIDLLE » de cette page contient quatre tableaux représentant les différents pays, ainsi que des liens vers les organismes travaillant sur le projet « MAWARI ». Au centre se trouve le formulaire de login.

L'utilisateur doit saisir son identifiant et son mot de passe pour pouvoir accéder à l'application.

Si il valide le formulaire (bouton « Ok ») et l'identifiant et le mot de passe saisis sont corrects, il accède à l'application, dans le cas contraire, un message d'erreur s'affiche.

ACCUEIL

Le header de la page « ACCUEIL » (et des tous les autres pages de l'application sauf le « portail ») affiche le logo de l'application et une barre de navigation.

La barre de navigation permet à l'utilisateur de naviguer entre les différents modules.

Dans la première version de l'application, le module « Administration » n'est visible que pour les utilisateurs qui ont le statut d'administrateur.

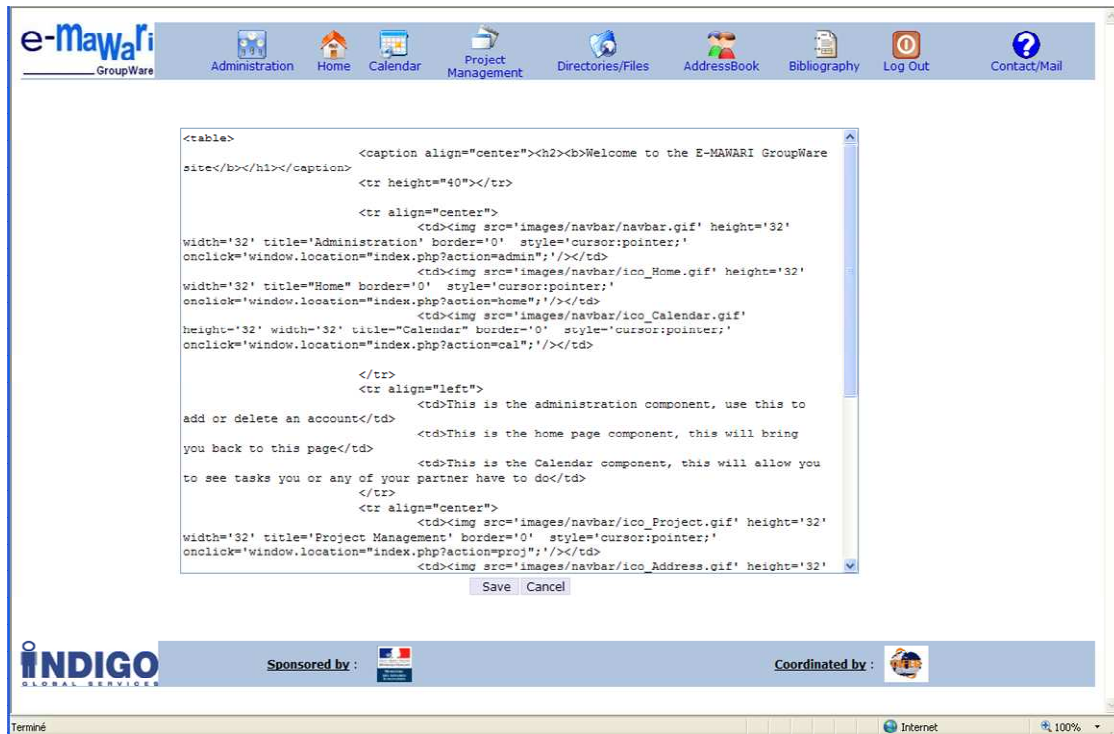
Au centre de page, il y a un descriptif de chaque module de l'application.

Conforme au cahier des charges, le client souhaite donner accès à tous les utilisateurs au module « Administration » (avec des différents droits en fonction de leur statut).

Après avoir fait les modifications nécessaires, le module « Administration » est visible et accessible à tous les utilisateurs.

Le client a souhaité avoir la possibilité d'éditer la page afin de pouvoir la mettre à jour avec diverses informations.

Pour cela j'ai ajouté le bouton « Modify » qui est visible pour les utilisateurs qui ont le statut d'administrateur. L'action de ce bouton est d'ouvrir la page « ACCUEIL » un mode édition.



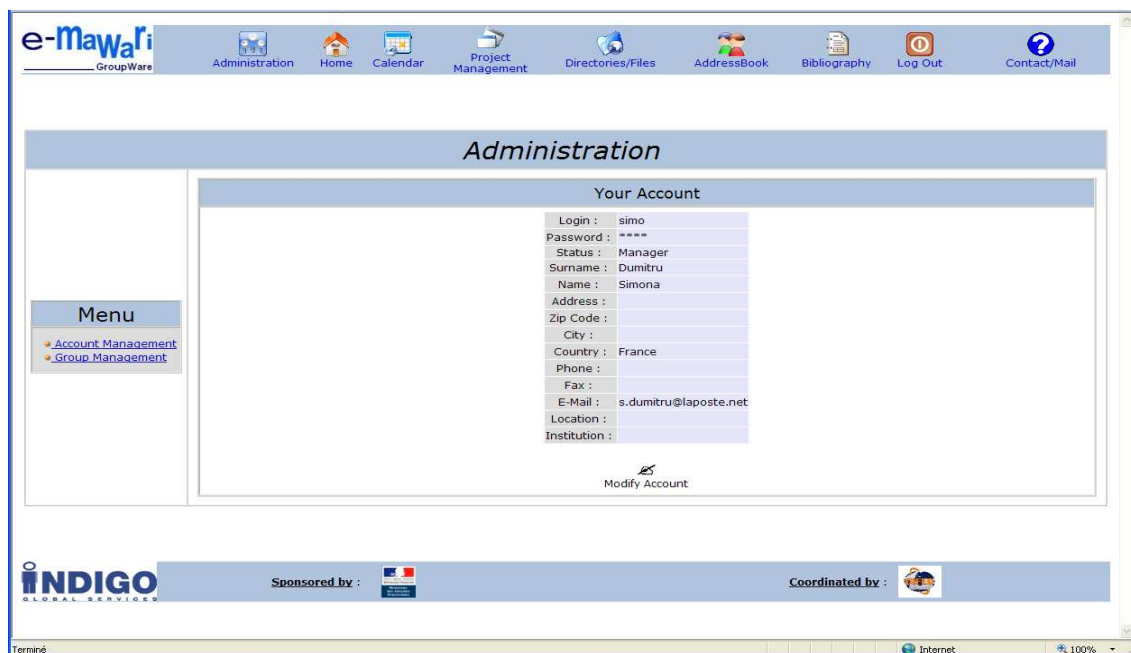
En mode édition la page « ACCUEIL » affiche une zone de texte qui contient le code HTML de la page. Ce code peut être modifié et sauvegardé dans la base de données, pour mettre à jour la page « ACCUEIL ».

L'action de bouton « Save », est de sauvegarder les données de la zone de texte dans la base de données et afficher la page « ACCUEIL » en mode visualisation.

L'action de bouton « Cancel » est d'afficher la page « ACCUEIL » en mode visualisation sans enregistrer les modifications.

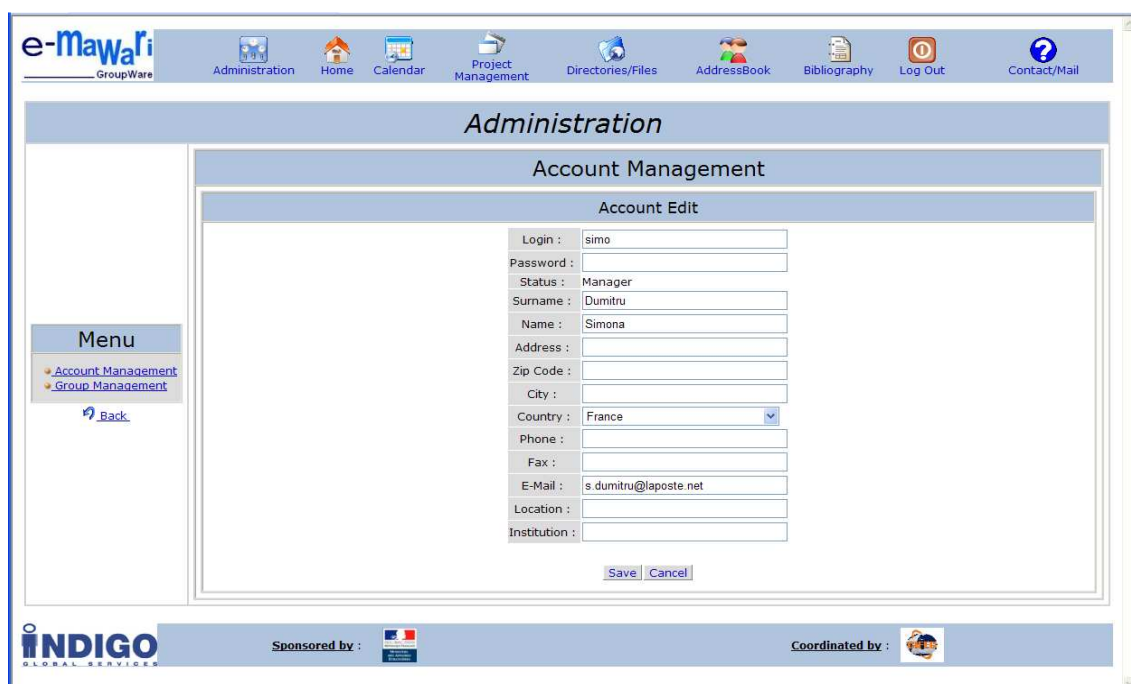


## ADMINISTRATION



Le module « Administration » est composé d'un menu permettant d'accéder à la gestion des comptes et à la gestion des groupes d'utilisateurs et d'une partie d'affichage qui, lors du premier lancement du module, affiche le compte de la personne connectée.

Le lien « Modify Account » ouvre la page en mode édition pour pouvoir modifier le compte.



Dans la première version de l'application, seuls les utilisateurs qui ont le statut « Admin » ont accès au module « Administration ».

Le client souhaite que tous les utilisateurs puissent visualiser et modifier leurs données personnelles (sans pouvoir modifier leur statut) et que les « Manager leader » des groupes, puissent créer, supprimer et modifier des comptes pour leurs groupes de travail.

### Réalisation

Le menu pour accéder à la gestion des comptes ou à la gestion des groupes d'utilisateurs est affiché que pour l'administrateur et pour les « managers » qui sont leader de projet.

Un projet peut avoir un seul leader qui est choisi au moment de la création de projet, parmi les managers de projet.

Les autres utilisateurs ont accès qu'à leur propre compte qu'ils peuvent visualiser et modifier.

```
<?php
$requeteLider="select id_user from projet where id_user=".$_SESSION['id_user']."' group by id_user";
$reponseLider=query($requeteLider);
$row = mysql_num_rows($reponseLider);

/* menu_admin est affiché pour Admin ou Manager(Lider)*/
if((($_SESSION['user_status']=='Admin') || ((($_SESSION['user_status']=='Manager') && ($row==1))))
(
?>
<td style='border-width:2px; border-style:solid; border-color:lightgrey; border-right:none;'>
<?php
/* Affichage du menu */
include "../admin/menu_admin.php";
?>
</td>
<?php
)
}
```

The screenshot shows the 'Administration' section of the e-mawa.fr application. The top navigation bar includes icons for Administration, Home, Calendar, Project Management, Directories/Files, AddressBook, Bibliography, Log Out, and Contact/Mail. The main content area is titled 'Administration' and contains a 'Your Account' section. This section displays a list of user details in a table-like format:

Login :	user
Password :	****
Status :	User
Surname :	user_surname
Name :	user_name
Address :	addressUser
Zip Code :	
City :	
Country :	France
Phone :	
Fax :	
E-Mail :	
Location :	
Institution :	

Below the table, there is a 'Modify Account' link with a pencil icon.

- **Gestion des comptes d'utilisateurs**

Pour les utilisateurs qui le statut « Admin » la partie gestion des comptes d'utilisateurs « Account Management », affiché la liste de tous les utilisateurs enregistrés dans la base de données avec la possibilité d'ajouter (le bouton « Create ») des nouveaux comptes ou d'effacer (le bouton « Delete ») les comptes sélectionnées.

The screenshot shows the 'Administration' section of the e-mawafi GroupWare interface. The main content area is titled 'Account Management' and displays a list of users. Each user entry includes a checkbox, the user's name, and a 'View' link. At the bottom of the list are 'Create' and 'Delete' buttons. The interface also features a top navigation bar with various icons and a left sidebar with a 'Menu' containing 'Account Management', 'Group Management', and a 'Back' button. The footer includes the INDIGO logo, 'Sponsored by' with a French flag, and 'Coordinated by' with a globe icon.

<input type="checkbox"/>	Account name	<a href="#">View</a>
<input type="checkbox"/>	Bonnot Nicolas	<a href="#">View</a>
<input type="checkbox"/>	Pinard François	<a href="#">View</a>
<input type="checkbox"/>	Jalludin Mohamed	<a href="#">View</a>
<input type="checkbox"/>	Orlyk Sylvie	<a href="#">View</a>
<input type="checkbox"/>	Barongo Justus	<a href="#">View</a>
<input type="checkbox"/>	Berhanu Gizaw	<a href="#">View</a>
<input type="checkbox"/>	Dagnachew Legesse	<a href="#">View</a>
<input type="checkbox"/>	Giséle Blain	<a href="#">View</a>
<input type="checkbox"/>	Dubié Cédric	<a href="#">View</a>
<input type="checkbox"/>	Laval Michel	<a href="#">View</a>

Réalisation

Dans la nouvelle version de l’application, les utilisateurs qui ont le statut de « Manager leader » d’un groupe ont eux aussi accès à la gestion des comptes d’utilisateurs pour leur groupe.

Si un utilisateur est le « Manager leader » des plusieurs groupes, la page affiche les noms des groupes pour lesquels il a ce statut, et les utilisateurs qui font partie de chacun de ces groupes.

Les requêtes pour sélectionner les groupes et les membres des groupes respectifs sont :

```

<?php
if($_SESSION['user_status']=='Manager')
{

    $requeteUser="SELECT groups.id_group,groups.group_name"
    ." FROM groups, user, projet "
    ." WHERE user.id_user = projet.id_user "
    ." AND projet.id_group = groups.id_group "
    ." AND user.id_user =" .$_SESSION['id_user'];
    $resUser=query($requeteUser);

    while($donneesUser = mysql_fetch_assoc($resUser))
    {
        $requete="SELECT user.id_user, user.surname, user.name FROM groups, user, user_group "
        ." WHERE user.id_user = user_group.id_user "
        ." AND user_group.id_group = groups.id_group "
        ." AND groups.id_group =" . $donneesUser['id_group'] ;

        $res=query($requete);
    }
}
    
```

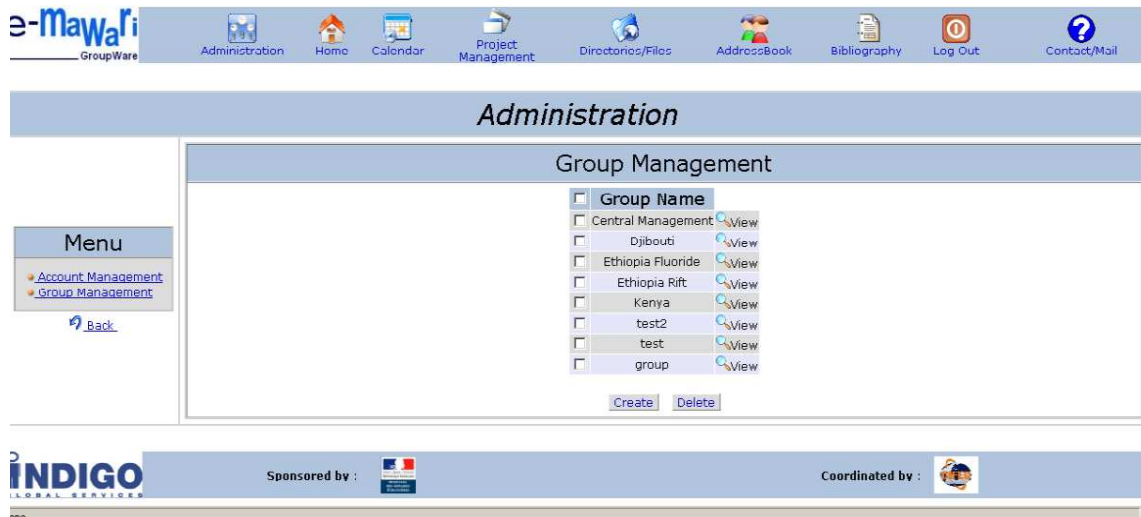


Le « Manager leader » peut créer des nouveaux comptes d’utilisateurs pour son groupe, mais il n’a pas le droit de supprimer des comptes.

- **Gestion des groupes d'utilisateurs**

*Affichage de la liste des groupes*

Dans le cas ou l'utilisateur connecté a le statut « Admin », la gestion des groupes d'utilisateurs « Group Management » affiche la liste des tous les groupes avec la possibilité de créer de nouveaux groupes, de supprimer ou de visualiser en détails les groupes.



*Réalisation*

Si l'utilisateur connecté a le statut « Manager leader », la gestion des groupes d'utilisateurs affiche la liste des groupes pour lesquelles il a ce statut. Il peut visualiser en détails les groupes, mais il n'a pas le droit de créer des nouveaux groups ou de supprimer des groupes.



### *Affichage des détails du groupe sélectionné*

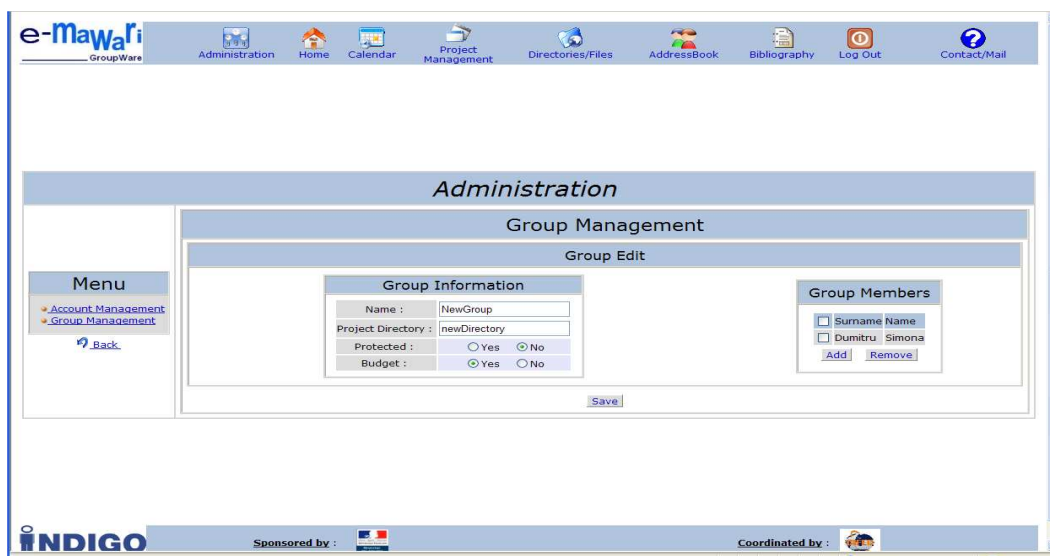
Dans les détails du groupe d'utilisateurs, on retrouve les informations concernant le groupe et la liste de membres du groupe. On peut ajouter ou supprimer des membres pour ce groupe.

Lors de la création d'un groupe d'utilisateurs, un répertoire de travail est automatiquement créé, dans lequel est créé un répertoire « Budget ».

Le client souhaite qu'un groupe ait la possibilité de choisir s'il veut ou non avoir le répertoire « Budget ».

### Réalisation

Pour cela j'ai ajouté deux radios boutons qui permettent la création ou non du répertoire « Budget ».



Avec l'ajout de choix du répertoire « Budget », j'ai dû résoudre plusieurs contraintes :

- Il faut impérativement qu'un groupe ait un « Projet Directory » pour pouvoir avoir un répertoire « Budget »
- Si le « Projet Directory » d'un groupe est supprimé, alors le répertoire « Budget » est lui aussi supprimé.
- Le répertoire « Budget » (comme tous les autres répertoires d'un groupe) peut être créé ou supprimé aussi dans le module « Directories/Files ». Donc si une telle action a lieu, il faut faire les modifications nécessaires dans la base de données.

## PROJECT MANAGEMENT

### *Affichage liste de projets*

Le module gestionnaire des projets permet à tous les utilisateurs de voir la liste des projets, les détails des projets et en plus pour les utilisateurs qui ont le statut « Admin », leur permettre de créer et de supprimer des projets.

Dans la première version de l'application, tous les projets (même celles qui ont été défini en tant que projet « protégé ») sont visibles pour tous les utilisateurs.

<input type="checkbox"/>	Project Name	Description	Project Leader	Beneficiary	
<input type="checkbox"/>	Central Management	Management of tasks and documents not related to a specific project	Pinard François	Cifeg	<a href="#">View</a>
<input type="checkbox"/>	Dilboubi	Gestion durable de la ressource en eau dans le système du Rift est-africain. Application au site expérimental d'Atar	Jalludin Mohamed	Cifeg	<a href="#">View</a>
<input type="checkbox"/>	Ethiopia Fluoride	Origin, genesis and distribution of fluoride in the Ethiopian Rift and development of defluoridation technologies.	Berhanu Gizaw	Geological Survey	<a href="#">View</a>
<input type="checkbox"/>	Ethiopia Rift	Systematic groundwater investigation in selected sites of the Ethiopian Rift: Towards an understanding of the hydrogeology of the Rift and adjacent highlands.	Dagnachew Legesse	Addis Ababa university	<a href="#">View</a>
<input type="checkbox"/>	Kenya	Vulnerability of groundwater to pollution in the East African Rift System in Kenya.	Barongo Justus	University Of Nairobi	<a href="#">View</a>
<input type="checkbox"/>	projet	projet	Nom Prenom		<a href="#">View</a>
<input type="checkbox"/>	ProjetGroup	descrGroup	Dumitru Simona		<a href="#">View</a>
<input type="checkbox"/>	test	test	Nom Prenom		<a href="#">View</a>

[Create](#) [Delete](#)

**INDIGO** GLOBAL SERVICE  
Sponsored by : Coordinated by :

Le client souhaite qu'un projet « protégé » ne soit visible que pour les membres de ce projet.

### Réalisation

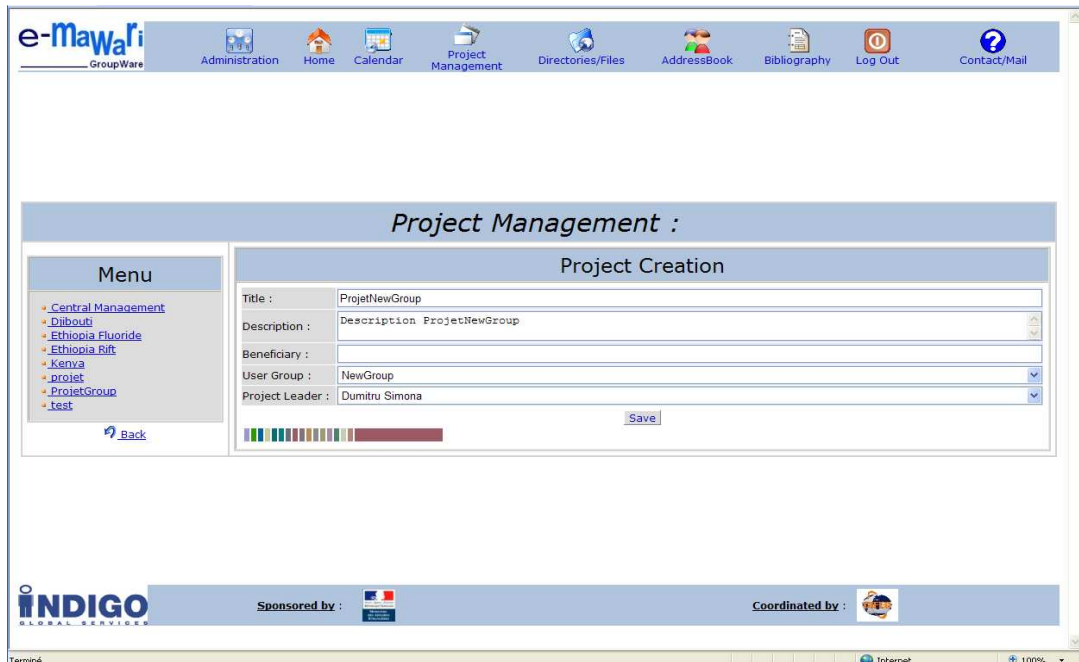
Pour répondre à cette demande j'ai fait une requête qui sélectionne tous les projets qui ne sont pas protégé et les projets protégés qui ont comme membre l'utilisateur connecté.

```
if($_SESSION['user_status']!='Admin')
{
$res=query("select * from projet order by title");
}
else
{
$res=query('SELECT *
. FROM projet, groups'
. WHERE projet.id_group = groups.id_group'
. AND groups.id_group NOT
. IN (
. SELECT groups.id_group'
. FROM groups, user_group'
. WHERE groups.protected =1'
. AND groups.id_group not in'
. (SELECT groups.id_group'
. FROM groups, user_group'
. WHERE groups.id_group = user_group.id_group'
. AND groups.protected =1'
. AND user_group.id_user = '.$_SESSION['id_user'].'')
. )'
. group by groups.id_group'
. order by projet.title');
}
$i=0;
while($donnees=mysql_fetch_array($res))
{
```



### Création nouveau projet

Pour répondre à la demande du client de pouvoir choisir une couleur différente pour chaque projet, dans la page de création de nouveau projet j'ai ajouté une liste des couleurs d'où l'utilisateur peut sélectionner la couleur souhaitée pour son projet.



```

<?php
if($_GET['manage']==0)
{
    $couleur_ligne = array('9999CC','339900','006699','CCCC99','0C6E75','007F83',
    '676F76','9C5A66','847376','B58C5E','8A8A8A','9C9D76','AC8BA9','4C8470','C8CDB5','AE8979');
    for($i=0;$i<count($couleur_ligne);$i++)
    {
        ?>
        <td align='left' bgcolor="<?php echo $couleur_ligne[$i] ?>"
        style='cursor:pointer' onclick="document.getElementById('divColor').style.backgroundColor = '<?php echo $couleur_ligne[$i]
        tdColor.bgColor='<?php echo $couleur_ligne[$i] ?>'; txColor.value='<?php echo $couleur_ligne[$i] ?>'; ">&nbsp;</td>
        <?php
    }
    ?>
    <td id=tdColor width=100><div id=divColor><input id=txColor name='color' type='hidden'></div></td>
    <?php
}
?>

```

## CALENDAR

L'affichage de tâches de chaque projet est réalisé en ce module. Nous avons la possibilité de visualiser les tâches par mois ou par semestre.

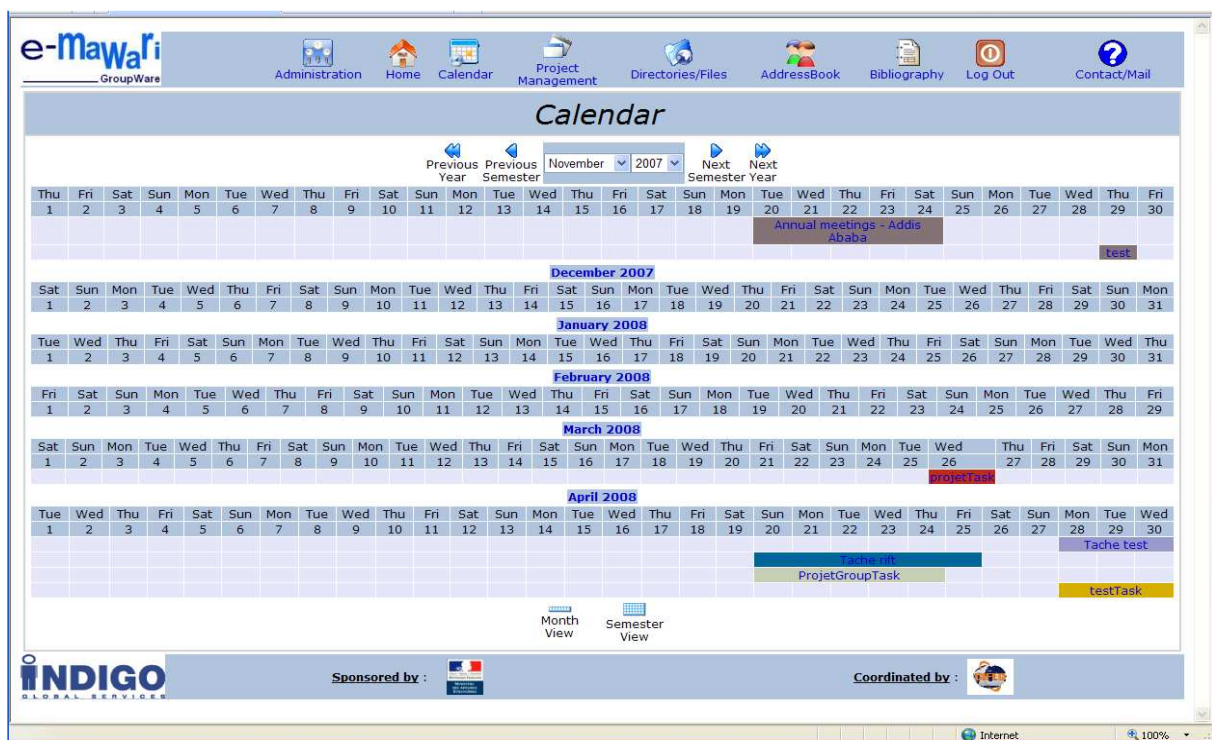
Dans la première version de l'application, l'affichage des tâches se fait dans une couleur unique.

Le client souhaite que les taches s'affichent dans la couleur correspondant au projet auquel elles sont rattachées.

### Réalisation

Pour chaque tache je sélectionne la couleur du projet auquel elle est rattachée et je l'utilise pour l'affichage.

```
$res=query("select projet.color, projet.id_projet from projet,
events where projet.id_projet=events.id_projet and events.id_events=".$events[$i][$j]['id_events']");
$data=mysql_fetch_array($res);
echo "<td bgcolor='". $data['color'] ."'
onclick='window.location=\"index.php?action=cal&task=".$events[$i][$j]['id_events']."\";'
colspan='". $nb_col ."' title='". $events[$i][$j]['description'] ."' onmouseover='changelink(this,true);'
onmouseout='changelink(this,false);'
style='cursor:pointer; color:MediumBlue; text-decoration:none;'>". $events[$i][$j]['nom'] . "</td>";
```

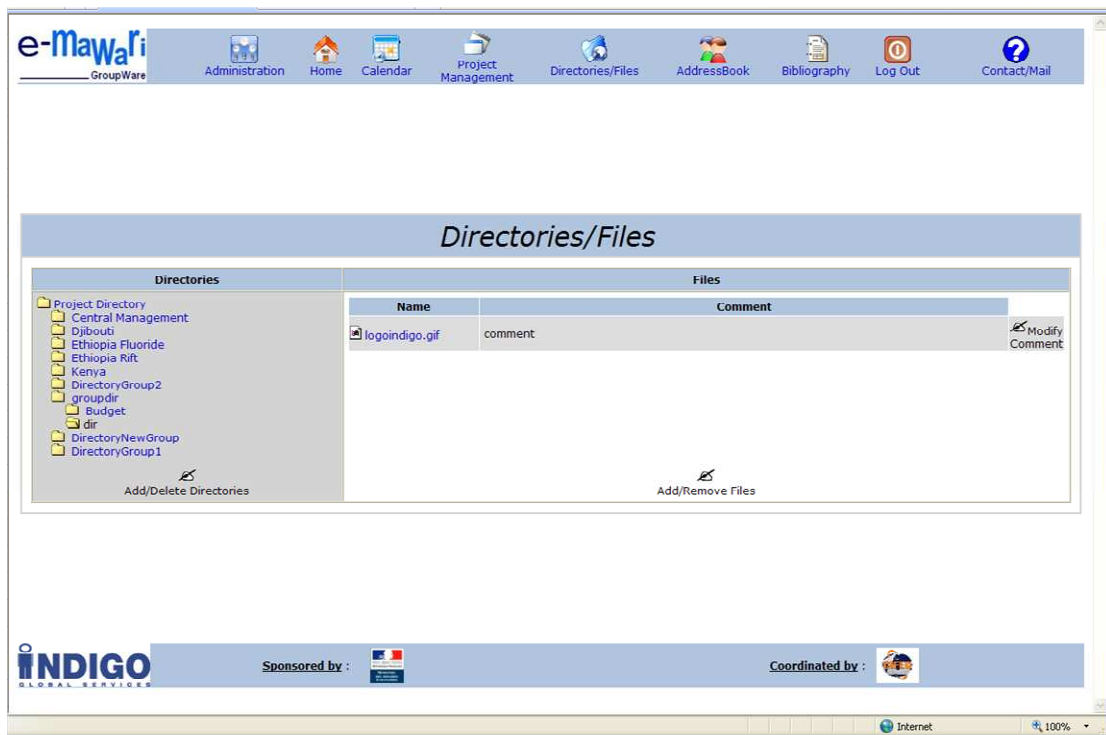


L'affichage des tâches dépend du statut de l'utilisateur qui est connecté.

Pour les administrateurs toutes les tâches sont affichées.

Pour les autres utilisateurs sont affichées les tâches rattachées aux projets non protégés et celles rattachées aux projets protégés qui ont comme membre cet utilisateur.

**DIRECTORIES/FILES**



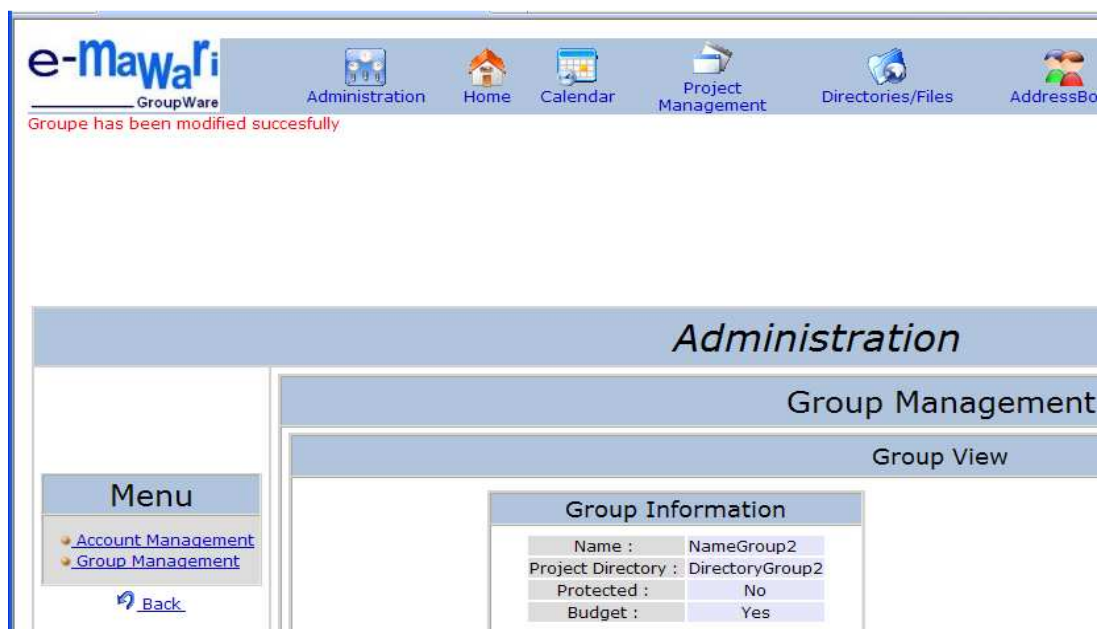
Le module « Directories/Files » permet de visualiser les répertoires de travail des différents projets (sauf ceux qui sont en mode « protégé », qui eux ne sont pas accessibles qu’aux utilisateurs concernés et aux administrateurs).

## LES MESSAGES ET LES BOÎTES DE DIALOGUE

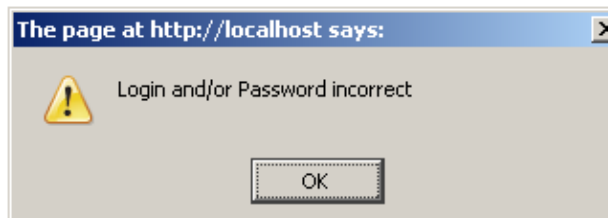
Pour avertir l'utilisateur quand il y a une action de modification ou de suppression, j'ai créé des boîtes de dialogue.



Si une action est exécutée correctement, un message est affiché pour notifier l'utilisateur.



Sinon, une alerte indique la nature d'erreur qui s'est produite.



## **II.2 LE CODE**

### *Mise à jour de la base de données en production*

Pour mettre à jour la base de données en production j'ai réalisé un script PHP qui fait:

- la connexion à la base de données

```
//connexion à la base de données

$link = mysql_connect('localhost','root','');
if (!$link) {
    die('Impossible de se connecter : ' . mysql_error());
}

$db_selected = mysql_select_db('mawari', $link);
if (!$db_selected) {
    die ('Impossible de sélectionner la base de données : ' . mysql_error());
}
```

- la création de la table « accueil » et l'initialisation de cette table avec les données initiales.

```
//creation table accueil

$sql = "CREATE TABLE if not exists accueil (description text)";
mysql_query($sql, $link);
```

- l'ajout de champ « budget » dans la table « groups » et la mise à jour de la table

```
//ajout de champ budget dans la table groups

$sql="ALTER TABLE groups ADD budget tinyint(1)";
mysql_query($sql, $link);

//update table groups( insertion de budget)

$sql="update groups set budget=1 ";
mysql_query($sql);
```

- l'ajout de champ « color » dans la table « projet » et le la mise à jour de la table

```
//ajout de champ color dans la table projet

$sql="ALTER TABLE projet ADD color varchar(8) ";
mysql_query($sql,$link);

//update table projet(insertion des couleurs)

$sql="update projet set color='#8A7E6B' where id_projet=22";
mysql_query($sql);
$sql="update projet set color='#12AD2B' where id_projet=24";
mysql_query($sql);
$sql="update projet set color='#94A978' where id_projet=25";
mysql_query($sql);
$sql="update projet set color='#58ACD1' where id_projet=26";
mysql_query($sql);
$sql="update projet set color='#008ED2' where id_projet=27";
mysql_query($sql);
```

- l'ajout de champ « lenPassword » pour la table « user »

Pour mettre à jour la base de données avec les mots de passes cryptés, j'ai sélectionné les mots de passes qui étaient enregistrés dans la table « user », j'ai fait leur cryptage et j'ai actualisé la table « user ».

```
//ajout champ lenPassword dans la table user

$sql="ALTER TABLE user ADD lenPassword int(2) ";
mysql_query($sql,$link);

//update champ lenPassword and password MD5 dans la table user avec les longueurs de password avant le cryptage et le password crypté

$q = mysql_query("SELECT CHAR_LENGTH(password) as length, id_user ,password from user ");
while($donnees=mysql_fetch_array($q))
{
    $passwordCrypt= md5($donnees['password']);
    $sql="update user set password='".$passwordCrypt."',
    ."lenPassword='".$donnees['length']."' where id_user='".$donnees['id_user']";
    mysql_query($sql);
}
```

## La gestion des erreurs

Pour ajouter des messages de notification d'utilisateur lorsqu'une action est correctement exécutée ou non, j'ai créé la fonction `form_hidden($adresse, $value)`. Les paramètres de cette fonction sont l'adresse de la page sur laquelle le message doit s'afficher et le type d'action exécutée.

```
<?php
function form_hidden($adresse, $value)
{
    echo "<form action='index.php?'.$adresse.'" method='post'>";
    echo "<input type='hidden' name='save' value='".$value.'" />";
    echo "</form>";
    echo "<script>document.getElementsByTagName('form')[0].submit();</script>";
}
?>
```

Chaque fois quand je dois afficher un message j'attribue des valeurs aux paramètres et j'appelle la fonction `form_hidden`.

```
$adresse='action=admin&manage=1';
$value='deleteGroupe';
form_hidden($adresse, $value);
```

La fonction `alert_manager()` gère l'affichage de messages.

L'appelle de la fonction est fait une seule fois dans le fichier « `index.php` ».

```
<?php
function alert_manager()
{
    if(isset($_POST['save']))
    {
        switch($_POST['save'])
        {
            case 'update' : echo 'Account has been modified succesfully';
                        break;
            case 'create' : echo 'Account has been created succesfully';
                        break;
            case 'delete' : echo 'Account has been deleted succesfully';
                        break;
            case 'updateProj' : echo 'Project has been modified succesfully';
                        break;
            case 'createProj' : echo 'Project has been created succesfully';
                        break;
            case 'deleteProj' : echo 'Project has been deleted succesfully';
                        break;
            case 'createProjTitle' : echo "<script> alert('You must chose a name for your projet');</script>";
        }
    }
}
```

# PROJET TABLEAU DE BORD



# I THEME DU PROJET ET CAHIER DE CHARGE

## **I.1 DESCRIPTIF DU PROJET**

L'objectif du deuxième projet sur lequel j'ai travaillé pendant mon stage en entreprise a été la réalisation d'un outil de gestion des projets et des temps affecté pour les différentes activités.

## **I.2 CHOIX TECHNIQUE**

Pour acquérir des nouvelles compétences de développement web nous avons choisi la technologie ASP.NET.

Ce choix technique est motivé par le fait que l'équipe de développement utilise cette technologie pour plusieurs projets.

ASP.NET est un ensemble de technologies de programmation web créé par Microsoft.

ASP.NET est utilisé pour la création des sites web dynamiques, des applications web ou des services web.

Pour développer des applications ASP.NET on peut utiliser tous les langages de programmation supportés par le Framework .NET : C#, Visual Basic .NET, JScript.NET.

Comme outil de développement j'ai utilisé Visual Web Developer.

Visual Web Developer est un environnement de développement d'applications Web en ASP.Net aussi facile à utiliser pour la saisie du code (coloration syntaxique, débogueur graphique ...) que pour la création visuelle des applications.

Ce logiciel intègre également des outils de gestion visuelle de bases de données associés à SQL Server Express Edition.

### **I.3 CAHIER DES CHARGES**

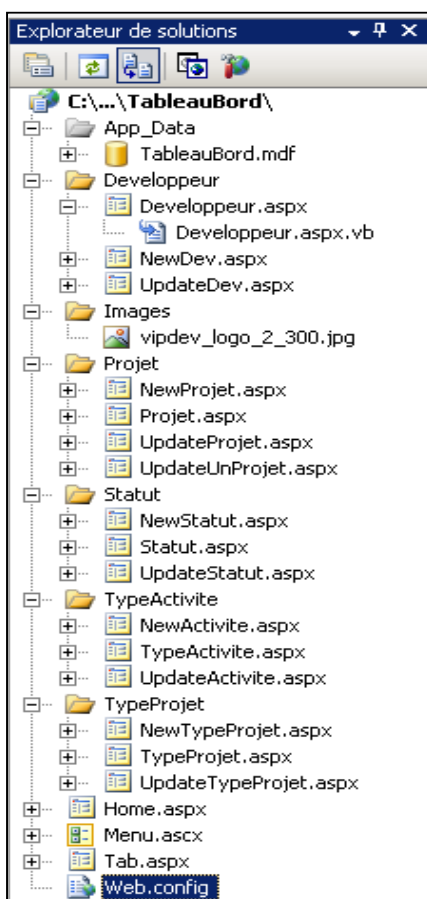
*Fonctionnement souhaité :*

L'application doit donner la possibilité à l'utilisateur de :

- Modifier, supprimer et créer des projets
- Modifier, supprimer et créer des comptes des développeurs
- Modifier, supprimer et créer des types de projets
- Modifier, supprimer et créer des types d'activités
- Mettre à jour les durées réelles et prévisionnelles destinées au développement des projets
- Afficher le rapport des durées par projet/activité/développeur

### **I.4 L'ARBORESCENCE DU SITE**

L'arborescence du site est :



Le répertoire App\_Data est utilisé par ASP.NET pour stocker la base de données locale de l'application. Ce répertoire est créé automatiquement à la création du site.

La partie graphique des pages (Code HTML + Contrôles) est enregistrée dans les pages .aspx (ex : Développeur.aspx) et la partie code (VB.NET) dans les pages .aspx.vb (ex : Développeur.aspx.vb). Les pages Développeur.aspx et Développeur.aspx.vb ne feront plus qu'une fois compilées.

Le fichier Menu.aspx est un fichier de type « Contrôle utilisateur Web ».

Le fichier Web.config contient la chaîne de connexion à la base de données.

```
<connectionStrings>
  <add name="TableauBordConnectionString2"
    connectionString="Data Source=SQL-VM1;Initial Catalog=TableauBord;User ID=sa;Password=simo"
    providerName="System.Data.SqlClient" />
</connectionStrings>
```

## **I.5 LA BASE DE DONNEES**

La première partie du travail a consisté dans l'analyse et la mise en place de la base de données.

Pour la réalisation de la base de données j'ai choisi la méthode Merise.

### *Le dictionnaire de données*

Prénom développeur

Alias développeur

Statut développeur

Nom projet

Client projet

Type projet

Nom activité

Durée prévisionnelle

Durée réelle

### *Analyse*

Un Développeur peut travailler sur plusieurs projets.

Sur chacun des projets il peut travailler sur plusieurs activités.

Pour chaque développeur/ activité /projet on a deux types de durées : réelle et prévisionnelle.

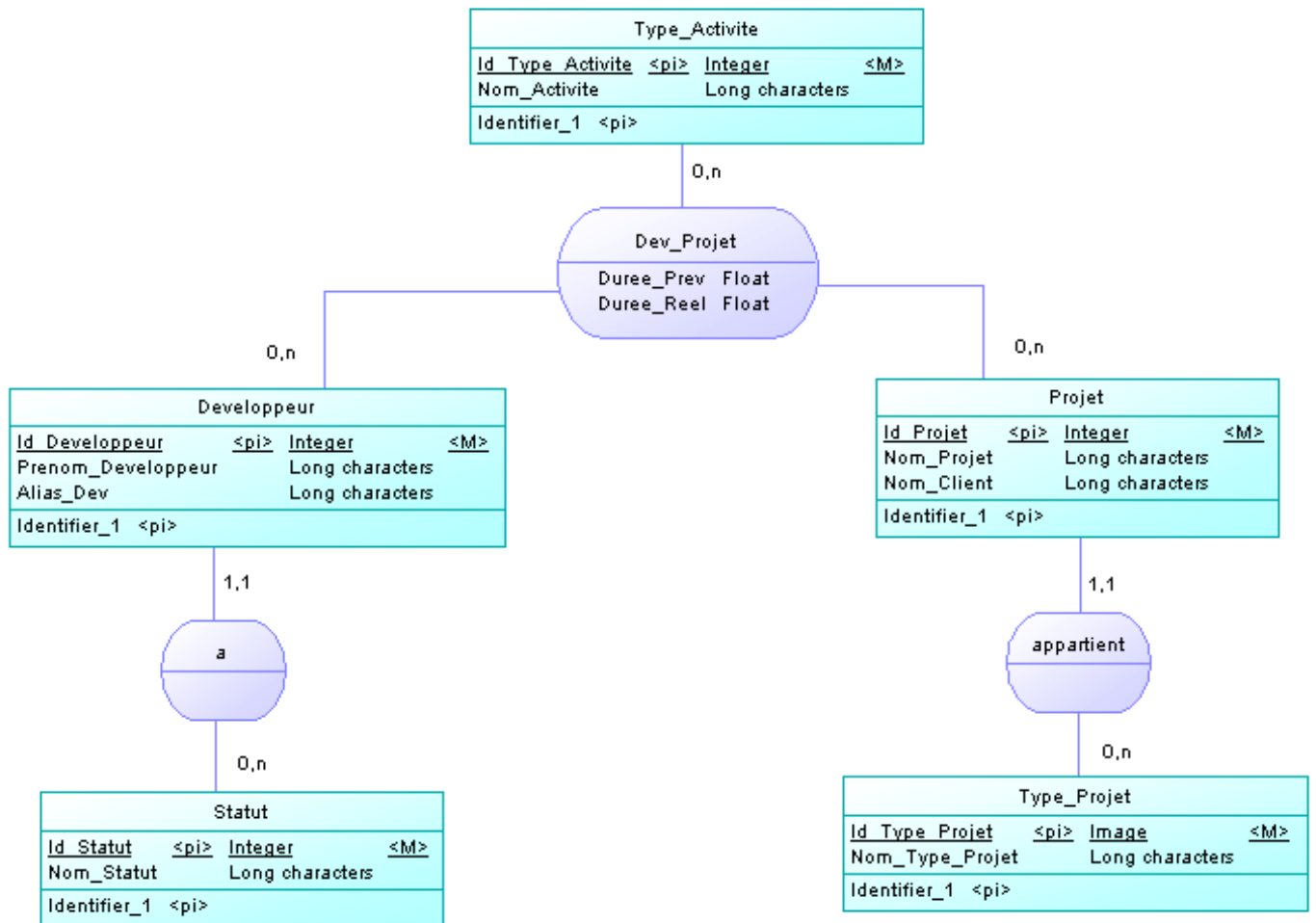
La durée réelle représente les jours travaillés sur une activité d'un projet.

La durée prévisionnelle représente les jours qu'un développeur prévoit travailler sur une activité d'un projet.

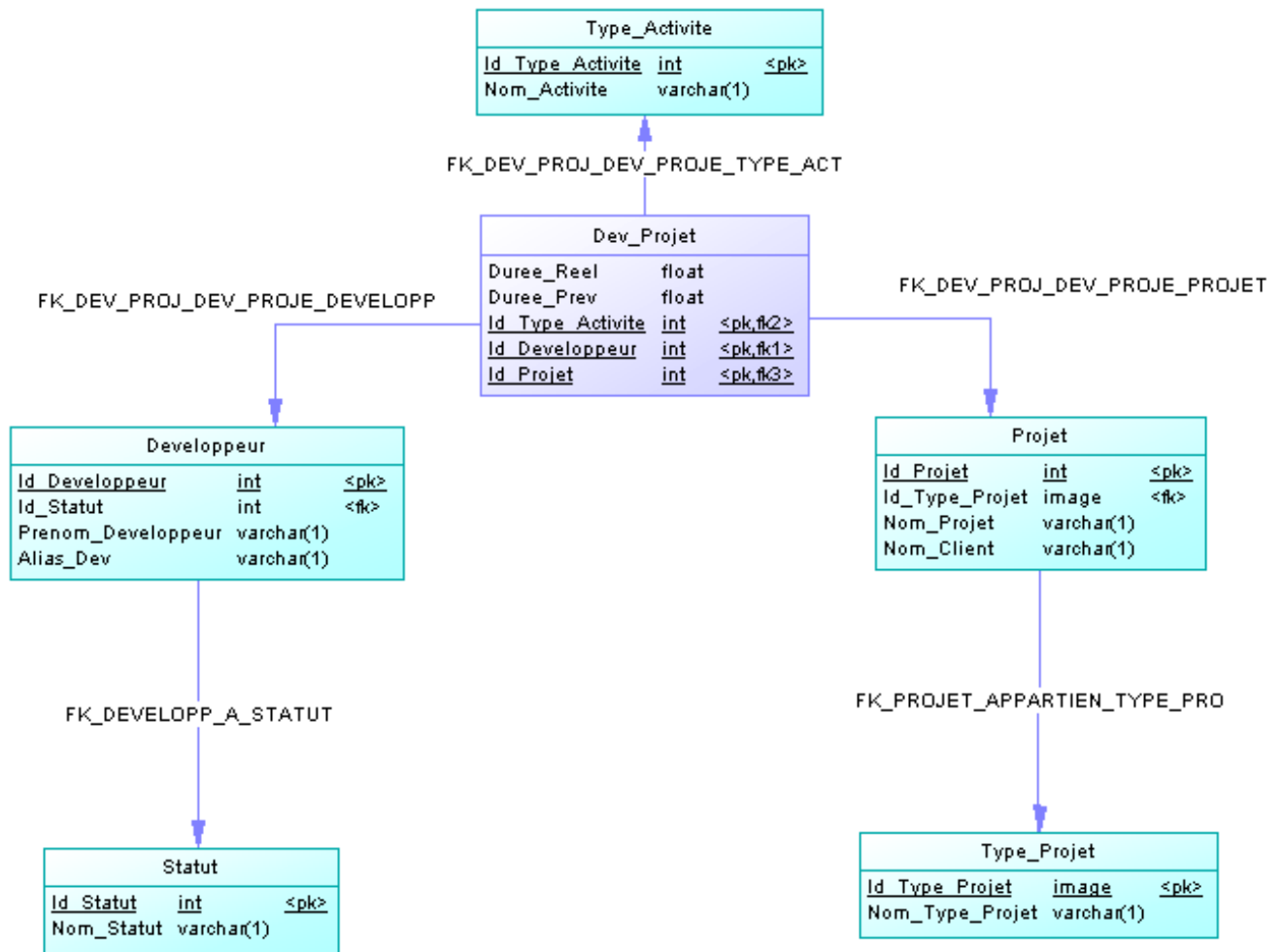
Un développeur peut avoir un seul statut et un statut peut être pour plusieurs développeurs.

Un projet peut avoir un seul type de projet et un type de projet peut être pour plusieurs projets.

### I.5.1 MODELE CONCEPTUEL DES DONNEES



### I.5.2 MODELE PHYSIQUE DES DONNEES





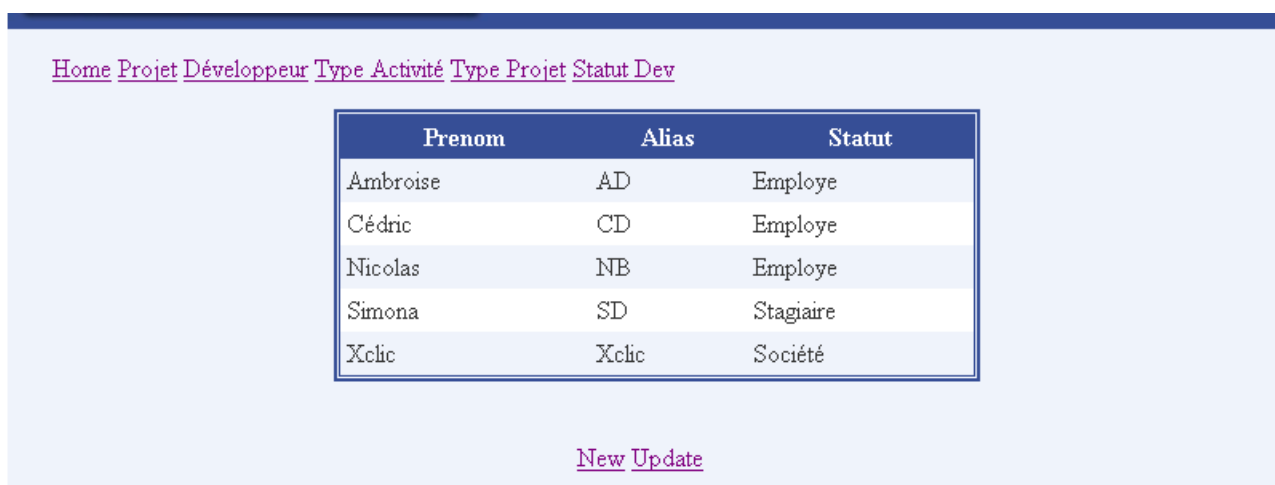
## Développeur

Le module développeur permet :

- la visualisation de la liste des développeurs
- la création des nouveaux comptes des développeurs
- modification/suppression des comptes des développeurs

### *Visualisation de la liste des développeurs*

En sélectionnant le module Développeur dans le menu, la page va afficher la liste de développeurs en mode visualisation.



Home [Projet](#) [Développeur](#) [Type](#) [Activité](#) [Type](#) [Projet](#) [Statut](#) [Dev](#)

Prenom	Alias	Statut
Ambroise	AD	Employe
Cédric	CD	Employe
Nicolas	NB	Employe
Simona	SD	Stagiaire
Xclie	Xclie	Société

[New](#) [Update](#)

Deux boutons sont affichées sur la page : un pour accéder au formulaire de création de nouveaux comptes et l'autre pour accéder à la liste de comptes en mode « update ».

Dans toutes les modules de l'application pour l'affichage des données en mode visualisation et modification j'ai utilisé des contrôles de type « GridView » qui permet de gérer facilement la suppression et la modification de données.

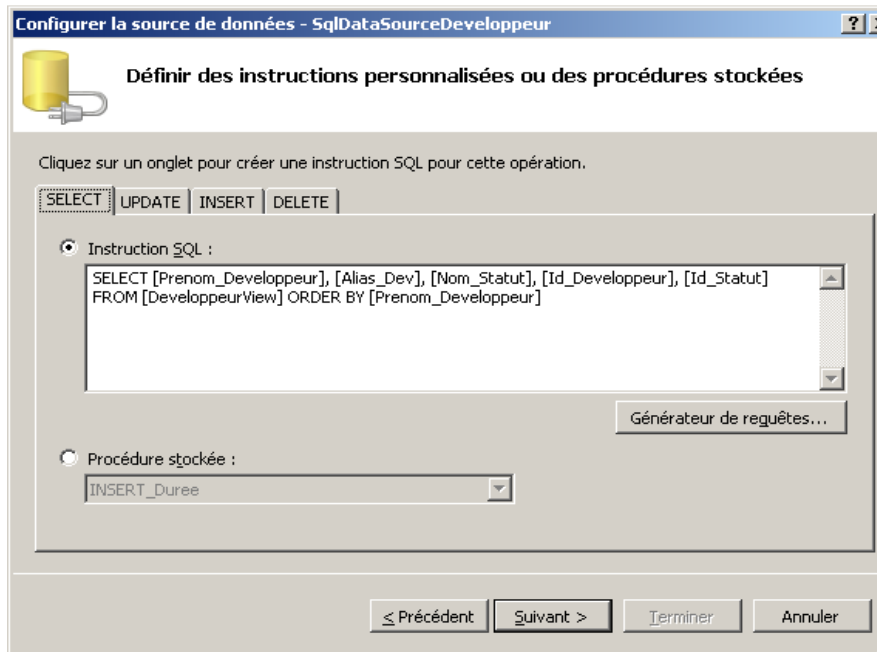
Pour peuplée le GridView avec la liste de données souhaité, j'ai choisi la source de données que j'ai ensuite configuré.

Pour configurer la source de données il faut :

- Choisir la chaine de connexion
- Configurer l'instruction SELECT

Pour configurer l'instruction select, on peut spécifier une instruction personnalisée ou une procédure stockée, ou choisir les colonnes d'une table ou d'une vue.

Dans la source de données on peut aussi créer des requêtes INSERT, DELETE et UPDATE.



### *Modification/suppression des comptes des développeurs*

[Home](#) [Projet](#) [Développeur](#) [Type](#) [Activité](#) [Type](#) [Projet](#) [Statut](#) [Dev](#)

	Prenom	Alias	Statut	
<a href="#">Update</a> <a href="#">Cancel</a>	Ambroise	AD	Employe	
<a href="#">Edit</a>	Cédric	CD	Employe	<a href="#">Delete</a>
<a href="#">Edit</a>	Nicolas	NB	Employe	<a href="#">Delete</a>
<a href="#">Edit</a>	Simona	SD	Stagiaire	<a href="#">Delete</a>
<a href="#">Edit</a>	Xcllic	Xcllic	Société	<a href="#">Delete</a>

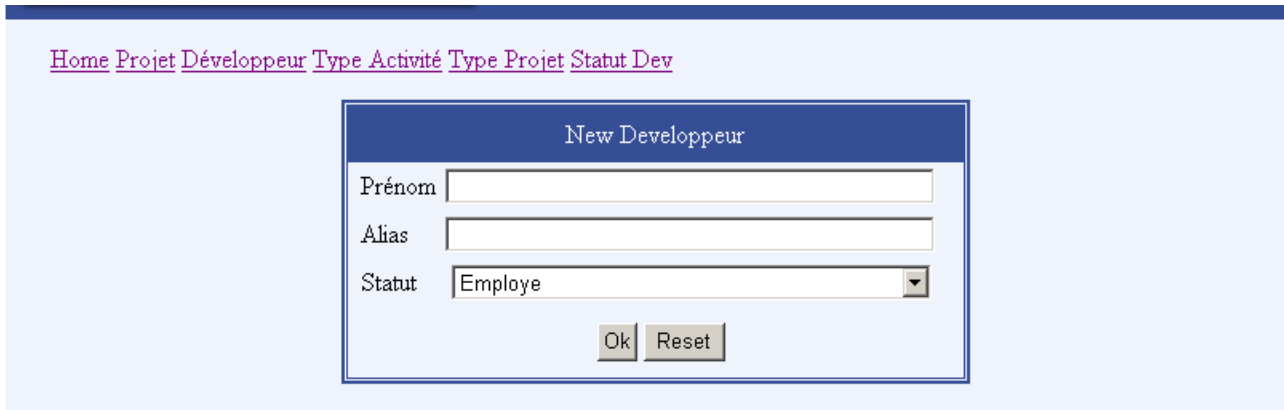
Pour l'affichage en mode « modification/suppression », j'ai ajouté dans le GridView un bouton de type "Edit, Update, Cancel" et un bouton de type "Delete".

Pour la colonne « Statut » en mode « Edit », j'ai remplacé le contrôle de type « textbox » qui est généré automatiquement, par un contrôle de type « DropDownList » que j'ai peuplé avec tous les types de statut enregistrés dans la base de données.



## *Création des nouveaux comptes des développeurs*

Le formulaire de création des nouveaux comptes des développeurs est composé de deux contrôles TextBox, un contrôle DropDownList et deux contrôles Button. Dans le contrôle DropDownList sont affichés les types des projets récupérés de la base de données.



Au click sur le bouton Ok, la procédure ButtonOkDev\_Click se déclenche.

```
Imports System.Data
Imports System.Data.SqlClient
```

```
Partial Class Developpeur_NewDev
    Inherits System.Web.UI.Page
    Public myConnection As SqlConnection
    Public myCommand As SqlCommand
    Dim requete As String

    Protected Sub ButtonOkDev_Click(ByVal sender As Object, ByVal e As System.EventArgs) Handles ButtonOkDev.Click
        myConnection = New SqlConnection(ConfigurationManager.ConnectionStrings("TableauBordConnectionString2").ConnectionString.ToString())
        myConnection.Open()
        requete = "INSERT INTO Developpeur (Prenom_Developpeur, Alias_Dev, Id_Statut)" _
            & "VALUES ('" & TB_Prenom.Text & "', '" & TB_Alias.Text & "', '" & DropDownListStatut.SelectedItem.Value & "')"
        myCommand = New Data.SqlClient.SqlCommand(requete, myConnection)
        myCommand.ExecuteNonQuery()
        LB_Message.Text = "Developpeur " & TB_Prenom.Text & " a été créé !"
        LB_Message.Visible = True

        myConnection.Close()
    End Sub
```

## Projets

### *Modification/suppression des projets*

[Home](#)
[Projet](#)
[Développeur](#)
[Type Activité](#)
[Type Projet](#)
[Statut Dev](#)

Détails	Projet	Type Projet	
<a href="#">Détails</a>	LCDJ	Développement	<a href="#">Delete</a>
<a href="#">Détails</a>	Mawari V2	Développement	<a href="#">Delete</a>
<a href="#">Détails</a>	Biblio	Développement	<a href="#">Delete</a>
<a href="#">Détails</a>	Dolibarr	Développement	<a href="#">Delete</a>
<a href="#">Détails</a>	Gestion Recrutement	Développement	<a href="#">Delete</a>
<a href="#">Détails</a>	Tableau de bord VIPC	Développement	<a href="#">Delete</a>
<a href="#">Détails</a>	Site Web Dev	Développement	<a href="#">Delete</a>
<a href="#">Détails</a>	Atrium Data - Event	Développement	<a href="#">Delete</a>
<a href="#">Détails</a>	Djibouti - Web - Lot 1	Développement	<a href="#">Delete</a>
<a href="#">Détails</a>	Atrium Data	Production	<a href="#">Delete</a>
<a href="#">Détails</a>	Mawari V1	Production	<a href="#">Delete</a>
<a href="#">Détails</a>	Fidelis	Production	<a href="#">Delete</a>
<a href="#">Détails</a>	Indigo-gs	Production	<a href="#">Delete</a>
<a href="#">Détails</a>	VIPC	Production	<a href="#">Delete</a>
<a href="#">Détails</a>	Gestion contact (comite des forêts)	Production	<a href="#">Delete</a>
<a href="#">Détails</a>	Formation DotNet/SQL	Organisation	<a href="#">Delete</a>

Dans la partie de mise à jour des projets, l'utilisateur peut supprimer les projets ou sélectionner le bouton « Détails » pour afficher les projets en détails.

La colonne Détails est une colonne de type HyperLinkField qui utilise la liaison de données (databinding) pour passer un paramètre dans l'url du lien.

Le code source de la collection <Columns> du contrôle <asp:GridView> :

```

<Columns>
  <asp:HyperLinkField DataNavigateUrlFields="Id_Projet" DataNavigateUrlFormatString="UpdateUnProjet.aspx?ID={0}"
    HeaderText="D&#233;tails" Text="D&#233;tails" />
  <asp:BoundField DataField="Nom_Projet" HeaderText="Projet" SortExpression="Nom_Projet" />
  <asp:BoundField DataField="Nom_Type_Projet" HeaderText="Type Projet" SortExpression="Nom_Type_Projet" />
  <asp:BoundField DataField="Id_Projet" HeaderText="Id_Projet" ReadOnly="True" SortExpression="Id_Projet"
    Visible="False" />
  <asp:CommandField ShowDeleteButton="True" >
    <ItemStyle ForeColor="#C00000" />
  </asp:CommandField>
</Columns>

```

**Détails du projet**

C'est dans la partie de détails du projet que l'utilisateur peut mettre à jour toutes les informations qui concernent un projet :

- ajouter des nouveaux développeurs sur un projet
- supprimer les développeurs qui travaillent sur un projet
- mettre à jour les durées prévisionnelles et réelles pour chaque développeur en fonction de chaque activité

**Mawari V2**

Projet		Client	Type Projet
<a href="#">Update</a> <a href="#">Cancel</a>	Mawari V2	CIFEG	Développement

Prenom\_Developpeur:

[Insérer](#) [Annuler](#)

		Developpeur	CommercialPrev	CommercialReel	RecherchePrev	RechercheReel	AnalysePrev	AnalyseReel	ConceptionPrev	ConceptionReel	RealisationP
<a href="#">Update</a> <a href="#">Cancel</a>	<a href="#">Cédric</a>		<input type="text" value="0"/>	<input type="text" value="0"/>	<input type="text" value="0"/>	<input type="text" value="0"/>	<input type="text" value="0"/>	<input type="text" value="0"/>	<input type="text" value="0"/>	<input type="text" value="0"/>	<input type="text" value="0"/>
<a href="#">Edit</a>	<a href="#">Delete</a>	Nicolas	0	0	0	0	0	0	0	0	0
<a href="#">Edit</a>	<a href="#">Delete</a>	Simona	0	0	0	0	3	2	5	4	5

**Home**

La page « Home » affiche :

- tous les projets et leur type
- les développeurs qui travaillent sur chaque projet
- les durées prévisionnelles et réelles en fonction de projet, de développeur et de type d'activité

[Home](#) [Projet](#) [Développeur](#) [Type](#) [Activité](#) [Type](#) [Projet](#) [Statut](#) [Dev](#)

Type	Client/Interne	Developpeur	CommercialPrev	CommercialReel	RecherchePrev	RechercheReel	AnalysePrev	AnalyseReel	ConceptionPrev	ConceptionRe
Développement	Atrium Data - Event	NB	0	0	0	0	0	0	0	0
Développement	Biblio	CD	0	0	5	0	3	0	0	0
Développement	Biblio	NB	0	0	15	8,5	10	0	10	0
Développement	Djibouti - Web - Lot 1	NB	0	0	0	0	0	0	4	2
Développement	Dolibarr	CD	0	0	1	1	2	1	2	1
Développement	Gestion Recrutement	CD	0	0	0	0	0	0	0	0
Développement	LCDJ	CD	0	1	1	1	2	0	0	0
Développement	LCDJ	Xclie	0	0	0	0	0	0	3	1

## II.2 SQL

La procédure stockée « delete\_projet » supprime les données associées aux clefs étrangères dans les tables « Dev\_Projet » et « Projet » pour un projet donné. L'utilisation de cette procédure peut être remplacée par un changement dans le schéma de base de données (l'utilisation d'une contrainte ON DELETE CASCADE).

```
USE [TableauBord]
GO
/***** Object: StoredProcedure [dbo].[pd_projet] *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
|
CREATE PROCEDURE [dbo].[delete_projet]
(
    @Id_Projet int
)
AS
BEGIN

DELETE FROM Dev_Projet WHERE Dev_Projet.Id_Projet = @Id_Projet
DELETE FROM Projet WHERE Id_Projet = @Id_Projet

END
```

Pour créer le rapport des durées prévisionnelles et réelles en fonction de projet, de développeur et de type d'activité j'ai créé des vues (tables virtuelles)

Column	Alias	Table	Output	Sort Type	Sort Order	Group By	Filter	Or...	Or...
Nom_Type_Pr...		Type_Projet	<input checked="" type="checkbox"/>			Group By			
Nom_Projet		Projet	<input checked="" type="checkbox"/>			Group By			

```
SELECT dbo.Type_Projet.Nom_Type_Projet, dbo.Projet.Nom_Projet, dbo.Developpeur.Alias_Dev, dbo.Type_Activite.Nom_Activite, dbo.Dev_Projet.Duree_Prev,
dbo.Dev_Projet.Duree_Reel, dbo.Developpeur.Prenom_Developpeur, dbo.Projet.Id_Projet, dbo.Developpeur.Id_Developpeur
FROM dbo.Type_Projet INNER JOIN
dbo.Projet ON dbo.Type_Projet.Id_Type_Projet = dbo.Projet.Id_Type_Projet RIGHT OUTER JOIN
dbo.Dev_Projet LEFT OUTER JOIN
dbo.Developpeur ON dbo.Dev_Projet.Id_Developpeur = dbo.Developpeur.Id_Developpeur INNER JOIN
dbo.Type_Activite ON dbo.Dev_Projet.Id_Type_Activite = dbo.Type_Activite.Id_Type_Activite ON dbo.Projet.Id_Projet = dbo.Dev_Projet.Id_Projet
GROUP BY dbo.Type_Projet.Nom_Type_Projet, dbo.Projet.Nom_Projet, dbo.Developpeur.Alias_Dev, dbo.Type_Activite.Nom_Activite,
dbo.Dev_Projet.Duree_Prev, dbo.Dev_Projet.Duree_Reel, dbo.Developpeur.Prenom_Developpeur, dbo.Projet.Id_Projet,
dbo.Developpeur.Id_Developpeur
```

Column	Alias	Table	Output	Sort Type	Sort Order	Group By	Filter	Or...
Nom_Type_Pr...		View_1	<input checked="" type="checkbox"/>			Group By		

```

SELECT DISTINCT
    Nom_Type_Projet, Nom_Projet, Alias_Dev,
    CASE WHEN [Nom_Activite] = 'Commercial' THEN MAX(Duree_Prev) ELSE 0 END AS CommercialPrev,
    CASE WHEN [Nom_Activite] = 'Commercial' THEN MAX(Duree_Reel) ELSE 0 END AS CommercialReel,
    CASE WHEN [Nom_Activite] = 'Recherche' THEN MAX(Duree_Prev) ELSE 0 END AS RecherchePrev,
    CASE WHEN [Nom_Activite] = 'Recherche' THEN MAX(Duree_Reel) ELSE 0 END AS RechercheReel,
    CASE WHEN [Nom_Activite] = 'Analyse' THEN MAX(Duree_Prev) ELSE 0 END AS AnalysePrev,
    CASE WHEN [Nom_Activite] = 'Analyse' THEN MAX(Duree_Reel) ELSE 0 END AS AnalyseReel,
    CASE WHEN [Nom_Activite] = 'Conception' THEN MAX(Duree_Prev) ELSE 0 END AS ConceptionPrev,
    CASE WHEN [Nom_Activite] = 'Conception' THEN MAX(Duree_Reel) ELSE 0 END AS ConceptionReel,
    CASE WHEN [Nom_Activite] = 'Réalisation' THEN MAX(Duree_Prev) ELSE 0 END AS RealisationPrev,
    CASE WHEN [Nom_Activite] = 'Réalisation' THEN MAX(Duree_Reel) ELSE 0 END AS RealisationReel,
    CASE WHEN [Nom_Activite] = 'Test' THEN MAX(Duree_Prev) ELSE 0 END AS TestPrev,
    CASE WHEN [Nom_Activite] = 'Test' THEN MAX(Duree_Reel) ELSE 0 END AS TestReel,
    CASE WHEN [Nom_Activite] = 'Validation' THEN MAX(Duree_Prev) ELSE 0 END AS ValidationPrev,
    CASE WHEN [Nom_Activite] = 'Validation' THEN MAX(Duree_Reel) ELSE 0 END AS ValidationReel,
    CASE WHEN [Nom_Activite] = 'Recette' THEN MAX(Duree_Prev) ELSE 0 END AS RecettePrev,
    CASE WHEN [Nom_Activite] = 'Recette' THEN MAX(Duree_Reel) ELSE 0 END AS RecetteReel,
    CASE WHEN [Nom_Activite] = 'Support' THEN MAX(Duree_Prev) ELSE 0 END AS SupportPrev,
    CASE WHEN [Nom_Activite] = 'Support' THEN MAX(Duree_Reel) ELSE 0 END AS SupportReel, Prenom_Developpeur, Id_Projet, Id_Developpeur
FROM    dbo.View_1
GROUP BY Nom_Type_Projet, Nom_Projet, Alias_Dev, Nom_Activite, Prenom_Developpeur, Id_Projet, Id_Developpeur
    
```

## CONCLUSION

Ce stage m'a permis de mettre en pratique les compétences acquises au cours de la formation à l'AFPA.

Il m'a permis de me familiariser avec l'outil de développement Visual Web Developer et la technologie ASP.NET.

Ce stage est une expérience enrichissante car il m'a fait progresser autant dans le domaine technique que relationnel.

## REMERCIEMENTS

Je tiens à remercier Indigo Global Services  
de m'avoir permis d'effectuer mon stage de formation professionnelle.

Merci à

Monsieur Stéphane MORIN  
Directeur d'Indigo Global Services  
de m'avoir accueilli dans son établissement

Monsieur Cédric DUBIE  
Chef du pôle développement  
de son encadrement et de sa disponibilité durant cette période

Merci à toute l'équipe de développement :  
Monsieur Ambroise DUMONT et Monsieur Nicolas BONNOT  
de m'avoir guidé tout au long de mon stage

Merci à l'ensemble du personnel de son accueil