



Vendredi 1 juin 2012

Documentation développeur

- [Structure](#) (page 2 à 5)
- [Fonctions](#) (page 6 à 9)
- [Développement](#) (page 10 à 11)
- [Versions](#) (page 12 à 15)

Auteur : **Emmanuel Delage**

Graphismes : **Cédric Gagnevin**

La structure du logiciel

Le programme phpMyLab est découpé en 26 fichiers :

Fichiers généraux

- index.php (30 lignes): redirige vers "configuration/index.php" si l'installation n'a pas été faite, sinon redirige vers "reception.php".
- config.php: contenant les variables de configuration du programme (génééré automatiquement).
- connectionPHPMYLABDB.php: information de connection à la base de données (génééré automatiquement).
- reception.php (670 lignes): gestion de l'identification, du choix du module et initialisation de la session PHP.
- calendrier_variables.php (30 lignes): variables pour les calendriers, ex: jours fériés en statique en fonction des années.
- javascript.js (50 lignes): contient les fonctions utilisées pour afficher l'heure, initialiser la date des calendriers (CONGES,MISSIONS) et limiter le nombre de caractères.
- mdp.js (140 lignes): contient les fonctions nécessaires pour déterminer la complexité d'un mot de passe (reception.php).
- en_tete.php (150 lignes): contient l'en-tête des pages (inclut dans les pages des modules).
- pied_page.php (30 lignes): contient le pied de page (inclut dans les pages des modules).
- style.php (1500 lignes): style de l'application.

Fichiers d'installation (dans le répertoire "configuration")

- index.php (250 lignes): détermine si les ré-requis sont respectés.
- etape1.php (150 lignes): étape 1 de l'installation, renseignement des paramètres relatifs à l'organisme.
- etape2.php (230 lignes): étape 2 de l'installation, création de la base de données.
- etape3.php (660 lignes): étape 3 de l'installation, choix des modules.
- etape4.php (240 lignes): étape 4 de l'installation, renseignement des derniers param&egrav;tres nécessaires.
- finConfig.php (340 lignes): génération des fichiers de configuration.
- style_config.css (380 lignes): style de la procédure d'intallation automatisée.

Module MISSIONS

- missions.php (1471 lignes): gestion des missions.
- calendrier_missions.php (244 lignes): dessin du calendrier des missions avec la bibliothèque graphique GD.

Module CONGES

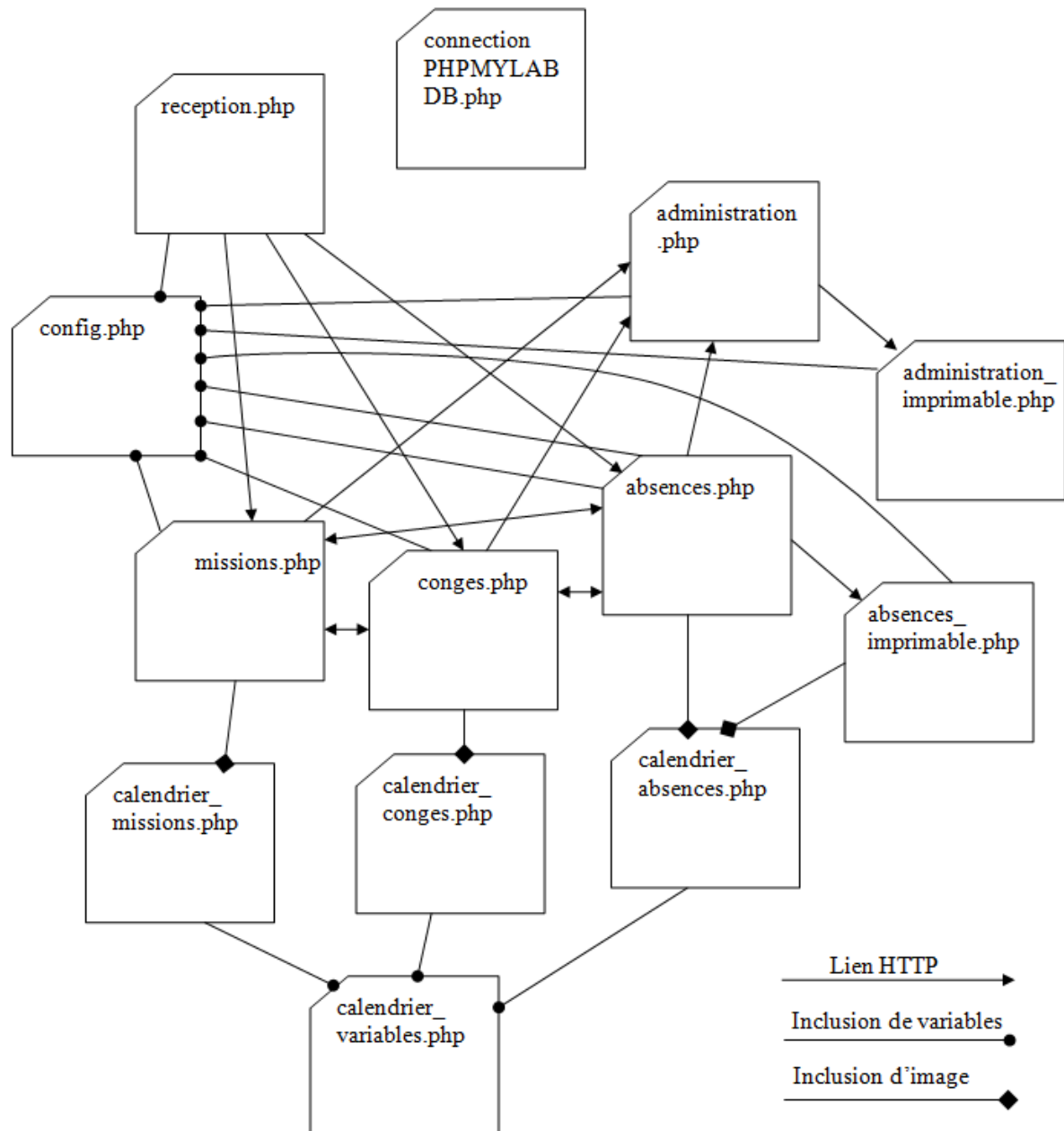
- conges.php (1702 lignes): gestion des congés.
- calendrier_conges.php (230 lignes): dessin du calendrier des congés avec la bibliothèque graphique GD.

Module PLANNING

- planning.php (504 lignes): gestion des absences.
- calendrier_absences.php (519 lignes): dessin du calendrier des absences avec la bibliothèque graphique GD.
- absences_imprimable.php (160 lignes): Mise en forme pour l'impression.

Page Administration

- administration.php (2195 lignes): administration de phpMyLab.
- administration_imprimable.php (198 lignes): Mise en forme pour l'impression des congés.



Liasons entre les fichiers du programme

Les inclusions des variables du fichier "connectionPHPMYLABDB.php" ne sont pas représentées pour des raisons de lisibilité. Ce fichier est inclus dès que le programme fait appel à la base données.

PhpMyLab a été développé à partir du programme MISSIONS qui n'est plus maintenu. Cependant, phpMyLab a hérité de la structure interne du programme MISSIONS sauf que la partie identification a été réécrite dans le fichier "reception.php".

le fichier "missions.php" contenait 1727 lignes.
 Voici l'exemple de structure interne du programme MISSIONS(v1.1) :

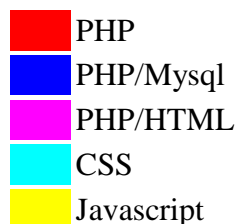
Partie	Titre	Ligne	Nb lignes	Proportion	Langages
A	Fonction sendmail()	24	81	5.3%	PHP
B	Initialisations générales (configuration du logiciel et configuration PHP)	105	30	2%	PHP
C	Initialisation de la session PHP et des variables du programme (tableau de connexion, de missions et de correspondance)	135	63	4.2%	PHP/MySQL
D	Gestion des mots de passe	298	142	9.4%	PHP/MySQL
E	Gestion des variables de recherche	440	43	2.8%	PHP
F	Gestion des variables de mission	483	40	2.6%	PHP
G1	Recherche simple des missions	526	131	8.7%	PHP/MySQL
G2	Recherche évoluée des missions	657	132	8.7%	PHP/MySQL
G3	Reinitialiser, page suivante et précédente la Recherche évoluée	789	23	1.5%	PHP
G4	Ajout d'une demande	812	181	11.9%	PHP/MySQL
G5	Nouvelle demande mission	933	74	4.9	PHP
G6	Valider demande mission	1007	73	4.8	PHP/MySQL
G7	Annuler demande mission	1080	81	5.4%	PHP/MySQL
H1	Script affichage heure	1179	44	2.9%	Javascript
H2	Script CSS	1223	48	3.2%	CSS
H3	Affichage non connecté	1288	100	6.6%	PHP/HTML
H4	Affichage entête connecté	1388	10	0.7%	PHP/HTML
H5	Partie recherche des missions	1398	99	6.6%	PHP/HTML
H6	Affichage des résultats de la recherche évoluée	1497	66	4.4%	PHP/HTML
H7	Partie demande de mission	1563	53	3.5%	PHP/HTML



Pourcentage d'occupation de code source par partie



Types de langages utilisés



Les fonctions

Certaines fonctions ont été récupérées sur le WEB. Par exemple, la fonction "sendmail()" formate les mails pour qu'ils soient lus par la majorité des outils de mail.

Un panel de fonctions PHP pour générer des requêtes MySQL sont décrites ci-dessous. Ces fonctions sont tirées du programme MISSIONS.

I) Initialisation

=> La "session" PHP est démarrée:

```
session_id($sid);
```

```
session_name('mamission');
```

```
session_start();
```

=> Les variables `$_SESSION['connection']` prennent des valeurs NULL.

=> Les variables `$_SESSION['mission']` prennent des valeurs par défaut.

=> Le code source suivant est commun à tous les accès à la base de donnée, par la suite nous ne nous intéresserons qu'à la requête.

```
$link = mysql_connect($mysql_location,$mysql_user,$mysql_password)
```

```
or die('Could not connect: ' . mysql_error());
```

```
mysql_select_db($mysql_base) or die('Could not select database');
```

```
$query = 'SELECT * FROM T_UTILISATEUR WHERE UTILISATEUR = \'".$_SESSION['connection']\n'utilisateur'.\'";
```

```
$result = mysql_query($query) or die('Connection Mysql ; Query failed: ' . mysql_error());
```

```
if ($result) {
```

```
$line = mysql_fetch_array($result, MYSQL_NUM);
```

=> Si le mot de passe est valide pour cet utilisateur alors le script initialise les variables `$_SESSION['connection']`.

=> Si aucun identifiant de demande de mission n'est transmis à travers l'URL, alors le script initialise les variables `$_SESSION['mission']` correspondantes à l'utilisateur en ligne.

```
$query2 = 'SELECT * FROM T_CORRESPONDANCE ORDER BY GROUPE';
```

=> La requête ci-dessus initialise les variables `$_SESSION['correspondance'][$i]`.

```
}
```

```
mysql_free_result($result);
```

```
mysql_close($link);
```

--> Retour au [menu](#) <--

II) Gestion des mots de passe

a) Changement de mot de passe

=> Vérification que les champs soient non vide.

=> La taille du nouveau mot de passe doit contenir entre 6 et 16 caractères.

```
$query = 'SELECT * FROM T_UTILISATEUR WHERE UTILISATEUR = \'$_SESSION[ 'connection' ][ 'utilisateur' ].\'';
```

=> Vérification de l'ancien mot de passe.

```
$query2 = 'UPDATE T_UTILISATEUR SET MOTDEPASSE = \'$_SESSION[ 'connection' ][ 'mot_de_passe_new2' ].\' WHERE UTILISATEUR=\'$_SESSION[ 'connection' ][ 'utilisateur' ].\'';
```

=> Mise à jour du mot de passe.

=> Envoie d'un mail de rappel à l'utilisateur (appel à la fonction "sendmail(")).

b) Rappel de mot de passe

```
$query = 'SELECT * FROM T_UTILISATEUR WHERE UTILISATEUR = \'$_SESSION[ 'connection' ][ 'utilisateur' ].\'';
```

=> Vérification de l'existence de l'utilisateur.

=> Envoie d'un mail de rappel à l'utilisateur.

--> Retour au [menu](#) <--

III) Recherche simple

=> Récupération de l'identifiant de la mission à rechercher soit par URL soit par l'interface graphique.

```
$query = 'SELECT * FROM T_MISSION WHERE ID_MISSION = '.$ind;
```

=> Vérification des droits d'accès de l'utilisateur à cette mission.

=> Mise à jour des variables `$_SESSION['mission']['???']`.

```
$query2 = 'SELECT NOM,PRENOM,SS FROM T_UTILISATEUR WHERE UTILISATEUR = \'$_SESSION[ 'mission' ][ 'utilisateur' ].\'';
```

=> Récupération du nom et du prénom à partir du "nom d'utilisateur".

```
$query2 = 'SELECT DATE_FORMAT (\'$_Sline[ "ALLER_DATE" ].\', \'%d/%m/%Y\');
```

=> Mise en forme des dates.

--> Retour au [menu](#) <--

IV) Recherche évoluée

=> Construction des "clauses where" en fonction des critères choisis par l'utilisateur. Les recherches peuvent être effectuées sur toute ou partie du "Nom", "Prénom" ou "Destination".

```
$where_nom = 'INSTR (T_UTILISATEUR.NOM ,UPPER(\'.$_SESSION[ 'recherche' ][ 'nom' ].\'))>0';  
$where_prenom = 'INSTR (T_UTILISATEUR.PRENOM ,UPPER( \'.$_SESSION[ 'recherche' ][ 'prenom' ].\'))>0';  
$where_groupe = 'T_MISSION.GROUPE = \'.$_SESSION[ 'recherche' ][ 'groupe' ].\'';  
$where_dest = 'INSTR (DESTINATION, UPPER(\'.$_SESSION[ 'recherche' ][ 'destination' ].\'))>0';
```

=> Concaténation des clauses.

=> Comptage du nombre de résultat.

```
$query = 'SELECT COUNT(*) FROM T_MISSION INNER JOIN T_UTILISATEUR ON  
T_MISSION.UTILISATEUR = T_UTILISATEUR.UTILISATEUR '.$where;
```

=> Construction de la clause "ORDER BY" pour le tri par date.

=> Ajout d'un interval de selection des résultats ("LIMIT").

```
$query = 'SELECT ID_MISSION, NOM ,PRENOM ,T_MISSION.GROUPE ,DESTINATION ,ALLER_DATE  
,RETOUR_DATE ,VALIDE FROM T_MISSION INNER JOIN T_UTILISATEUR ON T_MISSION.UTILISATEUR =  
T_UTILISATEUR.UTILISATEUR '.$where.' '.$orderby.' LIMIT '.$_SESSION[ 'recherche' ][ 'limitebasse'  
].', '.$_SESSION[ 'recherche' ][ 'nb_par_page' ].'';
```

=> Construction d'une liste des résultats (affectation des variables `$_SESSION['recherche'][$i]['???']`).

--> Retour au [menu](#) <--

V) Ajout d'une demande

En mode "edition" la fonctionnalité est accessible par l'utilisateur.

=> Vérification que les champs soient non vide.

=> Recherche d'un identifiant de mission.

```
$query = 'SELECT MAX (ID_MISSION) FROM T_MISSION';
```

=> Insertion d'une nouvelle demande

```
$query = 'INSERT INTO T_MISSION( ID_MISSION, UTILISATEUR, GROUPE, DEPART, DESTINATION,  
OBJET, TYPE, TRANSPORT, ALLER_DATE, ALLER_H_DEPART, ALLER_H_ARRIVEE, RETOUR_DATE,  
RETOUR_H_DEPART, RETOUR_H_ARRIVEE, COMMENTAIRE, VALIDE) VALUES ( '.$ind.', '.$_SESSION[  
mission ][ utilisateur ].'', '.$_SESSION[ mission ][ groupe ].'', '.$_SESSION[ mission ][ depart  
].'', '.$_SESSION[ mission ][ destination ].'', '.$_SESSION[ mission ][ objet ].'', '.$_SESSION[ mission ][
```



```
type ].'",'.$_SESSION[ mission ][ transport ].'", STR_TO_DATE(\". $_SESSION[ mission ][ date_aller
].'\',\'%d/%m/%Y\'),'. $_SESSION [ mission ][ heure_dep_aller ].','. $_SESSION[ mission ][
heure_arr_aller ].', STR_TO_DATE(\". $_SESSION[ mission ][ date_retour ].'\',\'%d/%m/%Y\'),'.
$_SESSION[ mission ][ heure_dep_retour ].','. $_SESSION[ mission ][ heure_arr_retour ].','.
$_SESSION[ mission ][ commentaire ].'",'. $_SESSION[ mission ][ valide ].');
```

=> Envoie d'un mail au demandeur de mission, au responsable de service ou d'équipe et aux administratifs avec une URL pour accéder directement à la demande après identification.

--> Retour au [menu](#) <--

VI) Valider une demande

Si la mission n'est pas terminée et que le responsable souhaite valider les demandes de missions, la fonctionnalité est accessible par l'utilisateur.

=> Validation de la demande par le responsable (qui a demandé la fonctionnalité de validation).

```
$query = 'UPDATE T_MISSION SET VALIDE="1" WHERE ID_MISSION = "'. $ind. "'';
```

=> Envoie d'un mail au demandeur de mission, au responsable de service ou d'équipe et aux administratifs avec une URL pour accéder directement à la demande après identification.

--> Retour au [menu](#) <--

VII) Annuler une demande

Si la mission n'est pas déjà annulée, et n'est pas terminée, la fonctionnalité est accessible par le demandeur et par le responsable.

=> Annulation de la demande par le demandeur de mission, le responsable de service ou d'équipe.

```
$query = 'UPDATE T_MISSION SET VALIDE="-1" WHERE ID_MISSION = "'. $ind. "'';
```

=> Envoie d'un mail au demandeur de mission, au responsable de service ou d'équipe et aux administratifs avec une URL pour accéder directement à la demande après identification.

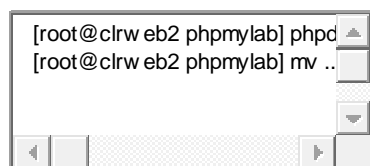
Développement

Généralités

Voici quelques indications pour aider au développement et au déploiement de l'application. Le langage de base est le PHP pour générer de l'HTML et pour accéder à la base de donnée. Le javascript est utilisé au minimum (pour ne pas ralentir l'exécution côté client) pour des effets qu'il est difficile ou impossible d'obtenir en HTML ou en CSS.

Génération de la documentation développeur

Le générateur de document utilisé est **phpDocumentor**. Le début des fichiers ".php" et quelques fonctions ont été commentées pour une meilleure interprétation par phpDocumentor.



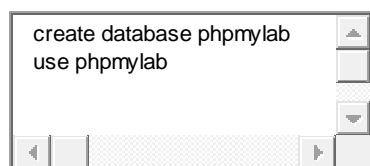
```
[root@clrw eb2 phpmylab] phpd
[root@clrw eb2 phpmylab] mv ..
```

Gestion de la base Mysql

La base est gérée au moyen de **phpMyAdmin**.

Pour exporter la base dans un fichier sql :

- Sélectionnez la base phpmylab (Server:localhost > Database phpmylab)
- Cliquez sur l'onglet "Export"
- Choisissez l'exportation SQL avec la structure et les data (ne pas sauver dans un fichier)
- Copiez/collez dans un fichier "script.sql" par exemple
- Ajoutez en début de ce fichier :

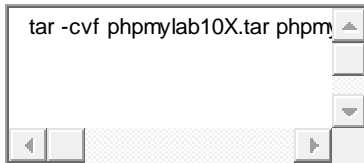


```
create database phpmylab
use phpmylab
```

Réalisation du package

Copiez dans un répertoire "phpmylab" le fichier "script.sql", la documentation HTML, la documentation développeur générée automatiquement et le code source!

Archivez le package :



Habillage

Le code CSS est dans un fichier séparé de l'HTML. Pour "reception.php" et les différents modules, la feuille de style est "style.css".

Le CSS relatif à la configuration de phpMyLab se situe dans /phpmylab/configuration/style_config.css.

Cependant, pour le fichier "administration.php", le CSS est situé dans les balises <HEAD></HEAD>.

NB: Le CSS a été codé afin que le design soit compatible sous IE 8+, Firefox 11+ (windows) / Firefox 4+ (mac), Opera 5+, Safari 5+, Chrome 18+.

```
/* Style du bouton "planning" */
#authentication input[name=planning_bouton]
{
    font-size:1.2em;
    height:35px;
    line-height:7px;
    text-align:center;
    color: #3E8187;
    background: #282828;
    background:-webkit-gradient(linear, left top, left bottom, from(#282828), to(#606060)); /* pour Chrome 10+ et Safari 5.1+ */
    background:-moz-linear-gradient(top,#282828,#606060); /* pour Firefox 3.6+ */
    background:-ms-linear-gradient(top,#282828,#606060); /* pour IE 10+ */
    background:-o-linear-gradient(top,#282828,#606060); /* pour Opera 11.10+ */
    background:linear-gradient(top,#282828,#606060); /* "vous devez toujours mettre la propriété non préfixée en dernier dans le seul but d'assurer la stabilité de votre
    -webkit-border-radius: 8px;
    -moz-border-radius: 8px;
    -ms-border-radius: 8px;
    -o-border-radius: 8px;
    border-radius: 8px;

    text-shadow:0px 1px 0px black;
    -webkit-box-shadow:0 0 5px rgba(0,0,0,0.5),
        0 -1px 0 rgba(255,255,255,0.4);
    -moz-box-shadow:0 0 5px rgba(0,0,0,0.5),
        0 -1px 0 rgba(255,255,255,0.4);
    -ms-box-shadow:0 0 5px rgba(0,0,0,0.5),
        0 -1px 0 rgba(255,255,255,0.4);
    -o-box-shadow:0 0 5px rgba(0,0,0,0.5),
        0 -1px 0 rgba(255,255,255,0.4);
    box-shadow:0 0 5px rgba(0,0,0,0.5),
        0 -1px 0 rgba(255,255,255,0.4);
    cursor: pointer;
}

/* Style lors du survol du bouton "planning" */
#authentication input[name=planning_bouton]:hover
{
    color:black;
    text-shadow:0px 1px 0px lightgrey;
    background: #62F7B2;
    background:-webkit-gradient(linear, left top, left bottom, from(#62F7B2), to(#50C991)); /* pour Chrome 10+ et Safari 5.1+ */
    background:-moz-linear-gradient(top,#62F7B2,#50C991); /* pour Firefox 3.6+ */
    background:-ms-linear-gradient(top,#62F7B2,#50C991); /* pour IE 10+ */
    background:-o-linear-gradient(top,#62F7B2,#50C991); /* pour Opera 11.10+ */
    background:linear-gradient(top,#62F7B2,#50C991); /* "vous devez toujours mettre la propriété non préfixée en dernier dans le seul but d'assurer la stabilité de votre
}

/* Style lors du clic sur le bouton "planning" */
#authentication input[name=planning_bouton]:active
{
    color:black;
    background: #50C991;
    background:-webkit-gradient(linear, left top, left bottom, from(#50C991), to(#62F7B2)); /* pour Chrome 10+ et Safari 5.1+ */
    background:-moz-linear-gradient(top,#50C991,#62F7B2); /* pour Firefox 3.6+ */
    background:-ms-linear-gradient(top,#50C991,#62F7B2); /* pour IE 10+ */
    background:-o-linear-gradient(top,#50C991,#62F7B2); /* pour Opera 11.10+ */
    background:linear-gradient(top,#50C991,#62F7B2); /* "vous devez toujours mettre la propriété non préfixée en dernier dans le seul but d'assurer la stabilité de votre
}

    -webkit-box-shadow:1px 1px 10px #3E986E inset, 0 1px 0 rgba(255,255,255,0.4);
    -moz-box-shadow:1px 1px 10px #3E986E inset, 0 1px 0 rgba(255,255,255,0.4);
    -ms-box-shadow:1px 1px 10px #3E986E inset, 0 1px 0 rgba(255,255,255,0.4);
    -o-box-shadow:1px 1px 10px #3E986E inset, 0 1px 0 rgba(255,255,255,0.4);
    box-shadow:1px 1px 10px #3E986E inset, 0 1px 0 raba(255.255.255.0.4);
}
```

Copie d'écran d'un extrait du code CSS des modules

Versions de PhpMyLab

Refonte de la documentation en ligne (<http://phpmylab.in2p3.fr>) et mise à jour selon la version stable 1.2.1

Passage à la version 1.2.1 (dernière modification le 31 mai 2012).

- **Refonte graphique** de l'interface d'administration. Le CSS est situé entre les balises <head></head> du fichier "administration.php"
- **Amélioration** Dans le module des congés, le calcul du nombre de jours ouvrés se fait automatiquement (utilisation d'AJAX). Ajout du fichier "ajax_calcNbJours.php"
- **Amélioration** Dans l'installation automatique lors de l'étape 2, il est désormais possible de se servir d'une base existante.
- **Bugs** Correction de nombreux warnings et notices pouvant apparaître lors de l'utilisation du logiciel.
- **Bugs** Correction d'un bug important dans le module congés. Le responsable d'équipe/service ne pouvait pas valider les demandes de congés.

Mise à jour de la documentation HTML en fonction de la version 1.2.0 (le 4 mai 2012).

Passage à la version 1.2.0 (dernière modification le 4 mai 2012).

- **Refonte graphique** Centralisation du CSS dans le fichier "style.css" (le design de la partie administration reste néanmoins inchangé).
- **Amélioration** Structuration des pages avec un en-tête (en_tete.php) et un pied de page (pied_page.php).
- **Amélioration** Création d'un fichier "javascript.js" qui regroupe les fonctions utilisées pour afficher l'heure, initialiser la date des calendriers (CONGES,MISSIONS) et limiter le nombre de caractères.
- **Amélioration** L'ergonomie de la page d'identification (reception.php) et des modules a été améliorée.
- **Ajout** Ajout d'un indicateur de complexité de mot de passe dans "reception.php". Création du fichier "mdp.js".
- **Bug** Correction d'un bug apparu notamment sous Mandriva, causé par l'usage de la fonction eregi() dans "calendrier_absences.php", "calendrier_missions.php" et "calendriers_conges.php". Remplacée par la fonction preg_match().
- **Amélioration** Ajout d'une procédure d'installation automatisée qui permet la génération des fichiers de configuration. Les fichiers utilisés pour l'installation sont dans le dossier "configuration/".

Version v1.1.2: Mises à jour (le 17 janvier 2012)

- **Bug** Prendre un congé sur la même journée qui commence l'après-midi et qui finit le matin n'est plus possible (conges.php L797,L810).
- **Amélioration** Gestion des chevauchements des périodes de congés lors de la demande (conges.php L746).
- **Amélioration** Impossible d'effectuer une demande de congés annuels lorsque son solde (CA) est négatif (conges.php L815).
- **Amélioration** Lors de la validation des congés, les mails d'information envoyés à son groupe précisent les demis journées (conges.php L1084).

Version v1.1.1: Mises à jour (le 06 janvier 2012)

- **Bug** Nouveau calcul des jours de congés (CA et CA-1) lors du choix du contrat (administration.php L412).
- **Bug** Correction de la fonction "réattribution des congés" lorsque le solde de l'utilisateur est négatif (administration.php L1508).

Version v1.1.0: Mises à jour importantes (modif. BD) (le 22 novembre 2011)

- **Ajout** d'un lien vers les mentions CNIL dans la page d'accueil.
- **Ajout** d'un champ dans la base de données pour permettre d'ajouter en option un deuxième responsable pour les déplacements/missions d'une équipe.
- **Amélioration** de la recherche évoluée pour les modules **CONGES** et **MISSIONS**.
- **Bug**: Les demi-journées en début et en fin de congés prises dans un week-end ou un jour férié ne sont plus décomptées.

Version v1.0.3: Mises à jour (le 22 mars 2011)

- **Ajout** de l'information des utilisateurs qui recevront par mail la demande de congé au moment de la confirmation de l'envoi.
- **Ajout** de l'information des dates dans l'objet du mail lors d'une demande de congé.
- **Remplacement** du nom du module ABSENCES par **PLANNING**.
- **Modification** de l'affichage la page d'accueil et ajout de gros boutons d'identification par module.

Mise en production du module CONGES (le 14 mars 2011)

Version v1.0.2: Mises à jour (le 26 janvier 2011).

- **Ajout** de la gestion de la quotité pour les congés (page administration.php et page conges.php). Le valideur du congé peut modifier le nombre de jours retiré à un solde.
- **Modification**: un congé "Compte épargne temps" ne peut être annulé s'il a été validé.
- **Modification**: suite à une demande de congé, le solde de récupération du demandeur apparaît dans la page administration si il est non nul.

Version v1.0.1: Mises à jour mineures (le 15 septembre 2010)

- **Ajout** de l'information des utilisateurs qui recevront par mail la demande de mission au moment de la confirmation de l'envoi.
- **Ajout** de l'information des dates dans l'objet du mail lors d'une demande de mission.
- **Modification** spécifique de la liste des suggestions des objets de la mission pour ATLAS (demande de David Calvet).

Création de la documentation HTML (le 24 août 2010)

Version v1.0: 1ère version "stable" (le 18 juin 2010)

- **Ajout** de calendriers pour le choix des dates.

Fin du développement du programme MISSIONS et création de phpMyLab (le 10 septembre 2009).

Passage à la version v1.3 (dernière modification le 22 Avril 2009)

- **Ajout** d'un calendrier sur une année dans la partie recherche (n'affiche pas les missions invalidées).
- **Modification** de la largeur de certains champs dans la partie "demande" en fonction de la taille des champs dans la base. La taille du champ "Destination" passe à 32 caractères.
- **Ajout** de 3 boutons d'aide dans la partie "demande".

Passage à la version v1.2 (dernière modification le 16 Septembre 2008)

- **Ajout** d'une fonctionnalité pour les chefs de service: accéder à toutes les missions de leurs agents.

Passage à la version v1.1 (le 28 Mars 2008)

- **Ajout** de la liste des véhicules du labo dans un menu déroulant. Quand on choisit le véhicule il vient se placer dans le champ "moyen de transport".
- **Utilisation** d'un code couleur proche du logo du LPC et du système de réservation utilisé au laboratoire sur Intranet.
- **Ajout** des entité dépensiaires (ligne budgétaire ou contrats de certaines équipes) dans la liste des équipes et des services.
- **Bug**: un chef d'équipe qui avait demandé la validation ne pouvait pas envoyer de demande de mission.
- **Bug**: un chef d'équipe ne pouvait pas lire une demande de mission lorsqu'un agent choisissait une entité dépensiaire.
- **Bug**: un chef de service ne pouvait pas lire une demande de mission lorsqu'un de ses agent partait pour le compte d'une équipe.
- **Création** du fichier "config.php".
- **Ajout** du bouton d'annulation de la saisie pour la demande.
- **Ajout** des boutons pour ouvrir la page FRAM et pour ouvrir la page du système de réservation du labo.

Version v1.0: 1ere version "stable" (27 Février 2008)