



Service de rangement de document automatisé

Florent Baillif | Francel Marcoz | Loïc Toulcanon

Projets tutorés 2010/2011
Département Réseaux et Télécom

1. BUT DU SUJET	4
2. LES FONCTIONNALITES	4
3. IDEES DE DEPART	5
4. RECHERCHES	6
5. CREATION DE LA PAGE WEB (EN PARTIE REALISEE)	10
6. SERVICE DE RANGEMENT DES DOCUMENTS	14
7. REALISATION DU PROJET	14
8. BIBLIOGRAPHIE	16

1. BUT DU SUJET

Qu'est-ce qu'un service de rangement de document automatisé ?

C'est un serveur qui reçoit des fichiers envoyés par différents utilisateurs (comme un hébergeur de fichiers). Dans notre cas, les utilisateurs seront les élèves (envoi de fichiers) et les enseignants (récupération de fichiers). En revanche, à la différence d'un hébergeur de fichier, le serveur propose un classement des fichiers reçus dans des répertoires/sous-répertoires selon des critères posés par les enseignants (extensions, contenus, etc...)

Il faut créer aussi un site web qui servira d'interface entre les utilisateurs et les fichiers (sur le serveur). En effet le site web permettra d'envoyer des fichiers et de les récupérer (selon enseignant/élève)

2. LES FONCTIONNALITES

Le service de rangement aura 3 fonctions principales :

- Tout d'abord l'envoi de fichier (upload), essentiellement pour les élèves. Il s'agit pour les élèves de rendre leurs travaux sous format numérique à leurs professeurs, selon des critères précis décidé par ce dernier (syntaxe du nom du fichier, extension particulière, créneaux d'heures) en passant par notre service. Nous pouvons aussi proposer des critères généraux aux professeurs qu'ils pourront imposer à leurs élèves. Nous pouvons par exemple proposer une syntaxe pour le nom des fichiers du type :

UE-T[P D][n]-nom1-prénoms1-nom2-prénom2.[ex]
où n est le numéro du TD ou du TP et ex l'extension.

Exemple: I1-TP2-marcoz-francel.pdf

- Le service aura aussi une fonction de rangement. Après récupération des travaux des élèves, le service aura pour fonction de classer, de ranger les fichiers afin d'être facilement accessible pour les professeurs
- Téléchargement des fichiers (download), essentiellement pour les professeurs. Après classement des fichiers, les professeurs auront accès aux fichiers envoyés par les élèves pour correction.

Quelles sont les différentes possibilités pour créer ce service ?

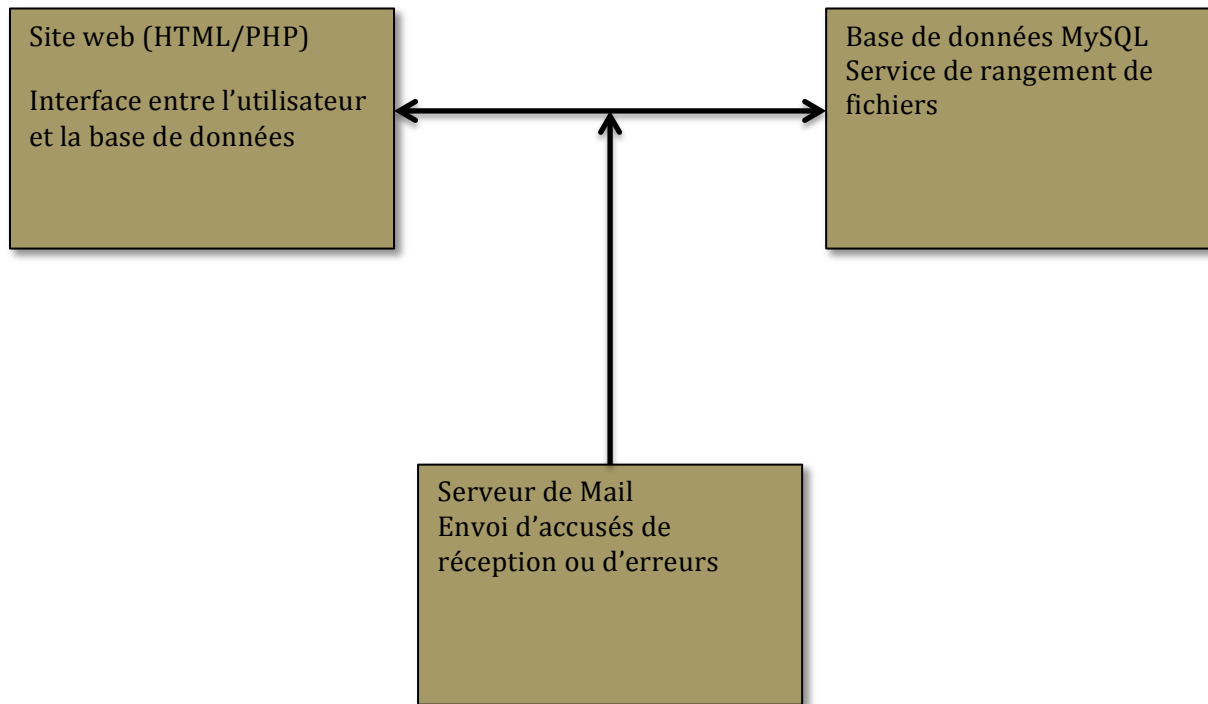
3. IDEES DE DEPART

- Nous avons d'abord pensé à créer un serveur mail. Le serveur reçoit les mails avec les pièces jointes (compte-rendu, .doc, .PDF) puis les stockent dans un répertoire. On programme ensuite un script bash qui va classer les fichiers stockés dans ce répertoire, selon le type (.doc, .PDF)
- Notre deuxième idée est de créer un serveur FTP (plus facile à mettre en place) à la place du serveur mail. Et de gérer le classement des dossiers grâce au même type de script qu'avec le serveur mail.
- Après discussions avec Mr MURAD nous nous sommes posé les questions suivantes : Quel type de serveur faut-il utiliser ? Serveur Mail ou FTP ?

A la fin on en a conclu qu'il faudra utiliser les deux types de serveur :

On doit d'abord créer une page web simple dotée des champs : Nom, Prénom, d'un champ permettant de choisir le fichier à envoyer et d'un bouton envoyer. Le fichier à envoyer doit être nommé manière précise (selon les exigences de l'enseignant ou selon un modèle que nous auront préalablement proposé) ou encore selon un créneau d'heure précis (la aussi décidé par le professeur). Si la syntaxe du nom du fichier n'est pas respectée, le site le signalera avant l'upload du fichier. Et dans le cas où l'envoi du fichier ne respecte pas le créneau un mail d'erreur est envoyé à la personne émettrice et un message d'erreur est affiché sur la page web. Dans le cas contraire, si le fichier possède la bonne syntaxe, un mail est envoyé à l'élève et au professeur correspondant, de type accusé de réception ; qui servira de preuve pour le professeur et de recépissé pour l'élève.

Dans une seconde partie, il faut mettre en œuvre une base de données MySQL (On tous d'abord pensé a MySQL, mais d'autres types de base de données sont possibles) qui va permettre de classer les fichiers reçus (par enseignants, par type de fichier etc.). Le transfert de fichier utilise le protocole FTP. Aussi, un service de réception des fichiers doit être mis en place. Par exemple, pour récupérer les compte-rendu des élèves, les enseignants accèdent, par l'intermédiaire du site, aux fichiers qui leur sont destinés.



Quelles sont les connaissances, les outils nécessaires dans la création de ce service ?

4. RECHERCHES

Qu'est-ce qu'un serveur ?

Un serveur est une machine ou un programme qui offre un service à un client. Celui-ci présente une architecture client/serveur.

Caractéristiques d'un serveur :

- il est initialement passif (ou esclave, en attente d'une requête) ;
- il est à l'écoute, prêt à répondre aux requêtes envoyées par des clients ;
- dès qu'une requête lui parvient, il la traite et envoie une réponse.

Caractéristiques d'un client :

- il est actif le premier (ou maître) ;
- il envoie des requêtes au serveur ;
- il attend et reçoit les réponses du serveur.

Le client et le serveur doivent bien sûr utiliser le même protocole de communication. Un serveur est généralement capable de servir plusieurs clients simultanément.

Mise en place

La mise en place d'un serveur se fait en deux étapes:

Installation (et configuration) du système d'exploitation

Installation (et configuration) de l'une ou des application(s) adaptée(s) au(x) service(s) désiré(s)

Les types de serveurs :

Dans le chapitre précédant, nous avons parlé de serveurs au sens large. Dans l'informatique, on distingue trois types de serveurs:

- Un **serveur de fichier** enregistre et distribue les documents et fichiers partagés par les utilisateurs. Les configurations ne sont généralement très grosses.
- Un **serveur d'application** permet d'utiliser un programme sur un serveur à partir de tous les postes clients simultanément, principalement des applications qui utilisent des bases de données (gestion de fabrication, commerciale, comptabilité, stock, ...). Ces applications doivent être programmée pour gérer les partages. La configuration de ces serveurs est généralement plus importante (par exemple des multi-processeurs). Dans tous les cas, un fichier n'est pas sauvegardé lors d'un backup si un utilisateur l'accède.
- Un **serveur d'impression** partage des imprimantes. Actuellement, différents modèles sont directement connectés à un réseau Ethernet, des boîtiers spécifiques sont également commercialisés.

En pratique, un serveur reprend souvent les trois applications.

Les applications de serveur :

Authentification

Bases de données

Courrier électronique

Gestion d'un réseau :

-Administration

-DHCP

-DHCP

-DNS

-Proxy

-Routage

-VPN

....

Qu'est ce qu'un serveur FTP ?

Le serveur FTP (File Transfer Protocol) permet, de transférer des fichiers par Internet ou par le biais d'un réseau informatique local (intranet). Si on a l'autorisation, notamment en se loguant, on peut télécharger et envoyer des fichiers sur un ordinateur distant faisant fonctionner un tel serveur.

C'est le protocole FTP permet le transfert de fichiers d'une machine locale (un ordinateur personnel) vers une machine distante (souvent un hébergeur de page personnelle ou un serveur quelconque).

Qu'est ce qu'un serveur mail ?

Un serveur de messagerie électronique est un logiciel serveur de courrier électronique (courriel). Il a pour vocation de transférer les messages électroniques d'un serveur à un autre. Un utilisateur n'est jamais en contact direct avec ce serveur mais utilise soit un client de messagerie, soit un courriel leur web, qui se charge de contacter le serveur pour envoyer ou recevoir les messages.

La plupart des serveurs de messagerie possèdent ces deux fonctions (envoi/réception), mais elles sont indépendantes et peuvent être dissociées physiquement en utilisant plusieurs serveurs.

- **L'envoi**

L'envoi d'un courrier électronique, entre l'utilisateur et le serveur se déroule généralement via le protocole SMTP. Puis c'est au serveur d'envoyer le message au serveur du destinataire, cette fonction est appelée Mail Transfer Agent en anglais, ou MTA.

- **Réception**

La réception d'un courrier électronique s'effectue elle aussi en deux temps. Le serveur doit recevoir le message du serveur de l'expéditeur, il doit donc gérer des problèmes comme un disque plein ou une corruption de la boîte aux lettres et signaler au serveur expéditeur toute erreur dans la délivrance. Il communique avec ce dernier par l'intermédiaire des canaux d'entrée-sortie standard ou par un protocole spécialisé comme LMTP (Local Mail Transfer Protocol). Cette fonction de réception est appelée Mail Delivery Agent en anglais, ou MDA.

Finalement, lorsque le destinataire final désire accéder à ses messages, il lance une requête au serveur qui transmet les messages reçus généralement via le protocole POP3 ou IMAP. La plupart des clients de messagerie sont configurés de manière à interroger régulièrement le serveur de messagerie (par exemple, toutes les minutes), ce qui rend l'étape 3 du processus complètement transparente pour le destinataire.

BASES DE DONNÉES ET INTERNET AVEC PHP ET MySQL Magali Contensin
Edition DUNOD

Qu'est ce que le HTML ?

HTML est le format de données conçu pour représenter les pages web. C'est un langage de balisage qui permet d'écrire de l'hypertexte. HTML permet également de structurer et de mettre en forme le contenu des pages, d'inclure des ressources multimédias dont des images, des formulaires de saisie, et des éléments programmables.

*Exemple de code (voir Annexe **Code 1**)*

Qu'est ce que le PHP ?

Le langage PHP est un langage de scripts exécutés par un serveur local (exemple : Wamp server) qui est dédié à la génération de pages WEB dynamiques, il permet l'extraction et la manipulation de données pour un grand nombre de bases de données.

*Exemple de code (voir Annexe **Code 2**)*

Qu'est ce qu'une base de données?

Une base de données est une entité dans laquelle il est possible de stocker des données de façon structurée et avec le moins de redondance possible. Ces données doivent pouvoir être utilisées par des programmes, par des utilisateurs différents. Ainsi, la notion de base de données est généralement couplée à celle de réseau, afin de pouvoir mettre en commun ces informations, d'où le nom de base. On parle généralement de système d'information pour désigner toute la structure regroupant les moyens mis en place pour pouvoir partager des données.

Qu'est ce que le MySQL ?

MySQL est un système de gestion de bases de données client-serveur. Ce SGBD est distribué sous licence GPL ou sous licence commerciale. Il est disponible pour de nombreux systèmes d'exploitation et propose des interfaces de programmation pour divers langage dont C, C++, Java, Perl, PHP et Python. MySQL comporte un serveur (mysqld), des clients (mysql, mysqldump, mysqlimport) et des outils d'administration. Il sert à effectuer des opérations ordinaires telles que consulter, modifier, construire, organiser, transformer, copier, sauvegarder ou restaurer des bases de données.

Au départ il nous paraissait impossible d'effectuer un rangement de fichier avec les scripts. Cependant, après discussion avec Mr Lorion, nous avons eu une approche différente.

Qu'est-ce que CouchDB ?

CouchDB est un serveur de base de données qui au lieu d'être ordonnée en lignes et en colonnes, la base de données est une collection de documents JSON. De plus, CouchDB contient un serveur HTTP qui permet d'effectuer des requêtes, et renvoie ses données sous forme JSON. On peut ainsi interroger un serveur CouchDB directement avec un navigateur web, ou on peut exécuter des requêtes avec Javascript.

Qu'est ce qu'un document JSON ?

JSON (JavaScript Object Notation) est un format de données pour l'échange entre le navigateur et le serveur.

JSON utilise la notation des objets JavaScript pour transmettre de l'information structurée, d'une façon plus compacte et plus proche des langages de programmation.

Nous pouvons donc séparer le projet en 3 grands blocs :

- La création d'une page web HTML ou PHP (Interface utilisateur)
- La mise en place d'une base de données MySQL (Classement des fichiers)
- Le serveur mail (Accusés, réception/erreur)

Quelle partie du projet peut-on accomplir avec nos connaissances actuelles ?

5. CREATION DE LA PAGE WEB (EN PARTIE REALISEE)

Pour la création de la page WEB on a du commencer par adopter les langages approprié tels que le PHP, le HTML/CSS (Appris en cours).

Pourquoi utilise-t-on le PHP ?

Comme notre site web sera composé de page de login d'une part pour les étudiant et d'autre part pour les enseignant, il se doit d'être dynamique. Le site doit aussi communiquer avec une base de données MySQL, le PHP semble être le langage approprié pour la page web, en plus du HTML/CSS qui serviront pour la mise en forme de la page.

Architecture du site web

Le site sera composé de plusieurs entités différentes, une page d'accueil, une page Upload, un Champ pour l'administration et un espace « enseignants » pour la récupération les fichiers demandés.

- **Page D'accueil**

Cette page sera fondait de façon à ce que l'utilisateur pourra se loguer et se faire identifier à l'aide des codes appropriés en langage web PHP (*voir Annexe **Code 3 Login***).

Page d'accueil Sous Firefox :



Cette page contient des champs pour l'envoi du nom d'utilisateur et pour le mot de passe, ces informations-là seront stockées dans des variables appelées variables **\$_POST** qui seront ensuite envoyées dans une base de données (voir chapitre 6) préalablement créée et remplie.

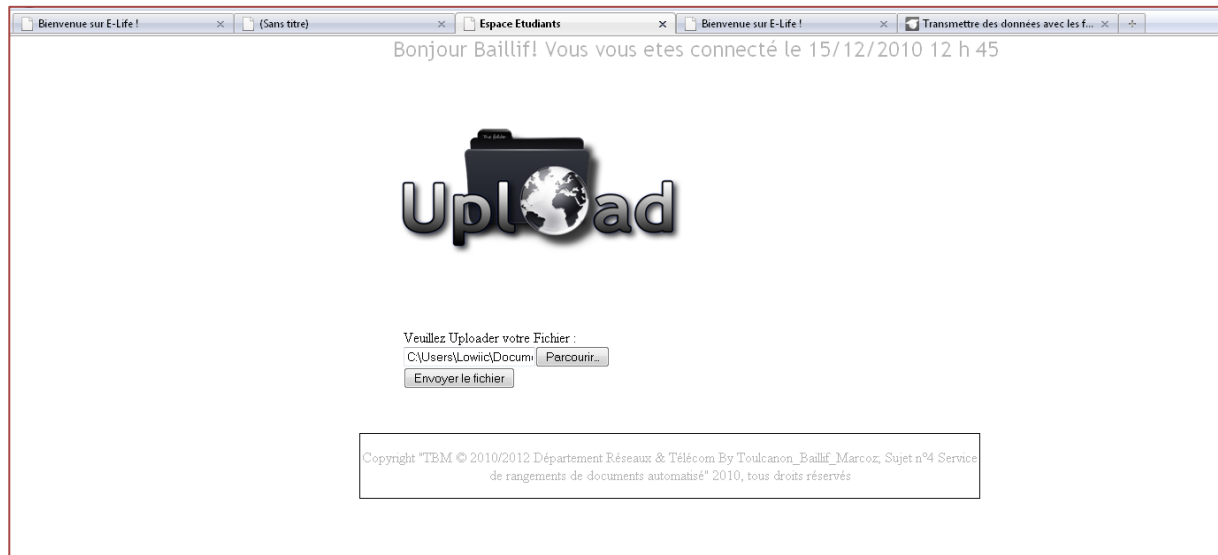
Le code vérifiera tout d'abord si le client existe dans la base de données et si et seulement si le client existe, il vérifiera le type d'utilisateur qui sera en train de se connecter.

Avec les informations collectées, si l'utilisateur est un étudiant, le code dirigera l'utilisateur vers la page Upload, sinon le code dirigera l'utilisateur vers l'espace enseignant.

- **Espace Etudiant Upload**

Cette page sera concrètement dédiée aux étudiants, ici ils pourront envoyer ces fichiers avec le format demandé (voir Annexe **Code 4 Upload**).

Espace Etudiants Sous Firefox :

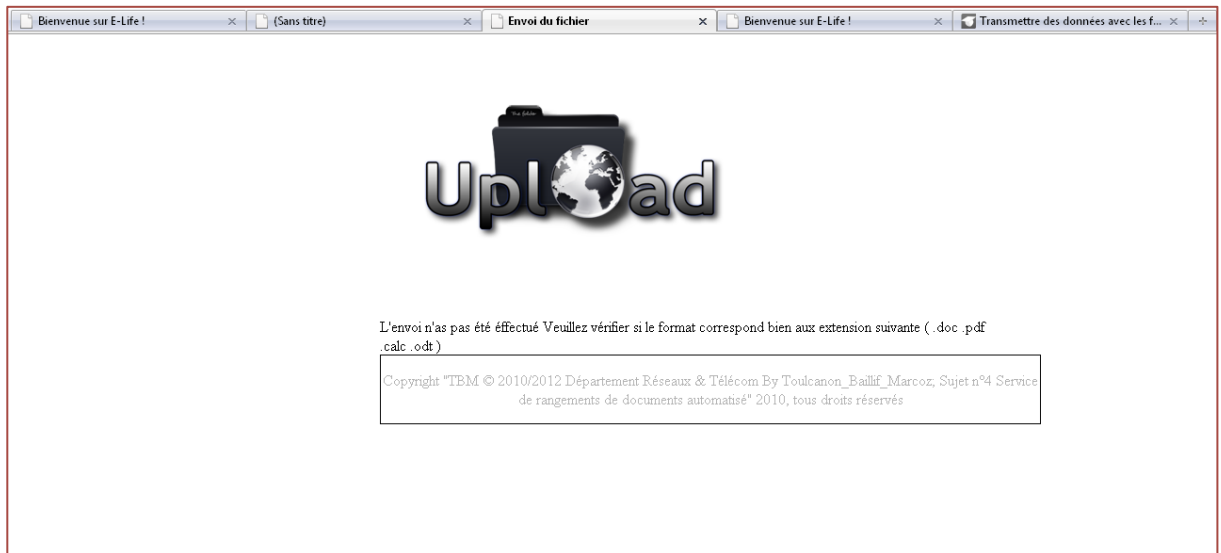


Dans espace nous avons poser plusieurs condition à l'envoi des fichiers (*voir Annexe **Code 5 condition***).

Si le fichier demandé est à l'extension demander dans notre cas (jpg , png , gmp).
Le fichier sera envoyé dans la base de données :



A contrario si le fichier n'est pas à l'extension demander
Le fichier ne sera pas envoyer et l'étudiant sera avertis par un message d'erreur :



- **Espace enseignant**

Et en ce qui cette espace, les enseignants pourront downloader ou récupérer les fichiers précédemment déposer par les étudiants dans la base de données

Problématique :

(« Cette espace nous a posé un problème pour sa fondation car pour pouvoir la créer il faut élaborer de redirection différente selon le type de client dans la page accueil c'est-à-dire faire de redirection différente , au jour d'aujourd'hui nous ne pouvons faire qu'une seule redirection dans notre cas nous l'avons faites pour l'espace Etudiant (voir Espace Etudiants Sous Firefox) »)

Conclusion :

Lorsqu'un utilisateur se connecte, la page web vérifie dans une liste (préalablement remplie) située dans la base de donnée, si il s'agit d'un élève ou d'un enseignant. Après s'être connecté deux types de page :

- **Pour les enseignants : une page permettant de récupérer les fichiers qui leur sont destinés (page de récupération)**
- **Pour les élèves : un page composée d'un champ permettant d'envoyer un fichier (page d'envoi)**

6. SERVICE DE RANGEMENT DES DOCUMENTS

Notre première idée fut d'utiliser une base de données MySQL pour le rangement des fichiers.

Utilisation de scripts bash

N'ayant aucune connaissance sur les bases de données, notre seconde idée est d'écrire des scripts bash : le serveur reçoit un fichier dans un répertoire de stockage qu'on pourrait qualifier de temporaire. En effet, les fichiers qui sont dans ce répertoire sont déplacés automatiquement grâce à un/plusieurs fichier(s) bash qui s'exécute (nt). Par exemple, on pourrait avoir un fichier bash qui scrute le nom, repérant le nom de l'émetteur ou autre, et qui appelle un autre script, permettant de déplacer les fichiers dans les dossiers adéquats (si il faut, créer des dossiers, et déplacer les fichiers dedans)

On s'est longtemps posé la question entre l'utilisation d'une base de données, ou de scripts bash.

7. REALISATION DU PROJET

Les grandes lignes dans la réalisation du projet :

```
graph TD; A[1. Mise en place de l'interface de connexion au service de rangement] --> B[2. Mise en place de la base de donnée du service et du système de vérification des fichiers]; B --> C[3. Interconnexion de l'ensemble : entre l'interface et la base de donnée];
```

1. Mise en place de l'interface de connexion au service de rangement

2. Mise en place de la base de donnée du service et du système de vérification des fichiers

3. Interconnexion de l'ensemble : entre l'interface et la base de donnée

- 1. Mise en place de l'interface de connexion du service de rangement

Il s'agit ici de créer une ou des pages web permettant d'une part aux élèves de rendre leurs travaux, aux professeurs de récupérer ces travaux, d'autre part, pour nous de pouvoir administrer à distance le service.

- 2. Mise en place de la base de données du service

Il s'agit des fondations du service, la majorité des fonctionnalités proposées par le service de rangement se repose sur la base de donnée. Ainsi le choix du système de gestion peut être crucial et pourra déterminer la qualité de service.

- 3. Interconnexion de l'ensemble

L'interconnexion de deux étapes précédents formera le produit final du service, c'est-à-dire sa mise en service, sa pleine utilisation.

Pour réaliser cela, nous avons donc réparti les tâches entre les 3 membres du groupe :

Toulcanon Loïc	<ul style="list-style-type: none">• Architecture de l'interface web
Baillif Florent	<ul style="list-style-type: none">• Système de gestion de base de données
Marcoz Francel	<ul style="list-style-type: none">• Systeme de gestion de base de données• Script pour la vérification de fichier

8. BIBLIOGRAPHIE

SITES INTERNET :

<http://doc.ubuntu-fr.org/serveur>
<http://fr.wikipedia.org/wiki/Client-serveur>
http://www.ybet.be/hardware2_ch2/hard2_ch2.htm#types-serveur
<http://doc.ubuntu-fr.org/serveur>
http://fr.wikipedia.org/wiki/Serveur_FTP
<http://www.dicodunet.com/definitions/reseaux/ftp.htm>
http://fr.wikipedia.org/wiki/Serveur_de_messagerie_%C3%A9lectronique
http://fr.wikipedia.org/wiki/Hypertext_Markup_Language

LIVRES :

« *Bases de données et internet avec PHP et MySQL* » de Magali CONTENSIN
« *Montez votre serveur web de A à Z* » de Sékine COULIBALY, Pierre FONTAINE, Francois HOUSTE, Pierre-Emmanuel MULLER

ANNEXE

Code 1 :

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="fr" >
  <head>
    <title>Exemple HTML </title>
    <meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />
    <link rel="stylesheet" media="screen" type="text/css" title="Design" href="design.css"
  />
</head>
<body>

<!-- Ici se trouvera le contenu de la page -->
</body>
</html>
```

Code 2 :

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="fr" lang="fr">
  <head>
    <title>Ceci est une page de test avec des balises PHP</title>
    <meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />
  </head>
  <body>
    <h2>Page de test</h2>

    <p>
      Cette page contient du code (x)HTML avec des balises PHP.<br />
      <?php /* Insérer du code PHP ici */ ?>
      Voici quelques petits tests :
    </p>
  </body>
</html>
```

Code 3 Login :

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="fr" >
  <head>
    <title>Sujet N°4 </title>
    <meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />
    <link rel="stylesheet" media="screen" type="text/css" title="Design" href="css_connexion.css" />
  </head>

  <body>
    <br/><br/><br/><br/><br/>
    <?php include("entete.php"); ?>
    <div id="connexion">
      <div class="connexion">
        <form action="Profile/profil.php" method="post"> <!--formulaire du profil -->
        <p>
          <label for="prenom">Nom</br></label> <!-- Ici l'utilisateur entre son nom d'utilisateur -->
          <input type="text" name="prenom" value="Admin" />
          <label for="pass"> Mot de passe </br> :</label><!-- Ici l'utilisateur entre son mot de passe -->
          <input type="password" name="pass" id="pass" tabindex="20" />
          <input type="submit" value="Connexion" /><!--formulaire du bouton valider-->
        </p>
      </div>
    <br/><br/><br/><br/><br/><br/><br/><br/><br/><br/><br/><br/><br/><br/><br/><br/>
    <?php include("pieddepage_connexion.php"); ?><!-- Le pied de page est appelé dans une autre page -->

  </body>
</html>
```

Code 4 Upload :

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="fr" >
  <head>
    <title>Bienvenue sur E-Life !</title>
    <meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />
    <link rel="stylesheet" media="screen" type="text/css" title="Design" href="design2.css"
  />
  </head>
  <body>
    <?php
    if (isset($_POST['prenom']) AND($_POST['prenom'] == "Toulcanon" OR$_POST['prenom'] ==
"Marcoz" OR$_POST['prenom'] == "Baillif") AND ($_POST['pass']) AND $_POST['pass'] ==
"1234" OR $_POST['pass'] == "florent" OR $_POST['pass'] == "Cindy" )
    { ?>
    <?php include("date_profil.php"); ?>
    <?php include("entete_profil.php"); ?>
    <!--Form concernant les input et le parcours de fichiers -->
    <form action="Formulaire.php" method="post" enctype="multipart/form-data" id="uplo">
      <p>
        Veuillez Uploader votre Fichier :<br />
        <input type="file" name="monfichier" /><br />
        <input type="submit" value="Envoyer le fichier" />
      </p>
    </form>
    <?php /*include("menu_profil.php");*/ ?>

    <?php include("pieddepage_profil.php"); ?>

    <?php
    }
    else // Sinon, on affiche un message d'erreur
    {
    ?>
      <?php include("error_connexion.php");
    }
    ?>
  </body>
</html>
```

Code 5 condition :

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="fr" >
  <head>
    <title>Envoi du fichier</title>
    <meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />
    <link rel="stylesheet" media="screen" type="text/css" title="Design"
href="design2.css" />
  </head>

  <body>
    <?php include("entete_profil.php"); ?>
    <?php
// Testons si le fichier a bien été envoyé et s'il n'y a pas d'erreur
if (isset($_FILES['monfichier']) AND $_FILES['monfichier']['error'] == 0)
{
    // Testons si le fichier n'est pas trop gros
    if ($_FILES['monfichier']['size'] <= 1000000)
    {
        // Testons si l'extension est autorisée
        $infosfichier = pathinfo($_FILES['monfichier']['name']);
        $extension_upload = $infosfichier['extension'];
        $extensions_autorisees = array('jpg', 'jpeg', 'gif', 'png');
        if (in_array($extension_upload, $extensions_autorisees))
        {
            // On peut valider le fichier et le stocker définitivement
            move_uploaded_file($_FILES['monfichier']['tmp_name'], 'uploads/' .
basename($_FILES['monfichier']['name']));
            echo "L'envoi a bien été effectué !";
        }
        else
        {
            echo "L'envoi n'as pas été effectué Veuillez vérifier si le format correspond
bien aux extension suivante ( .doc .pdf .calc .odt )";
        }
    }
}

?>
    <?php /*include("menu_profil.php");*/ ?>
    <?php include("pieddepage_profil.php"); ?></body></html>
```