

BARTHELET Benjamin
DE LOITIERE Sophie
MAUGER Ivan
SINQUIN Ryan



Mise en ligne des syllabus de cours et des CV des enseignants de l'ISTM

Tuteurs : Alain Lacombe
Iman Peymaneh

TABLE DES MATIERES

I. INTRODUCTION.....	2
II. CAHIER DES CHARGES.....	3
1. Analyse de l'existant	3
2. Analyse des besoins utilisateurs	3
3. Analyse fonctionnelle	4
4. Choix techniques.....	5
5. Planning et charges prévisionnels	5
III. DICTIONNAIRE DE DONNEES	6
1. Table PROF.....	6
2. Table MATIERE.....	6
3. Table AXE	7
4. Table FILIERE	7
5. Table CENTRE_INTERET	7
6. Table BIBLIO	7
7. Table TYPE_ BIBLIO.....	8
8. Table RECHERCHE	8
9. Table INDUS_CONSEIL.....	8
10. Table DROIT_ACCES	8
IV. ORGANISATION DE LA BASE.....	9
V. PROBLEMES RENCONTRES ET SOLUTIONS.....	11
1. Problèmes matériels	11
2. Problèmes logiciels	11
VI. PROGRAMMES IMPLEMENTES	12
1. Arborescence des fichiers	12
2. Liens existants entre les pages.....	12
VII. ETAT DU PROJET.....	13
VIII. CONCLUSION	14
IX. ANNEXES	15
1. Planning effectif	15
2. Notice d'administration.....	16
3. Glossaire.....	17
4. Page PHP : index.php.....	19

I. INTRODUCTION

Dans le cadre d'une amélioration de l'image et de la promotion de l'établissement auprès du public, l'ISTM souhaite placer sur son site web une description du syllabus des enseignements dispensés aux étudiants ainsi que les Curriculum Vitae des enseignants permanents.

Il s'agit de développer de nouvelles rubriques d'information au sein du site existant en vue d'offrir aux visiteurs des possibilités d'accès multiples. Il s'agit également de permettre la saisie des informations correspondantes grâce à des formulaires et depuis un client Web, à chacune des catégories d'utilisateurs responsables et chargés de la mise à jour des contenus. Ces fonctionnalités seront mises en œuvre grâce à la mise en place de pages dynamiques.

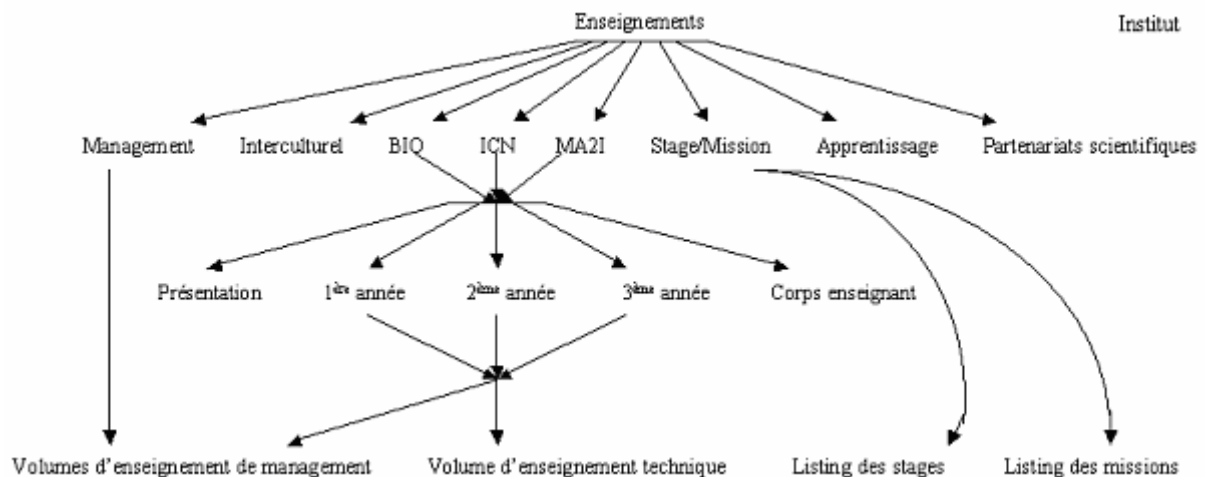
Cette tâche nous a donc été confiée dans le cadre du projet de synthèse de deuxième année, avec un effectif de quatre personnes et cinquante heures de travail réparties sur deux mois. Afin de nous aider à mener ce projet à son terme, nous avons bénéficié de l'aide de nos tuteurs, Alain Lacombe et Peymaneh Iman. Nous nous sommes également inspirés des exemples de la présentation souhaitée rédigés par Jean-Marc Pointet.

Nous allons tout d'abord rédiger le cahier des charges, mettre en place un dictionnaire de données afin d'organiser la future base, et consigner toutes ces informations dans ce rapport. Enfin, nous allons programmer les requêtes et les formulaires d'administration.

II. CAHIER DES CHARGES

1. Analyse de l'existant

Sur le site de l'ISTM, l'onglet "Enseignement" contient les informations suivantes :

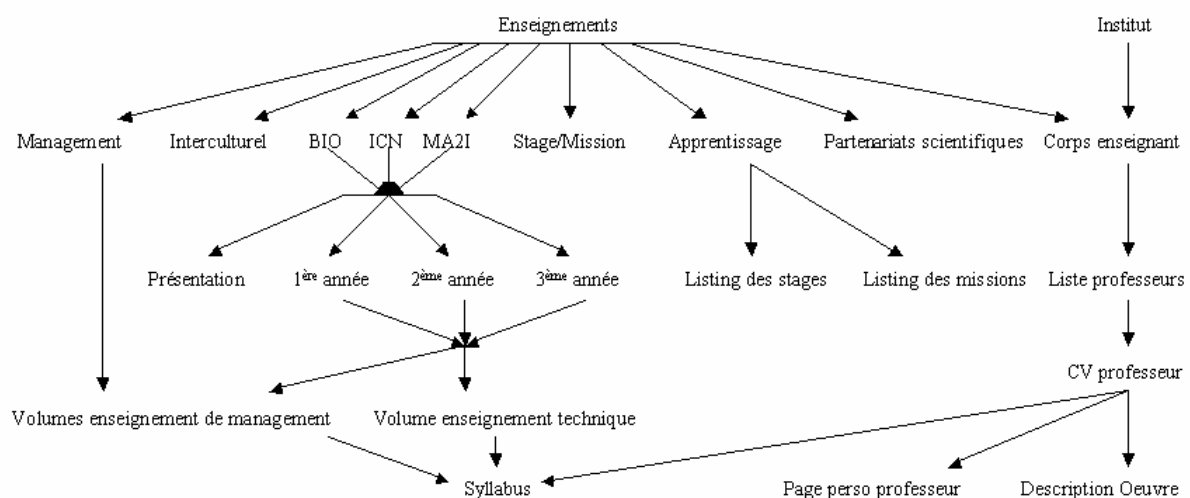


2. Analyse des besoins utilisateurs

2.1 Objectifs

Le projet qui nous a été confié consiste en la mise en ligne des **curriculum vitae des enseignants** permanents et de la description des **syllabus des enseignements**. Il s'agit de développer dynamiquement de nouvelles rubriques d'information au sein du site existant, pour offrir des possibilités d'accès multiples aux visiteurs.

Nouvelle architecture de l'onglet "Enseignements" :



2.2 Besoins initiaux

Afin de répondre correctement aux exigences de ce projet, nous devons tout d'abord définir les besoins des utilisateurs et effectuer l'analyse fonctionnelle. Après avoir effectué les choix techniques qui s'imposent, nous installerons et paramètrons l'environnement technique (Base de données), et nous pourrons alors concevoir les pages et les procédures pour l'affichage des données en dynamique sur la page de l'ISTM. Enfin, nous réaliserons et testerons l'application développée.

Pour définir les besoins utilisateurs, nous devons questionner les différents types de personnes concernées (étudiants, enseignants, professionnels...), et nous mettre nous-même en situation. Ainsi, en visitant le site www.istm.fr, nous avons pu définir les besoins suivants en ce qui concerne l'onglet "Enseignements" :

L'utilisateur doit pouvoir accéder directement à une liste des enseignants permanents.

Les enseignants doivent être répartis par spécialités (management, interculturel, BIO, ICN, MA2I).

A chaque enseignant doit correspondre un lien vers les informations qui lui sont propres (Descriptif, CV etc...).

L'utilisateur doit pouvoir accéder aux descriptifs des enseignements dispensés, par filières, puis par année.

Toutes les informations doivent pouvoir être imprimées très lisiblement par l'utilisateur ("calque" prédéfini pour l'impression).

3. Analyse fonctionnelle

3.1 Fonctionnalités

- Pour les professeurs :

Création, modification, consultation et suppression de leur CV et des syllabus les concernant.

Création d'un formulaire de saisie pour la création des CV et des syllabus, accessible après identification par login et mot de passe.

- Pour l'internaute :

Création d'un accès à la liste de tous les professeurs permanents en management, en langues, et par filière (ICN, BIO et MA2I).

Création d'un accès au CV d'un professeur grâce à un clic sur son nom.

Création d'un accès à la liste des enseignements par année et filière.

Création d'un accès au syllabus de chaque matière.

Intégration des pages créées dans la charte graphique existante.

3.2 Droits d'accès

- Les internautes : droits de consultation uniquement.
- Les professeurs : tous les droits sur leurs propres données.
- Les professeurs responsables de filières : tous les droits sur leurs propres données, et tous les droits sur les matières concernant leur filière.

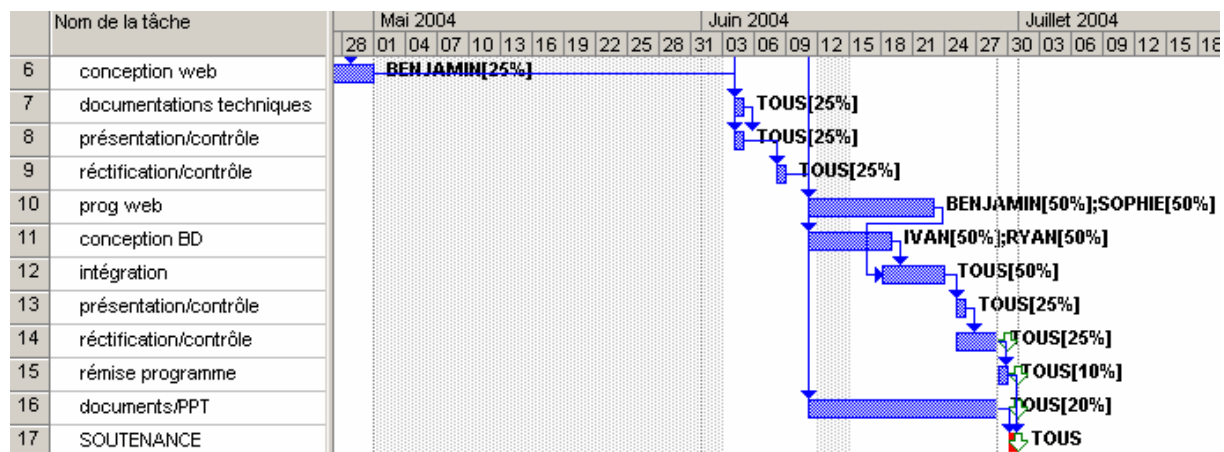
PROFESSEURS PERMANENTS	LOGIN	select	insert/update/delete
Agnès DEBBASCH	debbasca	all	lui-même et matières management
Alain LACOMBE	lacombea	all	all
Bernard KAHANE	kahaneb	all	lui-même
Bertrand ROUDIER	roudierb	all	bio
Derek MAINWARING	mainwade	all	langue
Eric LAHILLE	lahillee	all	lui-même et matières management
Hassane MIMOUN	mimounh	all	icn
Iman PEYMANEH	imape	all	all
Jean HACHE	hachej	all	bio
Jean-Marc POINTET	pointetj	all	lui-même
Michel DALIPAGIC	dalipagim	all	ma2i
Public	pas de login	all	
Sophie de LAFOND	delafons	all	lui-même et matières management
Souad LATORRE	latorres	all	ma2i

4. Choix techniques

- Serveur : Mandrake 9.1 pour les tests / Debian pour la production.
- Langages : php, posgresSql ? JavaScript.
- Base de données : PostgreSQL (outil installé sur le serveur).

5. Planning et charges prévisionnels

En demi-journée, sauf la dernière semaine.



III. DICTIONNAIRE DE DONNEES

Dans la base de données, nous avons créé les dix tables suivantes :

1. Table PROF

Clé primaire :

UID : chaîne limitée à 8 caractères, correspond au login du professeur

Champs :

PRENOM_PROF : chaîne limitée à 20 caractères

NOM_PROF : chaîne limitée à 20 caractères

DESCR_PROF : chaîne limitée à 1000 caractères (le descriptif du professeur)

MAIL_PROF : chaîne limitée à 30 caractères

TEL_PROF : chaîne limitée à 15 caractères au cas où l'on entre des points entre chiffres

RESP_DE_PROF : chaîne limitée à 200 caractères (soit le titre de responsabilité)

PAGE_PERSO_PROF : chaîne limitée à 100 caractères

Clé étrangère :

ID_FILIERE : entier de 1 à 5 (cf. NOM_ABV_FILIERE)

2. Table MATIERE

Clé primaire :

ID_MATIERE : chaîne limitée à 8 caractères maximum (par exemple G1-DROIT pour la matière « Droit » en 1^{ère} année)

Champs :

NOM_MATIERE : chaîne limitée à 50 caractères

CONTEXTE_MATIERE : chaîne limitée à 400 caractères

OBJECTIFS_MATIERE : chaîne limitée à 200 caractères

CONTENU_MATIERE : chaîne limitée à 3000 caractères

METHODES_MATIERE : chaîne limitée à 200 caractères

SUPPORTS_MATIERE : chaîne limitée à 200 caractères

INTERVENANTS_MATIERE : chaîne limitée à 200 caractères

EVALUATION_MATIERE : chaîne limitée à 100 caractères

NB_H_COURS_MATIERE : entier

NB_H_TD_MATIERE : entier

NB_H_TP_MATIERE : entier

NB_H_ETUDE_CAS_MATIERE : entier

CREDIT_ECTS_MATIERE : entier court

ANNEE_MATIERE : entier court de 1 à 3

Clé étrangère :

ID_AXE : entier de 1 à 23

3. Table AXE

Clé primaire :

ID_AXE : entier de 1 à 23 (car 23 axes)

Champs :

NOM_AXE : chaîne limitée à 60 caractères

Clé étrangère :

ID_FILIERE : entier de 1 à 5 (cf. NOM_ABV_FILIERE)

4. Table FILIERE

Clé primaire :

ID_FILIERE : entier de 1 à 5 (cf. NOM_ABV_FILIERE)

Champs :

NOM_ABV_FILIERE : chaîne limitée à 4 caractères (soit BIO, ICN, MA2I, MGT pour le management ou LANG pour les enseignements culturels)

NOM_FILIERE : chaîne limitée à 60 caractères

5. Table CENTRE INTERET

Clé primaire :

ID_CENTRE_INTERET : entier auto-généré par une séquence SQL

Champs :

NOM_CENTRE_INTERET : chaîne limitée à 50 caractères

6. Table BIBLIO

Clé primaire :

ID_BIBLIO : entier auto-généré par une séquence SQL

Champs :

NOM_BIBLIO : chaîne limitée à 200 caractères

DESCR_BIBLIO : chaîne limitée à 10000 caractères

AUTEUR_BIBLIO : chaîne limitée à 100 caractères

EDITION_BIBLIO : chaîne limitée à 200 caractères

ANNEE_BIBLIO : entier de 9 caractères (par exemple 2004-2005)

Clé étrangère :

ID_TYPE_BIBLIO : entier de 1 à 9

7. Table TYPE BIBLIO

Clé primaire :

ID_TYPE_BIBLIO : entier de 1 à 9 (car 9 types de bibliographie)

Champs :

NOM_TYPE_BIBLIO : chaîne limitée à 100 caractères

8. Table RECHERCHE

Clé primaire :

ID_RECHERCHE : entier auto-généré par une séquence SQL

Champs :

NOM_RECHERCHE : chaîne limitée à 200 caractères

9. Table INDUS CONSEIL

Clé primaire :

ID_INDUS_CONSEIL : entier auto-généré par une séquence SQL

Champs :

NOM_INDUS_CONSEIL : chaîne limitée à 200 caractères

DESCR_INDUS_CONSEIL : chaîne limitée à 5000 caractères (le descriptif)

ANNEE_INDUS_CONSEIL : chaîne de 9 caractères (par exemple 2004-2005)

10. Table DROIT ACCES

Elle n'est pas reliée aux autres tables, mais elle permet d'y accéder. Elle est une table publique.

Clé primaire :

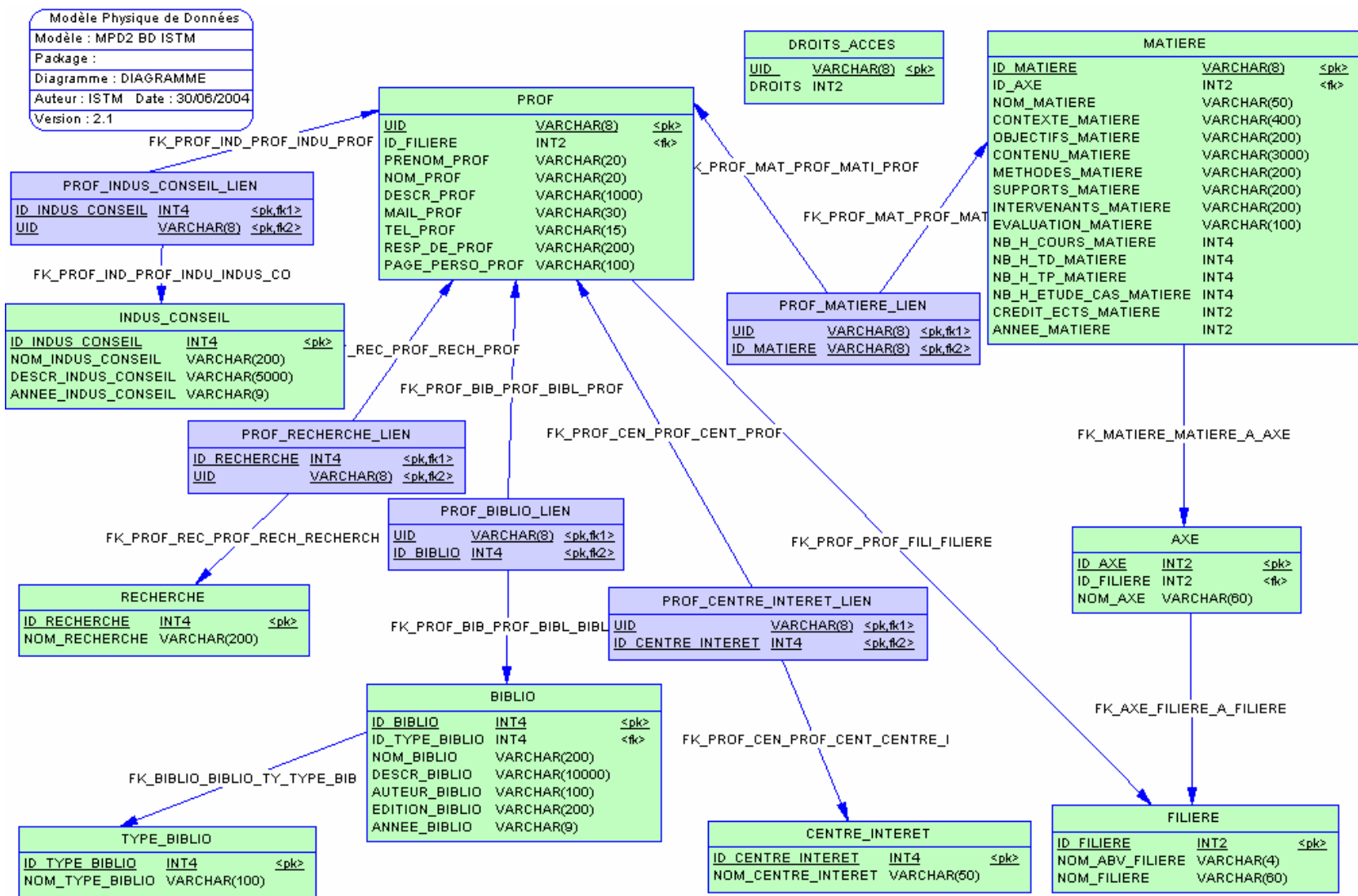
UID : chaîne limitée à 8 caractères, correspond au login du professeur lors de l'authentification.

Champs :

DROITS : entier court qui peut prendre les valeurs suivantes :

- 0 : pour les personnes qui n'ont accès qu'à leur compte, c'est à dire les professeurs.
- de 1 à 5 : pour les personnes qui ont accès à leur compte et aux informations relatives à leur filière, c'est à dire les responsables de filière.
- 99 : pour les personnes qui ont tous les droits, soit les administrateurs Peymaneh et M. Lacombe.

IV. ORGANISATION DE LA BASE



V. PROBLEMES RENCONTRES ET SOLUTIONS

1. Problèmes matériels

- Lecteur CD :

Le lecteur du serveur était défectueux, et nous pouvions donc pas installer les modules nécessaires. Pour remédier à ce problème, nous l'avons changé.

- La carte réseau :

Elle n'était pas reconnue, et après avoir longuement essayé de l'utiliser, nous avons finalement décidé de la changer. Une fois la carte activée, il fallait autoriser les communications à la base de donnée via TCP/IP.

Nous avons donc modifié la configuration de la base :

- dans /var/lib/data/postgresql.conf, il fallait mettre `tcpip_socket=true`
- dans /var/lib/data/pg_hba.conf, il fallait mettre les droits d'entrée pour l'adresse du poste de travail que nous utilisons : `host all 147.215.181.34 255.255.255.0 trust`

- La souris

Pour la navigation dans l'environnement graphique, nous avons besoin d'une souris fonctionnelle. Avec la Debian, nous n'avons pas réussi à installer la souris et avons donc changé de distribution (Mandrake) pour y parvenir.

Ainsi notre serveur de test n'est pas identique au serveur de production, mais les modules que nous utilisons sont invariant.

2. Problèmes logiciels

- Le système d'exploitation :

L'OS (Linux) installé par défaut sur le serveur ne permettait pas d'accéder correctement au réseau, or ceci nous était indispensable pour le travail distant sur le serveur.

Nous avons donc formaté le disque et installé la Debian.

- La base LDAP :

Nous avons besoin de droits sur la base LDAP, et de connaître son fonctionnement. Ainsi, nous avons programmé un entretien avec Monsieur Delahaye.

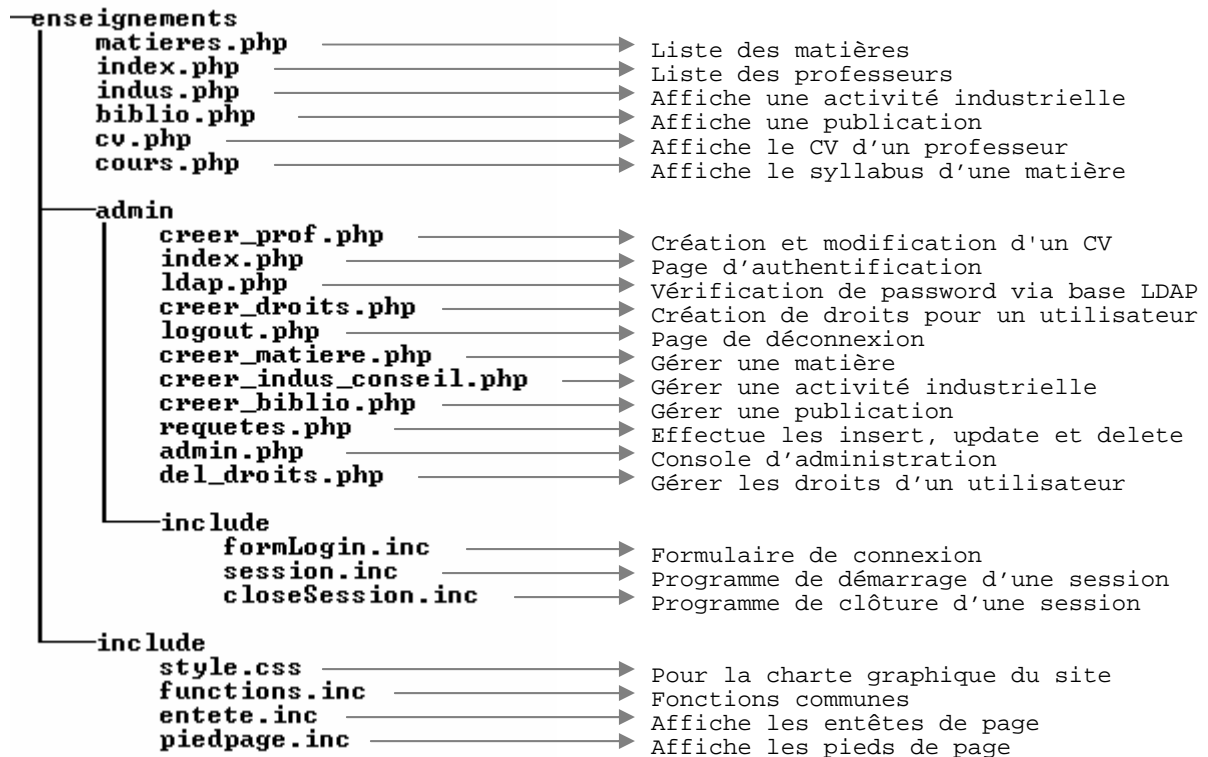
- PHP et sa sécurité

Nous devons sécuriser la partie administration et permettre aux utilisateurs d'accéder à leur propre console d'administration. Pour cela, il a fallu utiliser des sessions en PHP.

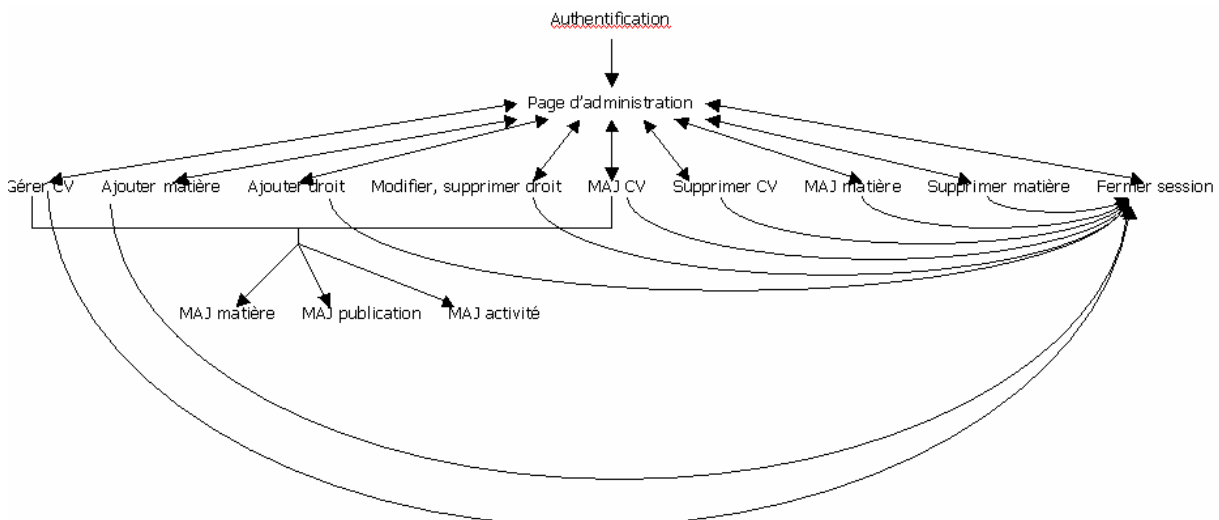
VI. PROGRAMMES IMPLÉMENTÉS

1. Arborescence des fichiers

Chemin d'accès général du site : /var/www/html/



2. Liens existants entre les pages



VII. ETAT DU PROJET

Au 1/07/2004

Il nous a fallu quelques heures après la soutenance pour finir totalement le projet.

Le projet est à présent implémenté et testé (pas totalement) sur le serveur de test :

- site : <http://147.215.181.250>
- administration : <http://147.215.181.250/enseignements/admin>.

Tous les documents rapport et notice ont été fournis, ainsi que les fichiers du projet.

La mise en production est reportée à la rentrée en deux étapes :

- Mise en production de la base et de la partie administration
Remplissage de la base et test.
- Mise en production du site avec les liens dynamiques.

VIII. REMERCIMENTS

Pour leur aide technique :

A. LACOMB

I. PEYMANEH

M. DELAHAYE

Christine et Fabienne,
pour leur bonne humeur quand nous en avons eu besoin.

CONCLUSION

La mise en ligne des curriculum vitae des enseignants permanents et de la description des syllabus des enseignements était un travail intéressant et motivant, car très utile pour l'ISTM.

En effet, les internautes du site de l'institut pourront consulter toutes les informations relatives aux professeurs et aux cours qu'ils enseignent. Cela est très important, car les étudiants souhaitant déposer leur candidature seront ainsi mieux informés, et les professionnels consultant le site pourront alors évaluer plus facilement la qualité des professeurs et des enseignements.

De plus, ce projet a fait la synthèse d'une grande partie de nos compétences organisationnelles et techniques, et nous a permis d'acquérir de nouvelles connaissances. Effectivement, en ce qui concerne le management, ce projet a fait appel à la planification, à l'étude des besoins, au cahier des charges la gestion de projet avec UML, la répartition des tâches, et la communication. Pour ce qui est de la technique, nous avons mis à profit nos compétences dans le domaine des bases de données (SQL, PostgreSQL, PgAdmin, LDAP), des langages (PHP, HTML, Javascript), et des systèmes d'exploitation (Linux).

Nous avons donc appris à gérer les contraintes techniques (langage de développement PHP imposé, configuration du serveur de test compatible avec le serveur de développement), matérielles (un poste de travail avec Apache et un serveur), et temporelles (peu de plages horaires disponibles et séances mal réparties dans le temps).

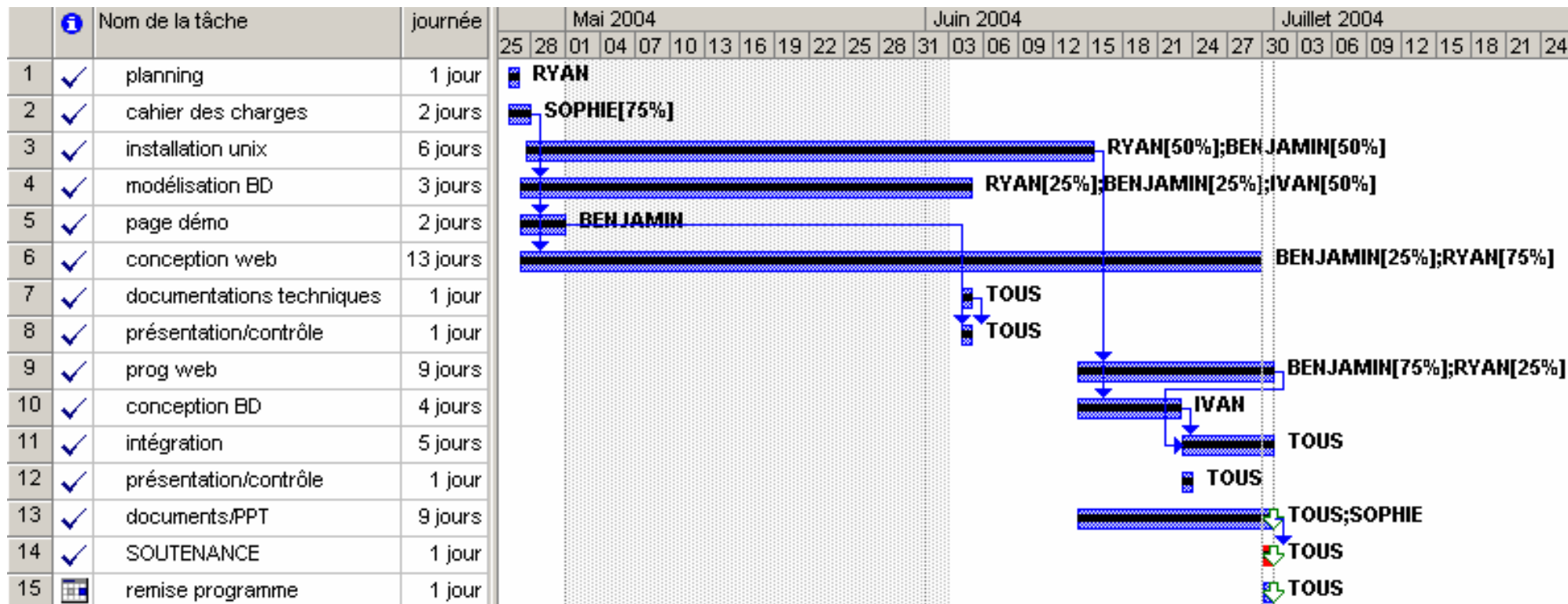
Toutes ces contraintes n'ont pas toujours été faciles à gérer, et nous ont souvent posé des problèmes au niveau du temps de développement et de la prise de recul par rapport au projet. Nous avons dû apprendre à travailler en équipe, à tirer parti des compétences de chacun, et à tenir compte des problèmes liés à l'accomplissement des tâches pour la planification.

Malgré les problèmes que nous avons rencontrés, et qui ont freiné l'avancement du projet, nous nous sommes tous impliqués pour le mener à son terme dans le délai qui nous était imparti. Ce projet nous a permis de mettre en œuvre nos compétences, et nous serons ainsi plus à l'aise en entreprise face aux projets qui nous seront confiés.

IX. ANNEXES

1. Planning effectif

En demi-journée, sauf la dernière semaine.





2. Notice d'administration

Pour les tests avant mise en production, utilisez les liens suivant :

Accès au site : <http://147.215.181.250/>

Accès aux formulaires : <http://147.215.181.250/enseignements/admin>

Sous PG Admin, pour recréer la base :

- il faut créer la base de données "istm" avec l'utilisateur "postgres", et en "SQL_ASCII".

- il faut lancer crebas.sql et crebasDroits.sql

Pour mettre en place l'interface d'administration, il faut copier les fichiers suivant dans le répertoire du site :

```
—enseignements
  matieres.php
  index.php
  indus.php
  biblio.php
  cv.php
  cours.php
  —admin
    creer_prof.php
    index.php
    ldap.php
    creer_droits.php
    logout.php
    creer_matiere.php
    creer_indus_conseil.php
    creer_biblio.php
    requetes.php
    admin.php
    del_droits.php
  —include
    formLogin.inc
    session.inc
    closeSession.inc
  —include
    style.css
    functions.inc
    entete.inc
    piedpage.inc
```

Pour mettre la base en production :

- il faut écraser : /body_index.js, /body.js, /plansite.html,
/institut/plan_institut.html et /enseignements/plan_enseignements.html
par les nouveaux du même nom

- il faut remplacer /enseignements/interculturel.html par
/enseignements/lang.html et /enseignements/management.html par
/enseignements/mgt.html

De temps en temps, dans /tmp, il faut faire rm sess_* (cela sert à
nettoyer les sessions qui n'ont pas été fermées correctement).



3. Glossaire

- Apache

C'est un serveur Web gratuit fonctionnant sous Linux et Windows NT, aujourd'hui le plus utilisé, hébergeant en effet plus d'un million de sites. Il est bien plus stable que le serveur web IIS de Microsoft.

- Debian

C'est la seule distribution Linux entièrement développée de manière coopérative par des bénévoles et sans but lucratif.

- HTML (HyperText Markup Language)

C'est un langage permettant au navigateur de reconnaître une page Internet ou Intranet, et de l'afficher en conséquence. Un document HTML n'est donc qu'un fichier texte.

Pour que le navigateur puisse interpréter la structure du document, on utilise des balises pour préciser un titre, sous titre, liste, une mise en forme de texte etc...

- Javascript

Langage de développement assez proche de Java, utilisé dans la conception de pages Web. Le JavaScript tourne dans la majorité des cas aussi bien sur les navigateurs de Netscape et de Microsoft.

- LDAP (Lightweight Directory Access Protocol)

Protocole utilisé pour communiquer à partir d'un programme appelant (exécuté sur un nœud comme un serveur de commerce) et d'un nœud annuaire. Les informations relatives aux utilisateurs et/ou aux services sont stockées sur le nœud annuaire LDAP.

Par exemple, l'annuaire peut être utilisé pour stocker les informations requises pour identifier les clients enregistrés (procédure d'authentification). Il peut être également utilisé pour stocker les informations à propos des fonctions que les clients peuvent effectuer une fois identifiés (procédure d'autorisation).

- Linux

C'est une version d'Unix gratuite et librement diffusable. Le kernel, c'est à dire le noyau, a pu se développer par l'intermédiaire d'Internet et des passionnés de programmation. L'avantage est qu'il est aisé de trouver des mises à jour.



- Mandrake

Distribution de Linux d'origine française par Mandrakesoft. Mandrake a associé un environnement de travail graphique à Linux, permettant de simplifier son utilisation.

- Merise

Méthode française de modélisation des processus de gestion. Créée sous l'impulsion du Ministère de la Recherche et de l'Industrie, elle a largement dominé le secteur de la modélisation de systèmes d'information durant les années 80 en France, et ce jusqu'à l'émergence des méthodes orientées objet, qui l'ont peu à peu supplantée durant les années 90. Depuis 2000 environ, elle cède la place aux méthodes basées sur la notation UML.

- OS (Operating System)

Programme assurant la gestion de l'ordinateur et de ses périphériques d'entrée, de sortie, et d'entrée-sortie. Les principaux systèmes d'exploitation sont Windows, Unix, Linux, Mac OS, Be OS et MS Dos.

- PgAdmin

C'est une plate-forme d'administration et de développement pour la base de données PostgreSQL. Elle a été conçue pour répondre aux besoins de tous les utilisateurs, depuis l'écriture de requêtes simples jusqu'au développement de bases de données complexes. L'interface graphique donne accès aux fonctionnalités de PostgreSQL les plus récentes, faisant de l'administration un véritable jeu d'enfant. L'application comprend également un constructeur de requêtes, un éditeur SQL etc...

- PHP (Hypertext Preprocessor)

Langage permettant la création de pages Web au contenu dynamique, analogue à la technologie ASP de Microsoft, mais provenant des environnements Unix-Apache et libre de droits.

- Postgres

Système de gestion de base de données relationnelle open source caractérisé en interne par une gestion de version multiples simultanée des enregistrements. Il dérive de Postgres95, version de Postgres modifiée pour ajouter le support de SQL.

- SQL (Structured Query Language)

Langage structuré de requête que l'on réserve à l'interrogation de bases de données importantes.



4. Page PHP : index.php

Exemple d'un des programmes d'affichage :

```
<?php
/*****
* Nom du programme : index.php
* Page d'accueil de la partie enseignements
* Date de génération du programme : mercredi 30 juin 2004
* Copyright : ISTM - Tous droits réservés.
*****/

/-- Titre de la page --
$titre = "Corps enseignant";
/-- Choix de l'image en haut de la page
$rubrique = "titres-enseignement.gif";
/-- Appel du fichier des fonctions de routines --
include("include/fonctions.inc");
/-- Appel de l'entete du fichier --
include("include/entete.inc");

/-----
/-- Construction de la page --
/-----

/** Tableau des filières **
$sql = "SELECT nom_filiere, id_filiere FROM filiere";
/-- Exécution de la requete --
dbconnect($sql);
$i = 0;
while ($ligne = pg_fetch_row($GLOBALS['resultat'])) {
    $sousTitre[$i][0] = $ligne[0];
    $sousTitre[$i][1] = $ligne[1];
    $i++;
}

/-- On construit la ligne de titre avec un rowspan du nombre de lignes --
$ligneTitre = constTitre($titre, (count($sousTitre)*2)+1);
/-- On construit une ligne de tableau <TR> --
$tableau = ligneTab($ligneTitre);
/-- On parcourt le tableau des filières et on construit les lignes --
for ($i = 0; $i < count($sousTitre); $i++){
    $ligneSousTitre = constSousTitre($sousTitre[$i][0],
    $sousTitre[$i][1]);
    $listeProfs = "";
    $sql = "SELECT uid, prenom_prof, nom_prof, resp_de_prof FROM prof
WHERE id_filiere='". $sousTitre[$i][1]. "'";
    dbconnect($sql);
    /** Sélection des profs de chaque filière **
    while ($ligne = pg_fetch_row($GLOBALS['resultat'])) {
        $sql = "SELECT nom_matiere FROM matiere WHERE
matiere.id_matiere = prof_matiere_lien.id_matiere AND prof.uid =
prof_matiere_lien.uid AND prof.uid = '". $ligne[0]. "'";
        $connexion = pg_connect ("host=localhost port=5432 dbname=istm
user=postgres") or die("Impossible de se connecter");
        $result = pg_query ($connexion, $sql);
        if (!$result) {echo "Une erreur est survenue.\n";}
        $mat = "";
        $listMat = "";
        while ($ligneMat = pg_fetch_row($result)) {
```



```
        $mat.= $ligneMat[0].", ";
    }
    $mat = substr($mat, 0, count($mat)-3);
    if ($mat != "") $listMat = " - Professeur de ";
    $listMat.= "<I>".$mat."</I>";
    if ($ligne[3] != "") $ligne[3] = " - ".$ligne[3];
    //-- Construction de la ligne des liens de profs --
    $listeProfs.= prof($ligne[1]." ".$ligne[2], $listMat,
$ligne[3],$ligne[0]);
    }
    //-- Construction du tableau final <TABLE> --
    $ligneListeProfs = constLiens($listeProfs);
    $tableau.= ligneTab($ligneSousTitre).ligneTab($ligneListeProfs);
}
//-- Enfin on affiche le tableau complet final --
echo tab($tableau);

//-- Appel du fichier de pied de page --
include("include/piedpage.inc");
?>
```

Exemple d'un des programmes outils :

```
<?php
/*****
* Nom du programme : fonctions.inc
* Date de génération du programme : dimanche 16 mai 2004
* Copyright : ISTM - Tous droits réservés.
*****/
global $racine;
$racine = "";

global $dynamique;
$dynamique = "enseignements";

//$_SERVER['DOCUMENT_ROOT'].
//-----
//-- Connexion à la base --
//-----
function dbconnect($sql){
    global $resultat;
    $connexion = pg_connect ("host=localhost port=5432 dbname=istm
user=postgres") or die("Impossible de se connecter");
    $resultat = pg_query ($connexion, $sql);
    if (!$resultat) {echo "Une erreur est survenue.\n";}
}
function dbclose(){
    pg_close ($connexion);
}

//-----
//-- Construction d'un tableau --
//-----
function tab($x){
    $txt = '
<TABLE BORDER="0" CELLSPACING="3" CELLPADDING="0" WIDTH="100%">
';
    $txt.= $x;
    $txt.= '
';
}
```

```
        </TABLE>';
        return $txt;
    }
function tab2($x){
    $txt = '
    <TABLE BORDER="0" CELLSPACING="3" CELLPADDING="0">
    ';
    $txt.= $x;
    $txt.= '
    </TABLE>';
    return $txt;
}
function ligneTab($x){
    $txt = '
    <TR>
    ';
    $txt.= $x;
    $txt.= '
    </TR>';
    return $txt;
}
function cellTab($x, $args){
    $txt = '
        <TD '.$args.'>
    ';
    $txt.= $x;
    $txt.= '
        </TD>';
    return $txt;
}

//-----
//-- Construit une ligne de deux cases --
//-----
function ligneDoubleCell($texte1,$texte2){
    if ($texte1 == '') $texte1 = "&nbsp;";
    if ($texte2 == '') $texte2 = "&nbsp;";
    return ligneTab(doubleCell('<span
class="titreb">'.$texte1.'</span>', "align='right'", $texte2, ""));
}

//-----
//-- Construit deux cases dans une ligne --
//-----
function doubleCell($texte1,$args1,$texte2,$args2){
    if ($texte1 == '') $texte1 = "&nbsp;";
    if ($texte2 == '') $texte2 = "&nbsp;";
    $colonne1 = cellTab($texte1,$args1);
    $colonne2 = cellTab($texte2,$args2);
    return $colonne1.$colonne2;
}

//-----
//-- Construction ligne de titre --
//-----
function constTitre($titre, $taille){
    return doubleCell('', 'WIDTH="6" ROWSPAN="'.$taille.'"', '<IMG
SRC="'.$GLOBALS['racine'].'/images/puce.gif" WIDTH="15" HEIGHT="15"
align="middle" HSPACE="5"><SPAN
CLASS="titreb">'.$titre.'</SPAN><BR>', 'COLSPAN="2"');
}

//-----
//-- Construction ligne de sous titre --
```

```
//-----  
function constSousTitre($sousTitre, $filiere){  
    switch ($filiere){  
        case 1:  
        case 2:  
            $filiere = "";  
            break;  
        case 3:  
            $filiere = "--vert";  
            break;  
        case 4:  
            $filiere = "--bleu";  
            break;  
        case 5:  
            $filiere = "--rouge";  
            break;  
        default: $filiere = "";  
    }  
    return doubleCell('', 'WIDTH="3%"', '<IMG  
SRC="'. $GLOBALS['racine']. '/images/puce'. $filiere. '.gif' " WIDTH="15"  
HEIGHT="15" HSPACE="5" align="middle"><SPAN  
CLASS="titre">'. $sousTitre. '</SPAN>', 'WIDTH="96%" HEIGHT="20"');  
}  
//-----  
//-- Construction d'une liste <UL> --  
//-----  
function liste($x){  
    $txt = '<UL>'. $x. '</UL>';  
    return $txt;  
}  
//-----  
//-- Construction d'une liste <UL> --  
//-----  
function li($x){  
    $txt = '<LI>'. $x. '</LI>';  
    return $txt;  
}  
  
//-----  
//-- Construction d'un lien --  
//-----  
function lien($texte, $lien){  
    $lien = '<A HREF="'. $lien. '">'. $texte. '</A>';  
    return $lien;  
}  
  
//-----  
//-- Construction d'un lien popup --  
//-----  
function popup($texte, $lien){  
    $lien = "<A  
HREF=\"javascript:openWin('". $lien. "', 'popup', 'scrollbars=yes, resizable=yes  
, width=760, height=700')\">". $texte. "</A>";  
    return $lien;  
}  
  
//-----  
//-- Construction d'un lien popup --  
//-----  
function popup2($texte, $lien, $args){  
    $lien = "javascript:openWin('". $lien. "', '". $texte. "', '". $args. "')";  
    return $lien;  
}  
}
```

```
//-----  
/-- Construction d'un paragraphe --  
//-----  
function p($texte){  
    return "<P><B>".text($texte)."</B></P>";  
}  
  
//-----  
/-- Construction d'un paragraphe blockquote --  
//-----  
function pb($texte){  
    return "<P><BLOCKQUOTE>".text($texte)."</BLOCKQUOTE></P>";  
}  
  
//-----  
/-- Construction d'un texte formaté --  
//-----  
function text($texte){  
    return nl2br($texte);  
}  
  
//-----  
/-- Construction des liens des profs --  
//-----  
function prof($nom, $prof, $resp, $id){  
    $ligneProf = '<LI><A HREF="cv.php?uid='.$id.'">'.$nom.'</A><SPAN  
STYLE="font-size: 10pt; font-family: Verdana; color:  
rgb(51,51,153);">'.$prof.' <I><strong>'.$resp.'</strong></I></span></LI>';  
    return $ligneProf;  
}  
  
//-----  
/-- Construction de ligne de lien des profs --  
//-----  
function constLiens($texte){  
    return doubleCell('', 'WIDTH="3%"', liste($texte), 'WIDTH="96%"  
HEIGHT="20"');  
}  
  
?>
```

Exemple d'un des programmes d'administration :

```
<?php  
/*****  
* Nom du programme : admin/creer_biblio.php *  
* Page de création d'une publication *  
* Date de génération du programme : dimanche 16 mai 2004 *  
* Copyright : ISTM - Tous droits réservés. *  
*****/  
include("include/session.inc");  
  
$titre = "Gérer une publication";  
$rubrique = "istm_haut.jpg";  
include("../include/functions.inc");  
include("../include/entete.inc");  
  
/-- Récupération des variables --
```



```
$ID_BIBLIO = $_GET['id_biblio'];
$uidLien = $_GET['uidLien'];
$typeReq = $_GET['typeReq'];

?>
<SCRIPT LANGUAGE="JavaScript">
<!--
//-- Javascript de vérification des champs du formulaire avant validation -
-
function valider(){

    if (document.forms[0].elements['NOM_BIBLIO'].value=="") {
        alert("Intitulé obligatoire");
        document.forms[0].elements['NOM_BIBLIO'].focus();
        return false;
    }
    if (document.forms[0].elements['DESCR_BIBLIO'].value.length>10000) {
        alert("Description de moins de 10000 caractères\nTaille de votre
champ : "+document.forms[0].elements['DESCR_BIBLIO'].value.length);
        document.forms[0].elements['DESCR_BIBLIO'].focus();
        return false;
    }
    if (document.forms[0].elements['EDITION_BIBLIO'].value.length>200) {
        alert("Moins de 200 caractères pour l'édition\nTaille de votre
champ : "+document.forms[0].elements['EDITION_BIBLIO'].value.length);
        document.forms[0].elements['EDITION_BIBLIO'].focus();
        return false;
    }
    //document.forms[0].action="<?php echo $PHP_SELF;?>?OK=1";
    return true;
}
//-->
</SCRIPT>
<?php

//-- Sélection de la bibliographie pour mise à jour (update) --
if ($typeReq != "insert"){
    $sqlBiblio="select * from biblio where id_biblio=$ID_BIBLIO";
    dbconnect($sqlBiblio);
    $row = pg_fetch_row($GLOBALS['resultat']);
}
//-- Titre de la page --
echo tab(ligneTab(constTitre($titre,0)));

//-- Début du formulaire --
echo "<CENTER>";
echo '<FORM NAME="creer_biblio" METHOD="POST" ACTION="requetes.php"
onsubmit="return valider()">';
echo "<INPUT TYPE=hidden SIZE=7 MAXLENGTH=7 NAME=ID_BIBLIO
VALUE=".$row[0].">";
echo "<INPUT TYPE=hidden SIZE=7 MAXLENGTH=7 NAME=uid VALUE=".$uidLien.">";
echo '<INPUT TYPE="hidden" NAME="typeReq" VALUE="'.$typeReq.'">';

//-- Sélection des types de biblio --
$sql = "SELECT id_type_biblio,nom_type_biblio FROM type_biblio";
dbconnect($sql);
$selectType = '<SELECT NAME="ID_TYPE_BIBLIO">';
while ($ligne = pg_fetch_row($GLOBALS['resultat'])) {
    $selectType.= '<OPTION VALUE="'.$ligne[0].'">'.$ligne[1].</OPTION>';
}
$selectType.='</SELECT>';
```




```
//-----  
//-- On construit les champs --  
//-----  
$tableau.= ligneDoubleCell("Nom de la publication :", "<INPUT TYPE=TEXT  
SIZE=70 MAXLENGTH=200 VALUE='$row[2]' NAME=NOM_BIBLIO>");  
$tableau.= ligneDoubleCell("Type de publication :", $selectType);  
$tableau.= ligneDoubleCell("Description :", "<TEXTAREA NAME=DESCR_BIBLIO  
ROWS=10 COLS=69>$row[3]</TEXTAREA>");  
$tableau.= ligneDoubleCell("Auteur(s) :", "<INPUT TYPE=TEXT SIZE=70  
MAXLENGTH=100 VALUE='$row[4]' NAME=AUTEUR_BIBLIO>");  
$tableau.= ligneDoubleCell("Edition :", "<TEXTAREA NAME=EDITION_BIBLIO  
ROWS=3 COLS=69>$row[5]</TEXTAREA>");  
$tableau.= ligneDoubleCell("Année :", "<INPUT TYPE=TEXT SIZE=9 MAXLENGTH=9  
VALUE='$row[6]' NAME=ANNEE_BIBLIO>");  
//-- Affichage du tableau final --  
echo tab2($tableau);  
echo '<INPUT TYPE="submit" VALUE="Valider">';  
echo '</FORM>';  
echo "</CENTER>";  
  
include("../include/piedpage.inc");  
?>
```