



ORGIDIAP
(LE COUTEAU SUISSE DE LA
GESTION D'UNE ENTREPRISE)

Jonathan Rey ¹ Frank Formaz ¹

IDIAP-Com 06-05

OCTOBER 2006

¹ IDIAP Research Institute, CP 592, 1920 Martigny, Switzerland, jrey@idiap.ch,
formaz@idiap.ch

IDIAP
Rue du Simplon 4
CH-1920 Martigny



Travail de Diplôme 2005 - 2006

ORGIDIAP

**Le couteau suisse
pour la gestion d'une entreprise**

Jonathan Rey
Section Informatique de Gestion
Répandant-entreprise: Frank Formaz
Répandant-école: Jean-Jacques Antille

École Supérieure de Commerce de Sierre
Chemin des Vieilles Cibles 6
3960 Sierre

Rey Jonathan
Route de Miège 21
3960 Sierre

Générique

MPC 2005-2006

ORGIDIAP : Le couteau suisse pour la gestion d'une entreprise

Présentation

Pour améliorer l'efficacité dans la gestion de l'Institut de recherche IDIAP, un outil doit être mis en place pour permettre dans un premier temps la gestion du personnel (arrivées, départs, données personnelles...), dans un deuxième temps la gestion des absences, des échéances pour les étudiants, des suivis de tâches ainsi que de futurs besoins encore non identifiés. Pour ce faire, une application modulaire centralisée se reposant sur une infrastructure linux, apache[1], mysql[2], php[3] (LAMP) va devoir être développée pour couvrir les besoins de l'institut. Un soin tout particulier doit être apporté à la sécurité des accès ainsi qu'à la simplicité d'utilisation de l'application.

Réalisation

Une application PHP utilisant une base de données MySQL a été développée. Sa conception complètement modulaire permet non seulement de couvrir les besoins initiaux, mais aussi sera à même de répondre à de nouveaux besoins.

Les modules qui ont été développés couvrent actuellement la gestion des personnes, des droits d'utilisation, des absences, du suivi des étudiants ainsi que la gestion personnelle des tâches.

Mots-clés

PHP, MySQL, PhpMyAdmin[4], ORGIDIAP, Bases de données relationnelles, gestion, modules application.

Remerciements

Merci à toutes les personnes qui m'ont aidé à mener ce projet à son terme et particulièrement à Frank Formaz, mon répondant entreprise, pour le travail et les conseils qu'il m'a donnés. Je remercie également le groupe système de l'IDIAP (Vincent Spano, Tristan Carron, Norbert et Bastien Crettol) pour leurs critiques constructives, le secrétariat (Nadine Rousseau et Sylvie Milus) qui ont fait de leur mieux pour m'expliquer leurs besoins en matière de gestion. Je remercie également l'école de m'avoir donné les connaissances techniques pour réaliser ce projet ainsi que mon répondant Jean-Jacques Antille.

Table des matières

I	ORGIDIAP	9
1	Introduction	11
1.1	L’Institut de Recherche IDIAP	11
1.1.1	Un peu d’histoire	11
1.1.2	Les trois missions de l’IDIAP	12
1.1.3	Thèmes de recherche à l’IDIAP	13
1.1.4	Finances de l’entreprise	15
1.2	Les besoins de l’institut	16
1.3	Environnement de travail	17
1.3.1	Systeme	17
1.3.2	Qu’est-ce qu’une base de données relationnelle?	17
1.3.3	Qu’appelle-t-on SQL?	19
1.3.4	Qu’est-ce que le PHP?	20
1.3.5	Outils utilisés	20
2	Description des modules	21
2.1	Personnes	21
2.2	Absences	22
2.3	Sécurité	24
2.4	Étudiants	25
2.5	Timemanager	27
2.6	Datechooser	28
3	Évolution	31
3.1	Nouveaux modules	31
3.2	Graphisme	31
3.3	Améliorations futures	31
4	Conclusions	33
4.1	Travail effectué	33
4.2	Connaissances acquises	33
4.2.1	Travail	33
4.2.2	Anglais	33
4.2.3	Connaissances scientifiques	34
4.2.4	Monde du travail	34

4.3 Déclaration sur l'honneur	34
Bibliographie	35
II Annexes	37
A Lexique	39
B Syntaxe de codage pour ORGIDIAP ?	41
C Description technique des modules	43
C.1 Personnes	43
C.2 Absences	47
C.3 Sécurité	50
C.4 Étudiants	51
C.5 Timemanager	53
C.6 Timechooser	56
D Organisation des fichiers	59

Première partie

ORGIDIAP

Chapitre 1

Introduction

1.1 L’Institut de Recherche IDIAP

1.1.1 Un peu d’histoire

Fondé en 1991 par la Fondation Dalle Molle pour la Qualité de la Vie, la Ville de Martigny, le Canton du Valais, Swisscom, l’Ecole Polytechnique Fédérale de Lausanne et l’Université de Genève, l’Institut de Recherche IDIAP, situé à Martigny, est indépendant et à but non lucratif. Juridiquement, son statut de fondation lui garantit une grande autonomie et lui donne accès aux programmes de financement publics de la recherche scientifique tout en le positionnant comme un interlocuteur neutre et crédible pour le monde académique et industriel.

A ses débuts, l’Institut comptait une dizaine de personnes actives principalement dans la reconnaissance vocale. Une première étape en 1996 a vu le retrait planifié de la Fondation Dalle Molle et l’arrivée du directeur actuel, Hervé Bourlard. Commence alors une croissance soutenue des activités de l’Institut qui l’ont amené aujourd’hui dans les premiers rangs de la recherche mondiale.

Si 2001, année du 10e anniversaire, a vu la consécration avec l’attribution du Pôle de Recherche National (PRN) IM2 (Gestion interactive et multimodale de systèmes d’information), les années 2002 à 2004 ont été marquées par une très forte croissance, symbolisée par la construction d’un second bâtiment. Il s’en est suivi une période de consolidation, aussi bien sur le plan du personnel, des infrastructures que des finances.

Cette phase de stabilisation structurelle est néanmoins marquée par des succès continus sur tous les plans : de nouveaux projets, des prix prestigieux, un nombre élevé de publications, plusieurs thèses de doctorat, de nombreux ponts jetés en direction de l’économie, enfin et surtout, le renforcement de l’effectif avec l’arrivée de plusieurs chercheurs de renommée mondiale en matière

d'intelligence artificielle.



FIG. 1.1 – Villa Tissières, local principal de l'IDIAP

L'IDIAP a également renforcé son implantation au sein du paysage académique suisse au travers d'une collaboration plus étroite avec l'EPFL. Symbolisée par la signature le 2 juin 2003 d'une convention de collaboration, elle s'est concrétisée depuis par la nomination de professeurs et la pleine adhésion de l'IDIAP et de sa trentaine de doctorants à l'Ecole Doctorale de l'EPFL.

1.1.2 Les trois missions de l'IDIAP

L'Institut reste fidèle aux missions qui lui ont été confiées, à savoir :

- la **recherche** fondamentale de très haut niveau dans les domaines de compétences de l'Institut, qui lui assure une place de leader au plan national, européen et mondial. L'Institut a un rayonnement international et collabore activement avec les principales universités mondiales ainsi qu'avec des centres de recherche reconnus.
- la **formation** de jeunes chercheurs au travers des doctorants mais aussi en attirant des universitaires de talent avant la fin de leurs études pour leur donner l'occasion de découvrir le monde de la recherche, ainsi que par des cours donnés en interne et à l'EPFL.
- la **valorisation** des résultats obtenus par le biais d'une diffusion aussi large que possible auprès des milieux scientifiques, mais aussi et surtout par des contacts étroits avec les milieux industriels pour assurer le transfert des connaissances et des technologies.

1.1.3 Thèmes de recherche à l'IDIAP

Apprentissage automatique



Développement de méthodes statistiques et d'algorithmes permettant à l'ordinateur d'apprendre à résoudre des tâches ou à reconnaître des données sur la base d'exemples préalablement fournis.

Applications : extraction et analyse de données, reconnaissance de modèles et d'échantillons de paroles, visages, lettres, musique, comportement humain.

Traitement de la parole



Traitement du signal audio, segmentation des divers types de sons (voix, musique, bruit de fond, ...), amélioration de la qualité du signal, reconnaissance vocale en temps réel et robuste au bruit.

Applications : localisation d'un locuteur, système de dictée automatique et de commande vocale, extraction et transcription automatique de documents audio, compression et transmission de la voix à bas débit.

Vision par ordinateur



Méthodes et algorithmes liés au traitement de l'image et des données multimédia, y compris la détection, le suivi et la reconnaissance de personnes, d'objets, de mouvements et d'expressions.

Applications : systèmes de localisation et de suivi de formes (vidéo surveillance, espace "intelligent"), suivi de visages sur PC et téléphones portables, analyse et segmentation de séquences vidéo, reconnaissance automatique de caractères manuscrits (services postaux, PDAs, notes manuscrites).

Gestion de contenus multimédia



Développement de systèmes d'analyse, d'indexation, d'extraction et d'archivage automatique de grandes quantités de données multimédia sur la base des résultats obtenus par la reconnaissance vocale, l'analyse de scènes et l'analyse de textes bruités.

Applications : interfaces de consultation web permettant l'indexation et la recherche de documents dans des archives télévisuelles ou radio ainsi que l'accès à des transcriptions multimédia de réunions, cours, conférences, consultations médicales.

Authentification biométrique



Mise au point de systèmes d'identification et d'authentification de personnes basés sur l'empreinte vocale, l'image du visage ou la combinaison de ces modalités.

Applications : systèmes de surveillance, systèmes de contrôle d'accès, login, sécurisation des transactions, messages vocaux.

Interaction homme-machine multimodale



Création d'interfaces multimodales associant l'information visuelle (gestes de la main, mouvements corporels, expressions du visage), auditive (parole) et le texte ; optimisation de la convivialité de ces interfaces ; analyse de l'activité cérébrale par interprétation des signaux électroencéphalographiques.

Applications : faciliter l'accès à tout type d'informations depuis tout type de terminal, contrôle d'un ordinateur ou d'une chaise roulante par une

personne handicapée, jeux vidéo.

Sources : Les informations sur l'IDIAP ont été récoltées sur le site Internet de l'Institut ainsi que dans son rapport annuel.

1.1.4 Finances de l'entreprise

Entre 2001 et 2004, le volume d'activité de l'IDIAP a triplé. L'année 2005 - avec "seulement" 5.4% d'augmentation - marque le début d'une période de consolidation comme on peut le constater dans la figure (1.2). A noter que la maîtrise des charges s'est poursuivie, permettant de renouer avec un léger bénéfice d'exploitation (investissements compris).

On peut distinguer trois types de financements :

Les fonds de base : La commune, le canton et la confédération.

Les projets : Projet USA, européens, IM2, CTI.

Le reste : Loterie, EPFL, industrie.

La commune met à disposition gratuitement les locaux pour l'IDIAP. Sans cette prestation en nature (qui représente tout de même plus de 550'000 CHF) le canton n'apporterait pas sa contribution financière de 700'000 CHF et la confédération ne donnerait rien non plus.

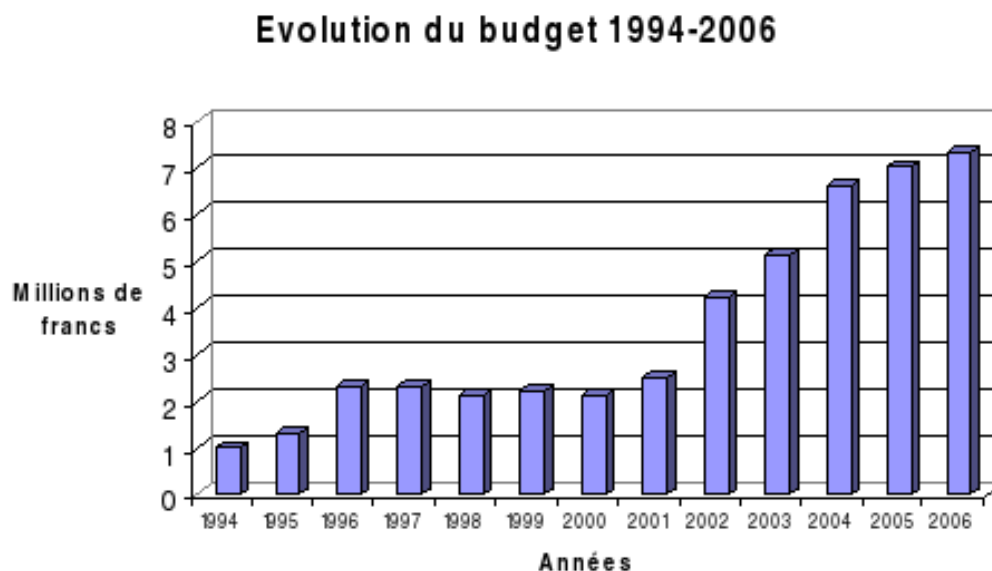


FIG. 1.2 – Finances de l'Institut de Recherche IDIAP.

Le déséquilibre entre fonds de projets et fonds de base s'est encore aggravé, atteignant en 2005 le plancher historique de 20% de fonds de base, principale-

ment à cause du retrait annoncé par la Confédération au 30 juin 2005. Cette dernière étant entre temps revenue sur sa décision et ayant assuré des montants importants pour 2006 et 2007, l'équilibre entre fonds de projets et fonds de base devrait se rapprocher à nouveau des 30% ces prochaines années.

La contribution importante de la Loterie Romande, Fr. 400'000.-, a servi à la seconde phase de l'extension des capacités de stockage. Sans ces contributions, l'IDIAP ne pourrait garantir des infrastructures informatiques de très haut niveau au personnel de l'IDIAP. Il es aussi très important de préciser que les salaires représentent plus de 70% des dépenses.

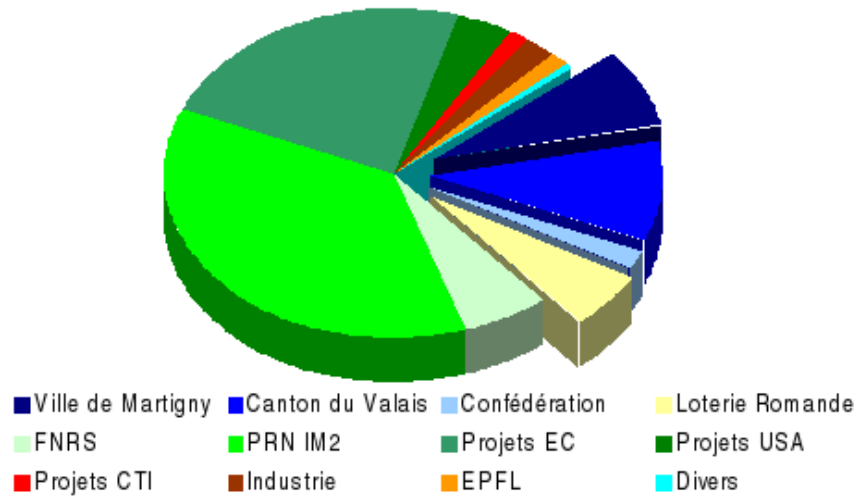


FIG. 1.3 – Participation au budget de l'IDIAP

La rubrique EPFL correspond à ce qui est versé pour que les 3 professeurs EPFL actifs à l'IDIAP viennent ponctuellement donner des cours sur leurs domaines à l'EPFL, ce chiffre représente 112'000 CHF.

1.2 Les besoins de l'institut

Ces dernières années, le personnel de l'IDIAP a augmenté de façon quasi exponentielle ce qui a eu comme conséquence d'accroître la charge du suivi administratif dans l'institut. Au fil du temps, des outils simples ont été développés au coup par coup pour permettre ce suivi mais sans liens entre eux. L'idée principale de ce projet est de pouvoir répondre aux questions suivantes : "Qui fait quoi?", "Qui part quand?", "Qui est qui?", "Qui est malade?", "Qui est en vacances?", "Qui est le chef de qui?", ...

Avant mon arrivée, des outils permettant de gérer les ordinateurs portables empruntés, la bibliothèque, l'usage des différentes salles de réunions/conférences, le planning des conférences, l'historique des achats pour les imprimantes, etc... existaient déjà. Mais ces applications n'avaient aucun lien entre elles et les

collaborateurs ne connaissaient pas leur existence, et ne savaient pas à fortiori où les trouver.

J'ai alors eu pour mission de mettre en place une plateforme unique permettant de gérer le plus de problèmes dans leur ordre d'importance. Mission que j'ai accomplie de mon mieux en élaborant dans un premier temps une application de gestion d'inventaires pour tout le matériel allant du composant informatique à l'utilisation des imprimantes en passant par les licences et les composants utilisés pour telle ou telle machine jusqu'aux logiciels et systèmes d'exploitation installés sur les machines de l'IDIAP. Dans un second temps, pour ce travail de diplôme, la plateforme ORGIDIAP, qui gère plus particulièrement les ressources humaines, a été mise en place.

1.3 Environnement de travail

1.3.1 Système

A l'IDIAP, plusieurs systèmes d'exploitation sont utilisés : l'administration travaille sous Windows tandis que les chercheurs utilisent plutôt un environnement LINUX.

1.3.2 Qu'est-ce qu'une base de données relationnelle ?

Une base de données relationnelle est en quelque sorte un gros fichier où sont stockées des données modifiables dynamiquement. L'avantage de ce système est que l'on modifie simplement les données contenues dans la base pour modifier dynamiquement des fichiers (par exemple les articles du site sont contenus dans une base de données et la page "article.php" est générée dynamiquement en fonction du numéro de l'article demandé). Conséquences : au lieu de modifier un nombre de pages égal aux nombres d'articles, je ne modifie aucune page, simplement la base de données, via des **requêtes**. Une requête est une commande qui permet d'effectuer diverses actions sur une base de données :

- Des calculs

- Une sélection de données

- Un ajout, une modification ou une suppression de données

- Une maintenance de la base (optimisation, réparation)

Schéma d'un exemple d'architecture faisant intervenir les bases de données :

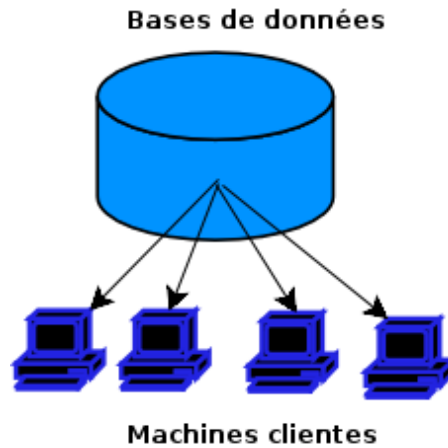


FIG. 1.4 – Exemple d’architecture d’une base relationnelle.

Dans cet exemple les machines clientes consultent le contenu de la base de données, l’affichage est ainsi toujours à jour. Une base de données contient des tables.

Qu’est ce qu’une table ?

Une table est en fait un comme un classeur sur un bureau, elle structure les différentes données qu’elle contient. Il peut y avoir plusieurs tables dans une base de données, contenant des données ayant un rapport radicalement différent entre elles.

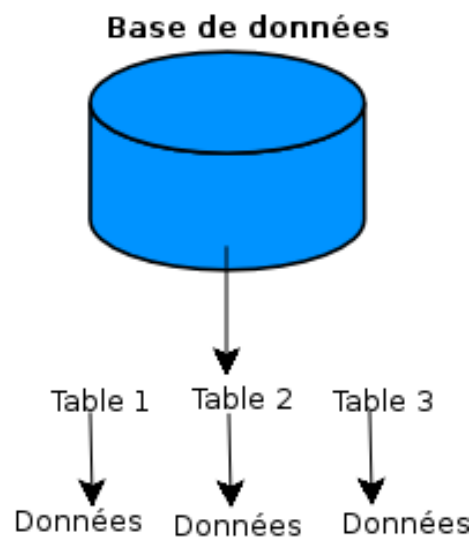


FIG. 1.5 – Fonctionnement d’une table.

1.3.3 Qu'appelle-t-on SQL ?

SQL (Structured Query Language, traduisez Langage de requêtes structuré) est un langage de définition de données (LDD, ou en anglais DDL Data Définition Language), un langage de manipulation de données (LMD, ou en anglais DML, Data Manipulation Language), et un langage de contrôle de données (LCD, ou en anglais DCL, Data Control Language), pour les bases de données relationnelles.

Le modèle relationnel a été inventé par E.F. Codd (Directeur de recherche du centre IBM de San José) en 1970, suite à quoi de nombreux langages ont fait leur apparition :

IBM Sequel (Structured English Query Language) en 1977
IBM Sequel/2
IBM System/R
IBM DB2

Ce sont ces langages qui ont donné naissance au standard SQL, normalisé en 1986 par l'ANSI pour donner SQL/86. Puis en 1989 la version SQL/89 a été approuvée. La norme SQL/92 a désormais pour nom SQL 2.

SQL est un langage de définition de données

SQL est un langage de définition de données (LDD), c'est-à-dire qu'il permet de créer des tables dans une base de données relationnelle, ainsi que d'en modifier ou d'en supprimer.

SQL est un langage de manipulation de données

SQL est un langage de manipulation de données (LMD), cela signifie qu'il permet de sélectionner, insérer, modifier ou supprimer des données dans une table d'une base de données relationnelle.

SQL est un langage de protections d'accès

Il est possible avec SQL de définir des permissions au niveau des utilisateurs d'une base de données. On parle de DCL (Data Control Language).

Typologie du langage

Il est possible d'inclure des requêtes SQL dans un programme écrit dans un autre langage (en langage PHP par exemple), ainsi que d'envoyer directement les requêtes SQL telles quelles au SGBD.

Le langage SQL n'est pas sensible à la casse (en anglais case sensitive), cela signifie que l'on peut aussi bien écrire les instructions en minuscules qu'en majuscules. Toutefois, cette insensibilité à la casse n'est que partielle dans la mesure où la différenciation entre minuscules et majuscules existe au niveau des identificateurs d'objets.¹

¹Source : www.ml.refer.org

1.3.4 Qu'est-ce que le PHP ?

Le langage PHP est un langage de programmation web côté serveur ce qui veut dire qu'un navigateur internet ne le comprend pas, c'est le serveur (la machine qui héberge le site web en question) qui va générer du code compréhensible par le navigateur. Voici un petit schéma pour illustrer le principe :

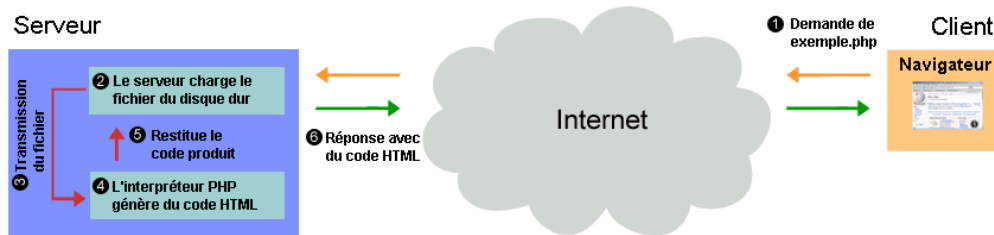


FIG. 1.6 – Principe du PHP (vu sur Wikipedia).

Son principal atout est de pouvoir communiquer avec des systèmes de gestion de bases de données tels que MySQL ou bien PostgreSQL. Dans la plupart des cas c'est lui qui traite les données de formulaires. Il a aussi l'avantage d'être libre, gratuit et simple d'utilisation et d'installation.²

1.3.5 Outils utilisés

Mon projet a été développé sur Linux, les logiciels suivant sont pour la plupart inclus dans les principales distributions :

Quanta :[5] Comme éditeur de code PHP, il est open source et simple d'utilisation.

Kile :[6] Pour écrire ce rapport, il permet de taper et compiler du \LaTeX très facilement.

phpMyAdmin : Pour la création de la base de données, la gestion des tables, la suppression d'informations et le contrôle des données.

firefox :[7] Comme navigateur principal, pour le test de mes scripts.

Database Designer for MySQL :[8] Pour faire les schémas de modélisation

Dia :[9] Pour dessiner les deux graphiques expliquant le BDD.

cogito :[10] Comme RCS (Revision Control Software)[11].

²Source : www.wikipedia.org

Chapitre 2

Description des modules

2.1 Personnes

Le module de gestion du personnel est sans doute l'application la plus utilisée par le secrétariat, car toutes les données sont nécessaires au fonctionnement des autres applications, ainsi que le site web public. Cette application existait déjà avant mon arrivée, mais je l'ai réécrite dans son intégralité car la gestion des différents formulaires n'était pas optimale ce qui m'a permis d'y ajouter tout une série de fonctionnalités.

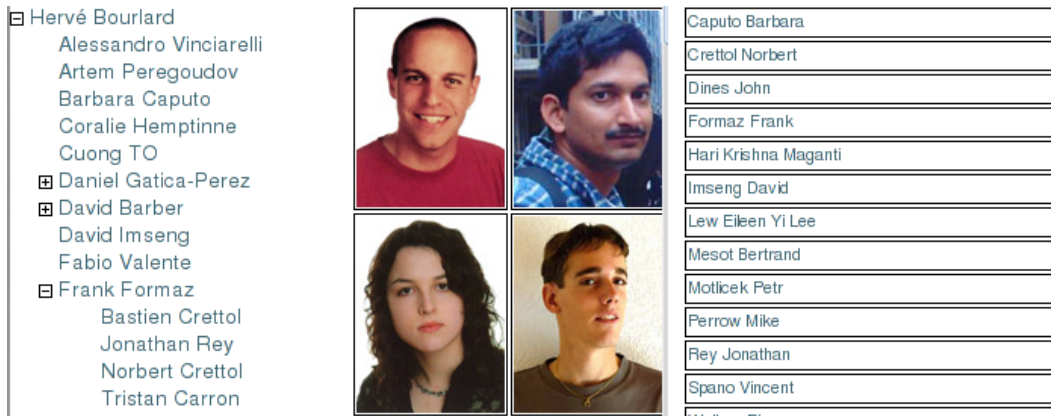


FIG. 2.1 – Différents modes de vue pour les personnes.

Dans cette partie gestion du personnel, on trouve beaucoup d'outils standards comme la gestion des arrivées, départs, numéro de téléphones internes, poste occupé, impression de cartes de visites, statistiques, tri sur les départs, mais il y a aussi des modules plus "fun" et vraiment propres à l'IDIAP comme l'impression des listes de numéros de téléphone, la consommation de café des employés ou la listes des personnes devant souhaiter un bon départ à un collègue (signer une carte de vœux), les cartes de visite. Seul le secrétariat et quelques privilégiés ont le droit d'ajouter, supprimer, modifier des données.



FIG. 2.2 – Spécimen de carte de visite généré

La majorité des listes de ce modules sont générées au format pdf pour une précision maximale lors de l'impression.

Points forts du module

Liste des numéros de téléphones, liste des personnes devant payer des cafés, liste des personnes devant signer une carte de voeux, cartes de visite IDIAP au format PDF. Gestion de plusieurs postes pour une même personne. Historique des postes occupés. Info-bulle contenant la photo, l'adresse email et le numéro de téléphone lors d'un passage avec la souris.

Difficultés rencontrées

J'ai dû écrire une classe basée sur la librairie fpdf[12] pour convertir mon code HTML en PDF et les instructions pour générer le PDF sont parfois très bizarres...

2.2 Absences

Dans toutes les entreprises, mais également dans les écoles, il est important de pouvoir rapidement retrouver pour quelle raison une personne est absente et où elle se trouve ; avant il n'existait pas d'application pour gérer ça, tout se faisait à la main sur un grand tableau affiché dans le secrétariat ; si une personne voulait savoir quand son collègue allait revenir, il fallait aller consulter le tableau. Maintenant, c'est possible de le consulter via le web, ce qui est bien plus pratique et d'y ajouter soi-même ses absences (distribution des tâches)

L'application comporte deux modes de vue, une par jours et une par mois. La vue par jour est utilisée par défaut, elle affiche les événements de toute

la journée par heures (de 8 à 19h). Quand on passe avec la souris sur un événement, la période durant laquelle la personne est absente se colorie avec une couleur spécifique. Ces différentes couleurs étaient déjà utilisées sur la version papier selon un code de couleurs prédéfini (brun pour absence injustifiée, bleu pour un voyage, noir pour l'armée, vert pour une conférence, rouge pour des vacances et orange pour un congé maladie). La base de cette application est l'interface de projet xCalendar programmée en python pour le CMS plone.

Whole-Month-Events or Events that already started but are still running.

Frank Formaz : Présentation SUN à Genève Mar-29 - Mar-29

<< previous month April 2006 next month >>>

Monday	Tuesday	Wednesday	Thursday	Friday	Saturday	Sunday
27	28	29	30	31	01	02
03	04	05 Kevin Smith : Workshop CLEAR-Southampton Apr-5 - Apr-9 Silye Ba : Workshop CLEAR-Southampton Apr-5 - Apr-9 Daniel Gatica-Perez : Workshop CLEAR-Southampton Apr-5 - Apr-7	06	07	08	09 Agnès Just : Southampton-IEEE conference Apr-9 - Apr-12 Guillaume Heusch : Southampton-IEEE conference Apr-9 - Apr-12
10 Romain Hérault : conference Apr-10 - Apr-13	11	12	13	14 Vendredi Saint/Good Friday	15	16
17 Lundi de Pâques/Easter Monday	18	19	20	21	22	23

FIG. 2.3 – Capture d'écran de la vue par mois.

Un petit calendrier permet de sélectionner la date ou le mois voulu, il existe aussi deux filtres, un pour les personnes et un autre pour les types d'absences. Par exemple, en moins de 5 clics il est possible de voir toutes les personnes ayant fait de l'armée au mois de mars 2006. Chacun est libre d'ajouter ses absences dans le système au moyen d'un formulaire. Quand une entrée est ajoutée ou modifiée, un mail est automatiquement envoyé au secrétariat pour assurer un suivi minimal.

Points forts du module

Vue immédiate des absences. Possibilité de les trier par types ou par personnes. Une absence peut être privée et dans ce cas, seul le secrétariat ou le "propriétaire" de l'absence pourrait la voir. Lors du passage avec la souris sur une absence, la zone de non-présence est coloriée avec une couleur spécifique et une info-bulle contenant tous les détails de l'absence est affichée à côté de la souris. Une absence commençant avant la zone disponible (par exemple, elle commence le 20 décembre, se termine le 20 janvier et nous affichons le mois de janvier) une table s'affiche au sommet de la page. Les jours fériés se colorient en rose.

Difficultés rencontrées

J'ai travaillé pendant plus d'une semaine sur une interface que l'on m'a par la suite conseillé d'abandonner, donc j'ai tout dû reprendre à 0. La gestion des droits n'a pas été simple car seuls le secrétariat a accès aux absences privées. J'ai aussi eu à calculer les jours fériés et pour cela, j'ai dû étudier un peu l'histoire catholique.

2.3 Sécurité

Ce module permet de sécuriser n'importe quelle page php très simplement, les login systèmes sont utilisés, les requêtes sont effectuées sur LDAP qui est un annuaire comme l'Active Directory de Windows Server.

Il n'existait aucune application de gestion de droits avant ce module, tout était géré au niveau du serveur apache avec le fichier ".htaccess". Cette page permet de naviguer à travers les fichiers du projet, en cliquant sur un d'eux, un formulaire demande qui a le droit d'y accéder. Certaines variantes sont également possibles (Masquer des informations confidentielles à certaines personnes).

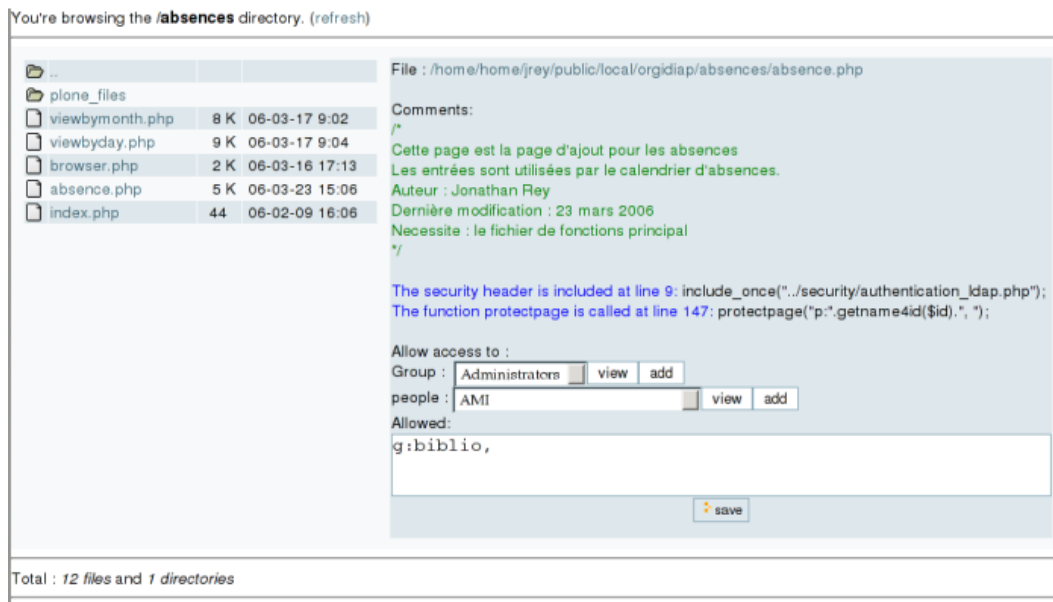


FIG. 2.4 – Capture d'écran de la page sécurité.

Points forts du module

Lecture des commentaires du fichier au niveau du système. Affichage des instructions permettant de protéger une page. Pas de table contenant les login car ils sont récupérés au niveau du système, ainsi que les groupes, les noms et les prénoms. Supporte des variantes permettant de donner une multitude de droits différents pour une même page.

Difficultés rencontrées

La gestion des groupes au niveau du système n'était pas des plus simples, car on venait de passer à LDAP et certaines informations de l'annuaire étaient lacunaires. La vérification de la validité d'un mot de passe n'a pas été simple non plus.

2.4 Étudiants

La majorité des employés de l'IDIAP sont des doctorants inscrits à l'EPFL. Dans une école, comme partout ailleurs, il y a des règles à respecter ; à l'EPFL, les dates sont importantes, par exemple l'étudiant a 30 jours à compter de son arrivée pour choisir ses crédits ou 30 jours après sa défense privée ; si l'étudiant n'a pas déposé sa thèse, elle est considérée comme nulle. Pour aider à ne pas oublier ces dates importantes j'ai dû programmer un module dans lequel on entre les dates importantes et les délais sont calculés et des mails de rappels sont envoyés aux étudiants, à leur superviseur et au secrétariat.

Messages | tâches | Aide

PhD informations

Personne :	<input type="text" value="Rey Jonathan"/>
Confirmation:	<input type="text" value="2006-04-17"/>
Arrivée:	<input type="text" value="2006-04-24"/>
Envoi doc EPFL:	<input type="text"/> (Max : 2006-05-01)
Choix crédits:	<input type="text"/> (Max : 2006-05-24)
Immatriculation:	<input type="text"/>
Thesis proposal:	<input type="text"/> (Max : Immatriculation + 1 année)
Accusé Réception EPFL:	<input type="text"/>
Prolongation:	<input type="text"/>
Délai accordé (jusqu'au):	<input type="text"/>
Défense privée:	<input type="text"/> (Max : Accusé Thesis proposal + 2 ans ou délai)
Jury de thèse:	<input type="text"/> (Max : Défense privée officielle - 7 semaines)
Envoi thèse EPFL:	<input type="text"/> (Max : Défense privée officielle - 35 jours ouvrables)
Défense publique:	<input type="text"/>
Dépôt thèse définitive:	<input type="text"/> (Max : Défense privée officielle + 30 jours)

FIG. 2.5 – Capture d'écran de la page de gestion d'étudiants.

Ce module n'a pas été le plus long à programmer, mais il a sans doute été le plus complexe, car d'une part les informations nécessaires à la création n'étaient pas des plus claires et d'une autre part, la page principale du module (index.php) compte plus de 100 conditions (if et else). Le nombre de variables est également élevé. Pour mieux gérer cette quantité de variables, elles sont appelées de manière spécifiques : Elles commencent par "c" quand elles sont calculées et par "p" quand elles sont postées.

Points forts du module

Envoi automatique de rappels. Les échéances dépassées sont affichées en rouge. Une page liste toutes les tâches à faire et les tâches faites pour les prochaines 24 heures, la semaine prochaine, le mois prochain et même l'année prochaine.

Difficultés rencontrées

Les instructions concernant les besoins n'étaient pas très claires, j'ai dû imaginer une solution pour répondre à un problème que je ne connaissais pas. Le nombre de variables et de cas possibles étant important le code n'a pas été des plus simples à gérer.

2.5 Timemanager

Le Timemanager comme son nom l'indique permet de gérer son temps afin de l'optimiser. Seul un employé utilisait un script python dans le genre, il entrait ses informations dans un fichier texte et il avait une moulinette lui permettant de calculer le nombre d'heures supplémentaires ou le pourcentage de temps passé sur un projet. J'ai eu la possibilité de prendre son script et ses fichiers textes, j'ai traduit le script python en PHP et j'ai testé avec ses fichiers texte en les uploadant dans le formulaire. J'ai par la suite créé trois formulaires pour entrer ces données via une interface.

The screenshot shows the Timemanager web interface with the following sections:

- Interface | Stats | Tasks** (top navigation)
- The timemanager will help you to manage your time** (introductory text)
- You have the choice to enter a file that will be processed or to enter the informations manually with the interface.**
- You can browse for your file and the script will process it** (file upload section with "Browse..." and "uploader" buttons)
- You can enter your infos manually with the interface** (form with fields: Date (YYYY-MM-DD), From (24:00), To (15:45), Project name, label, and an "add" button)
- You can generate your infos with the interface** (form with fields: Date (YYYY-MM-DD), Duration (24:00), Project name, label, and an "add" button)
- Long time tasks (VAC)** (form with fields: From Date (YYYY-MM-DD), To Date (YYYY-MM-DD), Project name, label, and an "add" button)
- Here are your data errors** (section for displaying errors)

FIG. 2.6 – Capture d'écran de l'interface du timemanager.

Le but de cette application n'est pas de pouvoir contrôler les employés dans leurs activités quotidiennes, mais d'avoir une idée sur le temps passé sur un projet donné, la page de statistiques présente ces informations de manière claire, avec un graphique et un tableau, la moyenne d'heures quotidiennes est calculée, ainsi que le nombre d'heures supplémentaires ou manquantes. Les statistiques sont réalisées par défaut sur toutes les données propres à la personne, mais il est possible de les afficher pour une période précise, une année, un mois, dix jours... Il y a deux formulaires permettant de choisir sa plage de dates : soit on entre deux dates et le résultat correspond à ce qui s'est passé entre ces deux dates, soit on sélectionne une année, un mois ou un jour, le résultat retourne

ce qui se trouve entre jour et jour + 1, mois et mois + 1 ou année et année + 1.

Points forts du module

Gestion des tâches soit via l'interface web, soit via un fichier texte uploadé dans le système. Possibilité de "générer" des heures en entrant dans le formulaire le nombre d'heures cumulé approximatif sur les tâches. Les données peuvent être éditées ou supprimées. Affichage des données non cohérentes au bas de la page (trop longues journées ou heures à double)

Difficultés rencontrées

La gestion des longues tâches sur plusieurs jours, car il fallait calculer les jours fériés, modifier les requêtes de sélection et traiter les données retournées dans le script. J'ai aussi eu un problème avec le graphique, je voulais faire un camembert en 3D et les chiffres étaient complètement faux et en le faisant en 2D ils étaient corrects.

2.6 Datechooser

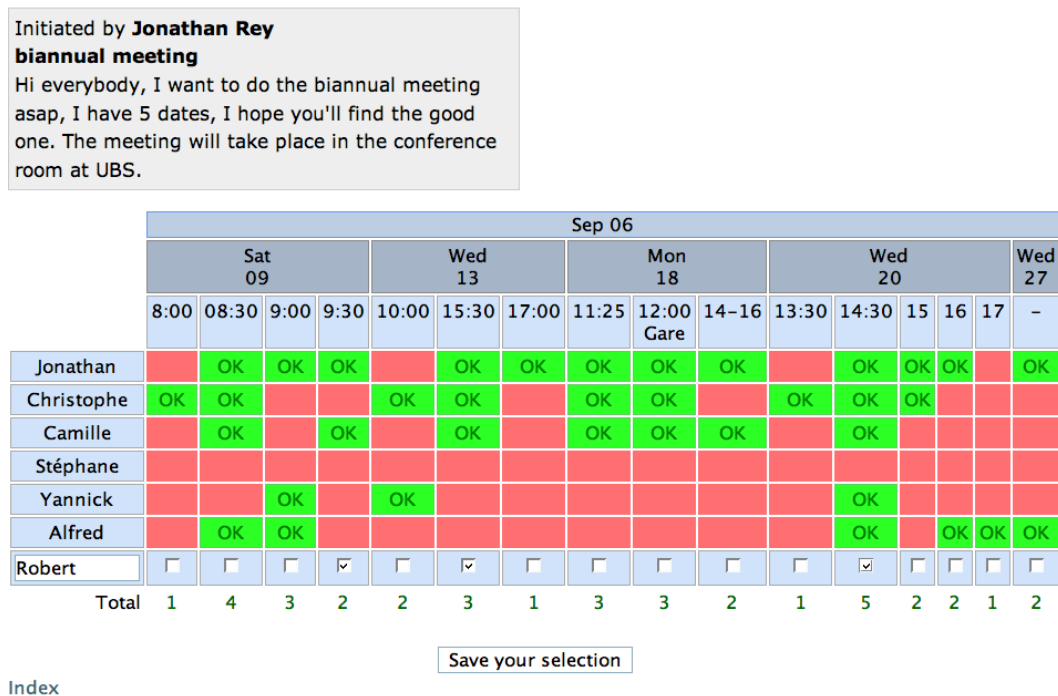


FIG. 2.7 – Capture d'écran de l'interface du datechooser.

Datechooser est une application ayant pour but de fixer une date pour un rendez-vous en fonction des disponibilités de chacun des invités. N'importe qui à l'IDIAP peut créer un nouveau poll et ainsi organiser une séance, une sortie au restaurant, un rendez-vous pour aller faire du ski, peu importe le genre de réunion et peu importe les invités (de l'IDIAP ou non)... Une fois les formulaires remplis (3 étapes : 1. Titre + description, 2. Dates, 3. Heures de disponibilité. L'étape 3 est facultative. Ce concept de poll a été inspiré du site <http://www.doodle.ch> qui offre le même service.

Points forts du module

Permet de décider d'une date de rendez-vous en tenant compte de l'agenda (chargé) de chacun des invités. Une fois le poll créé, il est possible de le modifier tant que personne n'a fait sa sélection.

Difficultés rencontrées

Aucune difficulté pour ce module, si ce n'est de faire une interface jolie qui ne ressemble pas trop à celle du site [doodle.ch](http://www.doodle.ch).

Chapitre 3

Évolution

3.1 Nouveaux modules

Les modules essentiels pour ORGIDIAP ont été créés durant mon stage en s'inspirant un peu des outils utilisés par le CERN (<http://ais.web.cern.ch/ais/-apps/default.asp>). Il est évident que ORGIDIAP n'est pas complet, d'ailleurs aucune application n'est complète, ce qui pourrait être fait d'utile sur ORGIDIAP serait un module de gestion de carnets d'adresses, on entrerait tous nos adresses et on pourrait se les partager et gérer les droits, un outil de publi-postage greffé à ce module serait également le bienvenu. Dans le même goût, un agenda que l'on pourrait partager serait utile à plusieurs personnes. Il est également possible d'utiliser des outils commerciaux à la place de créer des modules comme ça, par exemple Microsoft Exchange, mais le problème principale à l'IDIAP réside dans le fait qu'il y a plusieurs "familles" de systèmes d'exploitation et la meilleure solution reste quand même une application basée web.

3.2 Graphisme

N'étant pas graphiste, ni officiellement, ni à mes heures perdues, l'interface des modules peut laisser à désirer, il n'y a ni bandeaux, ni gros boutons colorés. Quand tout sera en place, je pense que si un graphiste passait à travers de tous mes modules pour élaborer une réelle interface graphique, ça serait vraiment plus agréable pour les yeux.

3.3 Améliorations futures

Dans un premier temps, toutes les interfaces d'administration pourraient être passées dans ORGIDIAP pour bénéficier en particulier du module de sécurité. Ces interfaces sont utilisées en particulier pour gérer le contenu du site web visible de l'IDIAP, mais il y a aussi quelques interfaces utilisées en

interne comme la gestion des bases de données pour les scientifiques (bases de données de voix, de têtes, d'images...). Dans un deuxième temps, le top du top serait de créer une console d'administration propre à l'IDIAP et d'y intégrer ORGIDIAP, Inventory et tous les outils d'administrations standards (phpmyadmin, statistiques sur les visites, utilisation de la bande passante, taille totale du site, webmail, monitoring réseau, web, DNS, mail, tutoriaux PHP, redirection d'emails... Je ne pense pas que ça donnerait beaucoup de travail car toutes ces interfaces existent déjà, le travail consisterait seulement à les rassembler et faire une interface.

Chapitre 4

Conclusions

4.1 Travail effectué

Le projet final se présente comme une console d'administration composée de plusieurs modules, chacun d'entre eux aura sa propre icône, mais pour l'instant, il n'y a que des liens sans mise en forme. Il est également possible d'en rajouter d'autres. Le site permet la saisie de nouvelles entrées dans la base telles que les nouvelles personnes arrivant à l'IDIAP, les personnes absentes, les jours fériés...

4.2 Connaissances acquises

4.2.1 Travail

Mes connaissances en programmation étant déjà plutôt bonnes à mon arrivée à l'IDIAP, j'ai eu l'occasion de les confirmer en les mettant en pratique tous les jours. Par contre, durant mon année de stage, je n'ai utilisé que des outils Linux, ce qui m'était totalement inconnu avant. J'ai également appris que faire un logiciel n'est pas forcément compliqué du côté technique, mais que satisfaire des exigences pas toujours très claires est plus difficile. Je tiens tout de même à préciser que j'ai très bien été encadré et je remercie encore Frank de l'avoir fait. Un des points les plus fastidieux est le côté documentation du travail, or, pour chacun des modules effectués, il fallait faire un fichier d'aide à l'utilisation et d'aide à la programmation.

4.2.2 Anglais

Durant mon stage, j'ai eu la chance d'avoir l'occasion de pratiquer et de perfectionner mon anglais tous les jours. Comme la majorité des employés sont d'origine étrangère, la langue utilisée pour communiquer est l'anglais.

4.2.3 Connaissances scientifiques

Comme je me trouve dans un institut de recherche, la plupart des personnes sont des scientifiques, je n'ai pas acquis de connaissances spécifiques dans un des domaines de l'intelligence artificielle, mais j'ai plutôt compris sur quoi ces personnes travaillent. Tous les mardis, un scientifique présente son travail durant environ trente minutes. Cette réunion hebdomadaire s'appelle TAM (Tuesday Afternoon Meeting). Au cours de ces réunions, j'ai surtout pu perfectionner mon anglais.

4.2.4 Monde du travail

Le monde du travail n'est pas le même que le monde des études. La chance de commencer "en douceur" m'a été donnée par l'IDIAP, car je n'ai pas eu à travailler sous la pression commerciale. Ce qui m'a le plus frappé, c'est que comme les horaires sont libres, chacun a son rythme, ses horaires. Certains travaillaient même durant la nuit.

4.3 Déclaration sur l'honneur

Je certifie, moi Jonathan Rey, être le seul auteur de cette application et ne pas avoir plagié du code provenant de tierces personnes. Je certifie avoir écrit la totalité de cette application ainsi que son rapport exceptés les pages/modules/librairies suivants qui ont été développés sous licence GPL ou simplement mis à disposition sur la toile et appartiennent à leurs auteurs respectifs.

Rekam Pour son application servant à saisir les dates via un calendrier.[13] Son applicaiton a été modifiée et sert également comme navigateur de dates dans le module absences.

Bernard Martin-Rabaud[14] Pour son application servant à saisir les couleurs au format hexadécimal. Son application est utilisée sur la page d'impression des listes téléphoniques internes.

Olivier Plathey Pour la librairie fpdf[12] servant à la génération de fichiers pdf utilisée dans plusieurs modules.

Aditus Consulting Pour la librairie jpgraph[15] servant à la génération de graphiques au format jpg à partir de code PHP.

CMS plone Pour le fichier de style et l'interface du calendrier vue par mois et vue par jours dans le module d'absences.

Bibliographie

- [1] *<http://www.apache.org/>*.
- [2] *<http://www.mysql.com/>*.
- [3] *<http://www.php.net/>*.
- [4] *<http://www.phpmyadmin.net/>*.
- [5] *<http://quanta.kdewebdev.org.net/>*.
- [6] *<http://kile.sf.net/>*.
- [7] *<http://www.mozilla.org/products/mozilla1.x/>*.
- [8] *<http://www.microolap.com/products/database/mysql-designer/>*.
- [9] *<http://www.gnome.org/projects/dia/>*.
- [10] *<http://sourceforge.net/projects/gipy/>*.
- [11] *<http://fr.wikipedia.org/wiki/RCS/>*.
- [12] *<http://www.fpdf.org/>*.
- [13] *<http://www.phpcs.com/>*.
- [14] *<http://www.javascriptfr.com/>*.
- [15] *<http://www.aditus.nu/jpgraph/>*.

Deuxième partie

Annexes

Annexe A

Lexique

PHP : Personal Hypertext Preprocessor
SQL : Structured Query Language
POSIX : Standard de systèmes d'exploitations (Linux, unix, solaris, Mac OS X)
GPL : GNU General Public License ¹
HTML : Hypertext Markup Language
CMS : Content Management System
LDAP : Lightweight Directory Access Protocol
IDHEAP : Institut De Hautes Études en Administration Publique
timestamp : Nombre de secondes écoulées depuis le 1er janvier 1970 à 00 :00 :00
CERN : Conseil Européen pour la Recherche Nucléaire
SGBD : Serveur de gestion de bases de données
BDD : Base De Données
EPFL : École polytechnique fédérale de Lausanne
LAMP : Linux, Apache, Mysql, PHP.

¹<http://www.gnu.org/licenses/gpl.html>

Annexe B

Syntaxe de codage pour ORGIDIAP ?

- Fonctions : Les fonctions doivent toutes être placées dans le fichier my-functions.php et être documentées comme toutes les autres,
- indentation : L'indentation du code est aussi très importante. Dans toutes les pages créées pour ORGIDIAP, aucune tabulation n'a été utilisée, deux espaces les ont remplacées,
- Commentaires : Les commentaires sont importants ; au début de chaque fichiers, quelques lignes de commentaires sont quasi indispensables. Pour commencer le header de commentaire il suffit de faire un slash + étoile et pour le terminer étoile + slash. Pour commenter une ligne, il suffit de mettre un double-slash et pour commenter une fonction, référez-vous au fichier de fonctions principal pour voir comment ça marche,
- Formulaire : Lorsqu'il s'agit d'enregistrer une informations dans une table, les formulaires se postent sur eux-mêmes. Pour le faire, il suffit d'écrire
`$_SERVER["PHP_SELF"]` dans le champ action du formulaire,
- Style : Pour garder le même style sobre que tous les autres modules, il suffit d'inclure le fichier style qui se trouve dans common/style, il s'agit d'une feuille de style très complète prise de plone.

Annexe C

Description technique des modules

C.1 Personnes

Le module de gestion de personnes utilise 6 tables au total.

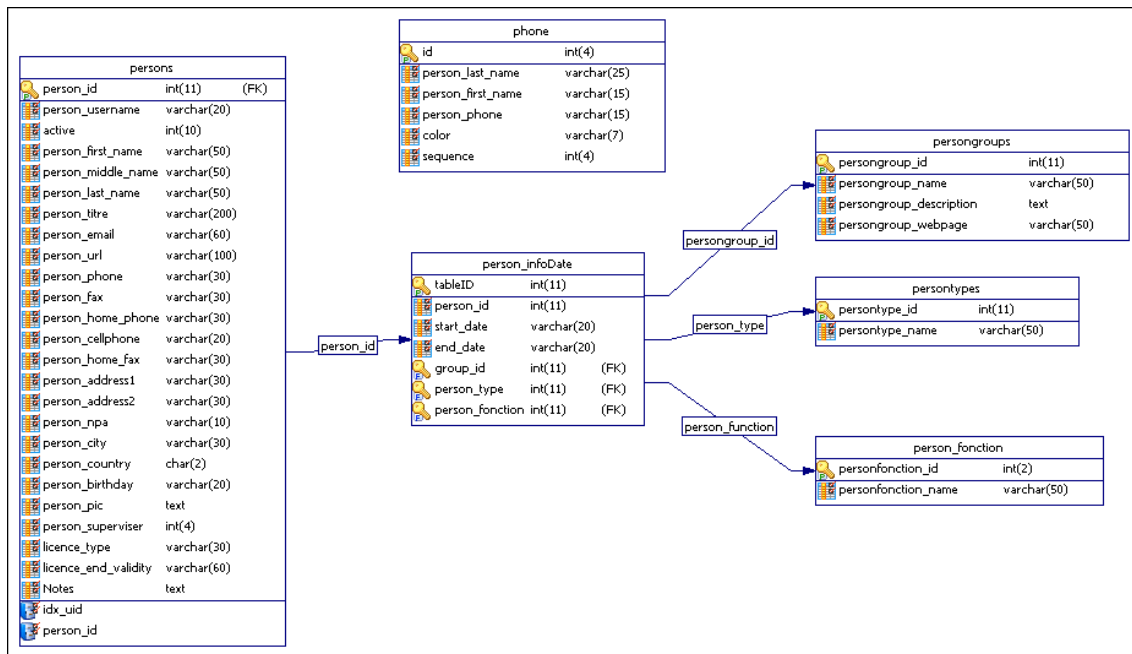


FIG. C.1 – Modélisation des tables pour le module personnes.

Description des différentes tables :

- `tbl_persons` : Table principale de l'application, elle a été en partie récupérée de l'application précédente. Cette table est utilisée par pratiquement tous les modules et par le site web également. C'est la table qui contient les informations propres à chaque employé de l'IDIAP.

Un détail spécial est à préciser : l'id du superviseur correspond à l'id de la personne se trouvant dans la même table.

person_id	247	
person_username	jrey	Login LDAP
active	1	1 pour oui, 0 pour non
person_first_name	Jonathan	
person_middle_name		
person_last_name	Rey	
person_titre		
person_email	jrey@idiap.ch	
person_url		Page Web personnelle
person_phone	741	Extension à l'IDIAP
person_fax	712	idem
person_home_phone	027/455.47.71	
person_cellphone	079/723.22.61	
person_home_fax		
person_address1	21, Route de Miège	
person_address2	NULL	
person_npa	3960	
person_city	Sierre	
person_country	ch	Code du pays
person_birthday	1987-09-06	
person_pic	jonathan_rey.jpg	
person_supervisor	81	Id du superviseur
licence_type		
licence_end_validity	0000-00-00	
Notes		

- person_infoDate : Cette table a été récupérée en intégralité de l'ancienne application. Elle contient les informations propres à la place occupée par la personne (par rapport à son id qui est lié à la table persons).

person_infoDate :		
tableID	281	
person_id	247	Rey Jonathan
start_date	2005-07-01	Entrée à l'IDIAP
end_date	0000-00-00	0000-00-00 pour indéfini
group_id	4	System Management
person_type	1	trainee
person_fonction	3	Development engineer

- phone : Cette table est utilisée pour l'impression des feuilles de numéros de téléphone, elle n'est liée à aucune des autres tables, elle ne sert qu'à

imprimer la dernière colonne de la fiche.

```

phone :
id                74
person_last_name  Bibliothèque
person_first_name
person_phone      704
color
sequence          9

```

- `persongroups` : Cette table contient les noms de groupes. Les différents groupes sont Speech Processing, Computer Vision, Machine Learning, System Management, Administration, Direction, Leave IDIAP, Information Retrieval, Biometric Authentication, Multimodal Interaction et d'autres peuvent être ajoutés à tous moments.

```

persongroup_id    2
persongroup_name  Computer Vision
persongroup_description Handwriting recognition
persongroup_webpage  computer_vision.php

```

- `persontypes` : Table qui contient les types de personnes ou plutôt les statuts qu'elles ont à l'IDIAP. Cette table a été récupérée de l'ancienne application et n'a pas été modifiée, ni dans sa structure, ni dans son contenu. Les types actuels sont `trainee`, `visitor`, `phd`, `postdoc`, `scientist`, `engineer`, `secretary`, `manager`, `director`, `friend`, `accounting`, `Unknown`.

```

persontype_id     1
persontype_name   trainee

```

- `person_fonction` : Cette table ressemble beaucoup à la `persontypes`, mais elle contient les fonctions précises occupées par la personne. Les différents types sont : `Director`, `Deputy Director`, `Development Engineer`, `Financial Manager`, `Program Manager`, `Public Relations`, `Research Assistant`... La liste est un peu longue, il n'est pas nécessaire de tous les citer.

```

personfonction_id 7
personfonction_name Research Assistant

```

Description des différentes pages utilisées par le module `people` :

- `dbPeople.php` : Ceci est la page principale, Il y a trois modes de vue possibles, soit par hiérarchie, soit par liste ou soit par photos. La vue par hiérarchie liste toutes les personnes sans superviseur (en général il n'y en a qu'une) et ensuite par récursivité, elle affiche toutes

les personnes supervisées par cette dernière. Le nombre de niveaux est illimité. Une personne peut avoir un chef supervisé par un chef qui lui est supervisé par un autre chef... En cliquant sur un nom vous pouvez modifier ses informations et afficher les personnes sous ses ordres.

La vue par noms liste toutes les personnes travaillant ou ayant travaillé à l'IDIAP, le tout est distinctement séparé en deux parties, les personnes actives et les personnes déjà formées à l'IDIAP. En cliquant sur un nom, vous allez pouvoir modifier les informations propres à cette personne.

La vue par photos est un trombinoscope, elle affiche comme la vue par noms toutes les personnes travaillant ou ayant travaillé à l'IDIAP, le tout est également séparé en deux parties. Cette page est pratique pour trouver le nom d'une personne par rapport à son visage.

- `add.php` : Page d'ajout et de modification de personnes. Un login est requis pour entrer dans cette page. Seuls les utilisateurs privilégiés ont accès à cette page et peuvent modifier les informations de tout le monde. Les droits d'accès sont gérés à partir du module sécurité. La page est composée de deux parties distinctes, la partie contenant les informations personnelles et la partie des informations propres au travail exercé à l'IDIAP par cette personne. La deuxième partie se trouve dans le fichier principal de fonctions et elle est appelée par la fonction `showWorkingInformations($idPersonne)`. Pour inscrire les dates, il suffit de cliquer sur le bouton à droite de champ ou de l'entrer manuellement.
- `phone.php` : Cette page permet de configurer (couleur et informations) la liste des numéros de téléphone interne. Les informations que l'on peut ajouter sont les numéros complémentaires pour la dernière colonne et les choix de couleurs. Il n'est pour l'instant pas possible de sauvegarder les couleurs en vue d'une utilisation ultérieure. Mais cela pourrait figurer dans la liste des améliorations possibles.
- `phone.pdf.php` : Cette page est normalement ouverte à partir de la page `phone.php`, avec les variables POST pour les couleurs des divers éléments de la fiche à imprimer.
- `depart.pdf.php` : Page "Reading list" Lors du départ d'un employé de l'IDIAP, on signe une carte de vœux et pour savoir qui a lu et qui doit lire cette carte, on imprime une liste avec deux colonnes ("à lire", "lu"); dès qu'une personne a signé la carte, elle coche la colonne "lu". Cette page génère un fichier pdf avec toutes ces informations.
- `peopleGone.php` : Page "PeopleGone" qui liste toutes les personnes formées à l'IDIAP (personnes inactives) et il est possible de les trier par nom, date de départ et type de poste occupé. Le tri sur le type de poste se fait sur l'id du poste et non pas sur le nom du poste.
- `cafe.pdf.php` : Page "coffee list". A la cafétéria de la villa Tissière, il y a une machine à café pour laquelle il faut payer 1 CHF par café consommé,

une liste est affichée au-dessus de la machine, et à chaque café pris, il faut mettre une coche dans la colonne et à la fin du mois, payer la facture. Cette page génère un fichier pdf avec une colonne nom, une pour les coches, une pour les cafés du mois précédent, une pour les cafés à payer et une pour la date à laquelle cette personne s'est acquittée. Une des améliorations possibles est la saisie des informations via une interface d'administration et l'envoi automatique de mail à la fin du mois pour la facture.

- categories.php : Page "Admin", permet de modifier les informations contenues dans les listes déroulantes, lors de la validation du formulaire, les informations sont modifiées dans une des trois tables. Il est possible d'ajouter ou modifier un type de personne, un type de fonction et un type de groupe.

C.2 Absences

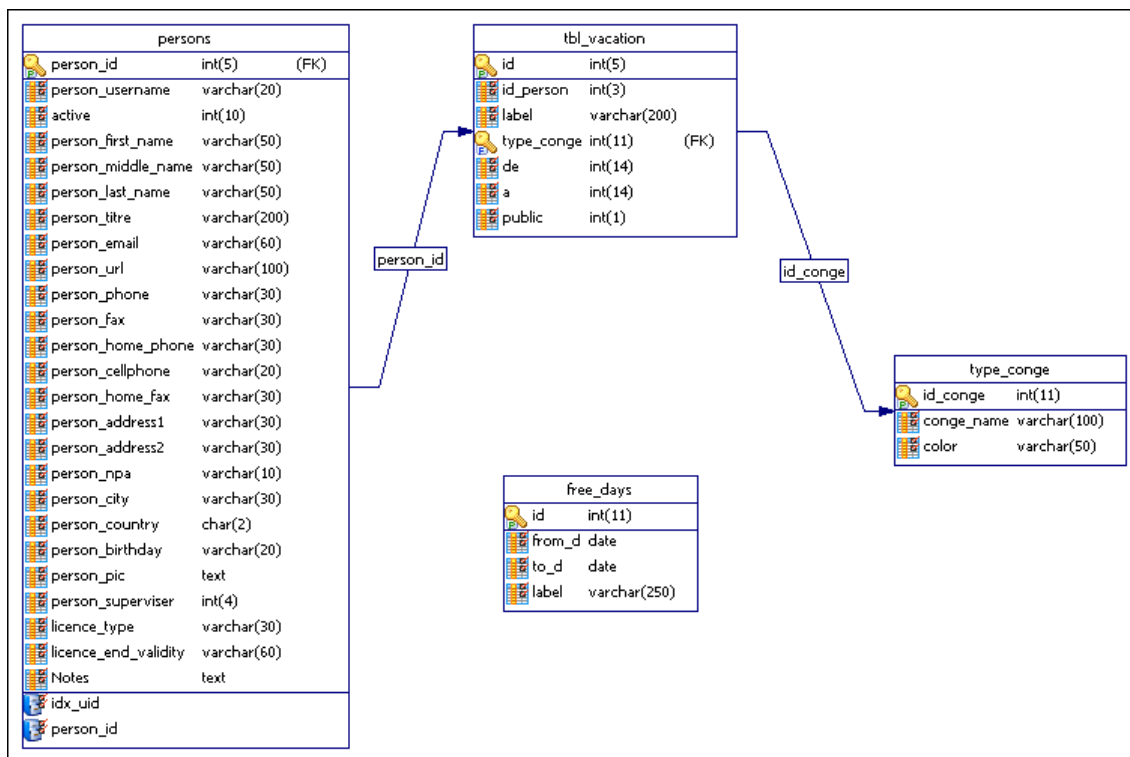


FIG. C.2 – Modélisation des tables pour les absences.

Description des tables du module d'absences :

- persons : C'est exactement la même table que celle du module people.
- tbl_vacation : C'est dans cette table que l'on stocke les absences ; pour savoir qui est absent, on garde l'id de cette personne, on précise de quel

type de vacances il s'agit (en faisant lien avec la table type_conge.

id	45	
id_person	247	Rey Jonathan
label	Recrutement	
type_conge	3	Army
de	1138347000	2006-03-07 à 08:30
a	1138383000	2006-03-08 à 18:30
public	1	1 pour OUI

- type_conge : C'est la table qui contient les différents types de congés en vigueur à l'IDIAP tels que no idea (?), traveling, army, conference, vacation, sick leave. Ces différents types ont chacun une couleur associée. Il n'existait pas d'application informatisée pour la gestion des absences, tout se faisait sur une grille sur papier.

id_conge	2	
conge_name	traveling	
color	blue	texte ou hexadécimal

- free_days : Table contenant les jours fériés elle est dans ce module utilisée pour colorier les jours fériés en rose dans le calendrier avec la vue par mois. Les jours fériés sont entrés via une page qui ne se trouve dans aucun module (feries.php).

id	1099
from_d Descending	2006-04-14
to_d	2006-04-14
label	Vendredi Saint/Good friday

J'ai passé du temps sur ce module, car j'ai travaillé pendant plus d'une semaine sur l'affichage des données et c'était vraiment mauvais, j'ai décidé de tout recommencer après une longue discussion sur la manière d'afficher ces données lors de la réunion hebdomadaire du groupe système. A la base, c'était juste un tableau tout simple avec quelques couleurs. Le choix de prendre une interface déjà existante et faite par des professionnels était plus judicieux que de réinventer la roue; il s'agit de calendrier plone. Plusieurs centaines de lignes de code en javascript sont nécessaires pour le faire fonctionner, par chance, je n'ai eu qu'à "dynamiser" l'interface qui avait été récupérée en html statique.

Ce module est composé de 3 pages :

- viewbyday.php : C'est la page de vue par jour du calendrier. Le fichier est composé de quatre parties importantes :
 1. Requête dans la table "tbl_vacation"

2. Écriture de la table du sommet.
 3. Écriture de la table des heures.
 4. Surlignage en bleu foncé des heures pendant lesquelles la personne n'est pas là.
- viewbymonth.php : C'est la page de vue par mois du calendrier. Le fichier est composé en deux parties importantes :
1. Requête dans la table "tbl_vacation"
 2. Écriture de la table du sommet et la table des heures

Les jours fériés se trouvent dans une table appelée `free_days` ; cette table est interfacée à partir du fichier `feries.php` dans "common/scripts/feries.php". La partie de l'affichage des informations a été prise d'un autre calendrier, le modèle s'appelle `xCalendar` et c'est le calendrier par défaut du Content Management System plone (<http://www.plone.org>) on l'utilise pour l'intranet, mais il a plusieurs défauts, il est fait en python, il est lent, il est compliqué à utiliser ; les informations sont stockées dans une table inaccessible à d'autres applications et il ne supporte pas les couleurs que l'on m'avait demandé de respecter dans le cahier des charges.

Privé - public : Les informations sur les absences sont par défaut accessibles par tout le monde, mais lorsque un employé décide de rendre son absence privée, seul lui et les personnes privilégiées ont accès à ces informations.

- absences.php : Cette page sert à ajouter les informations dans la base de données pour la gestion des absences. Il n'est pas possible d'ajouter une absence pour une autre personne que soi-même à moins d'être un utilisateur privilégié. Si, dans l'une des deux autres pages vous choisissez de filtrer les résultats, les filtres apparaîtront dans la page d'ajout, cela vous évitera de devoir sélectionner le type d'absences ou la personne pour laquelle l'absence doit être ajoutée. la date de début par défaut est celle sur laquelle l'utilisateur a cliqué pour arriver à cette page.

C.3 Sécurité

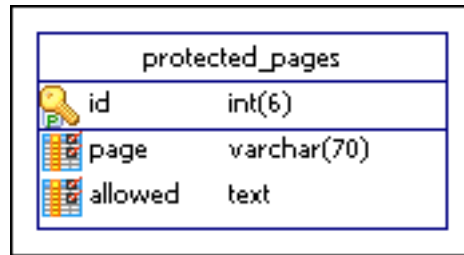


FIG. C.3 – Modélisation des tables pour la sécurité

Voilà, pour le module clé de l'application ORGIDIAP il n'y a qu'une seule table `protected_pages` :

<code>id</code>	<code>3</code>	
<code>page</code>	<code>AHVeY3xCYxhGsmUA==...</code>	<code>/databases/admin/edit.php</code>
<code>allowed</code>	<code>MQAWfxh1dkMHbiUV...</code>	<code>g:idiap, p:jrey,</code>
<code>type</code>	<code>page</code>	<code>Soit spec, soit page.</code>

Les deux champs `page` et `allowed` sont cryptés. La méthode est simple, mais efficace; il y a deux fonctions (`Crypte` et `Decrypte` dans le fichier `authentication_ldap.php`) il suffit de passer comme argument la clef principale et le texte à crypter ou décrypter comme argument lors de l'appel de la fonction. Ces données sont cryptées, car il en va de la sécurité d'ORGIDIAP, une personne mal intentionnée aurait pu par hasard accéder à la base et modifier les informations dans les champs. La chaîne de caractères décryptée du champ `allowed` est plutôt simple à comprendre : `g` signifie groupe et `p` signifie personne. Pour gérer les droits d'une page, on ajoute un groupe ou plusieurs avec une ou plusieurs personnes. Chacun des éléments est séparé par une virgule. Si une information n'est pas entrée, la page héritera des droits du dossier supérieur. Ces informations proviennent de la base LDAP et sont insérées via une interface dans la page `security.php` du module sécurité. Le champ `type` définit si la condition est valable pour une page entière ou pour des privilèges dans la page (par exemple dans le module `people`, les privilégiés ont plus de menus). Description des différentes pages utilisées par le module `people` :

- `security.php` : Page qui permet de gérer les droits de tous les autres modules d'ORGIDIAP. Cette page est un navigateur, et la racine de ce navigateur est le dossier ORGIDIAP. Il affiche comme n'importe quel navigateur une liste de fichiers ou dossiers. Celui-là n'affiche que les pages PHP et les dossiers, le reste n'est pas important. Si l'on clique sur un fichier, le navigateur affiche le header de commentaires. Pour que ces commentaires soient pris en compte, il faut impérativement qu'ils commencent par `/*` et se terminent par `*/`. Il est absolument inutile de faire

des lignes d'étoiles pour faire joli. Cette restriction est nécessaire, car pour commenter une fonction selon les règles de codage, le commentaire commence par `/**` et se termine par `*/`. Pour ne pas afficher les commentaires de fonctions, on affiche que ceux qui commencent par `/*`.

- `authentication_ldap.php` : Page principale du module de sécurité. Il n'y a pratiquement que des fonctions dans cette page.

C.4 Étudiants

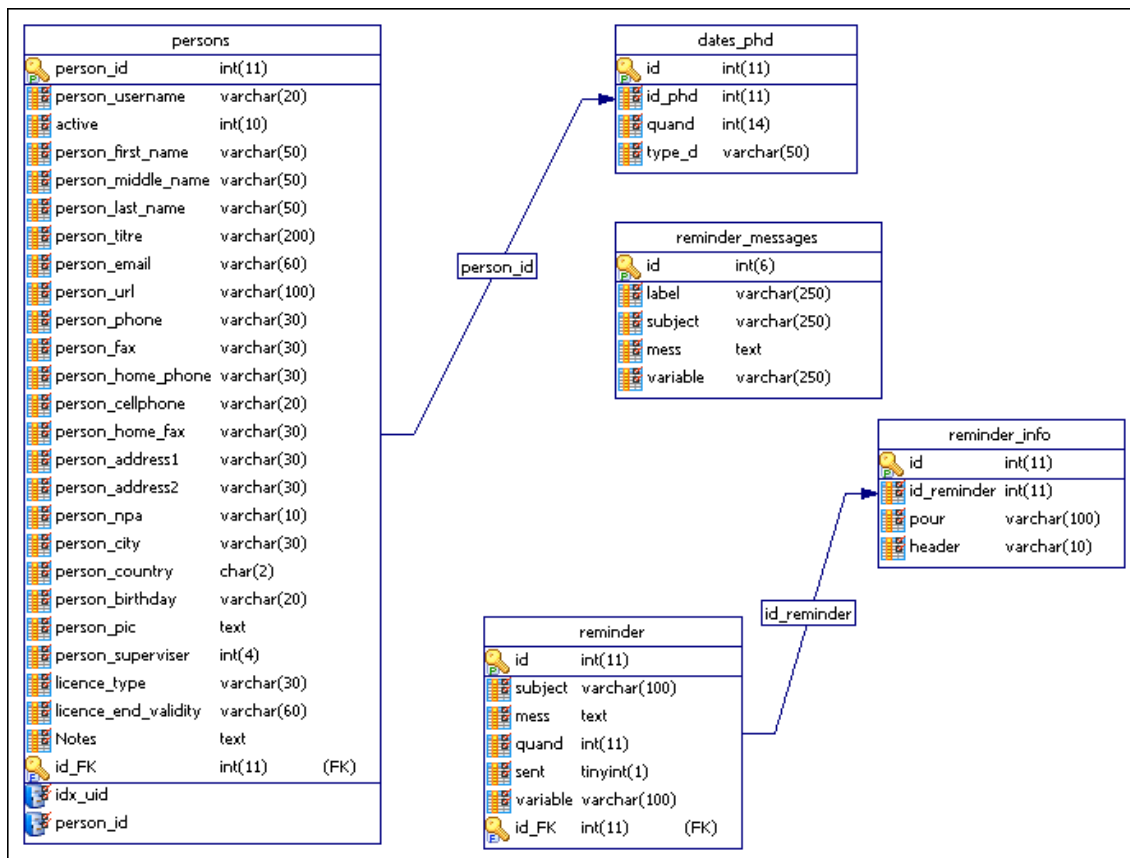


FIG. C.4 – Modélisation des tables pour le module PhD.

Le module de gestion d'échéances pour les étudiants comporte deux pages d'interface et cinq tables sont nécessaires à ce module. Ce module permet théoriquement d'éviter qu'un étudiant PhD oublie une échéance.

Description des tables du module students :

- `persons` : C'est la même table que celle du module `people`.
- `dates_phd` : Table servant à la partie graphique du module, on y stocke l'id de l'étudiant concerné, la date (timestamp) et le nom de la variable (la même que dans l'interface).

id	1	
id_phd	247	Rey Jonathan
quand	575247600	24 Mars 1988, minuit
type_d	pConfirmation	Nom de la variable

- reminder_messages : Cette table contient les emails à envoyer, chacun des messages a sa propre variable (soit confirmX ou soit mailX) où X est le numéro de variable. Le contenu de cette table est utilisée par la table reminder, les messages y sont simplement copiés. Il n’y a aucun lien entre la table reminder et la table reminder_messages parce qu’à partir du moment où un lien aurait existé, la table reminder n’aurait plus servi qu’au module étudiants, et le but est quand même de faire évoluer l’application. Il est possible d’utiliser la table de rappels pour une autre application.

id	1	
label	Remplir ...	Label pour interface
subject	Rappel : EPFL...	Sujet de l’email
mess	Bonjour, ...	Corps du message
variable	mail1	Variable du mail

- reminder : Cette table contient les rappels à envoyer avec leurs messages

id	1	
subject	Confirmation...	
mess	Bravo, ...	
quand	1001973600	2001/10/01
sent	0	0=No, 1=Yes
variable	confirm1:247	(variable:id_phd)

- teminder_info : Cette table est liée à la table reminder, elle contient l’adresse email de la personne à qui envoyer le message se trouvant dans la table reminder.

id	1227	
id_reminder	1	id table reminder.
pour	info@idiap.ch	
header		

Ce module utilise seulement trois pages :

- index.php : Formulaire d’ajout et de modification d’échéances pour les phd, cette page se présente comme un formulaire avec une liste déroulante pour sélectionner la personne et des champs dates pour insérer les échéances. Les variables commencent par "p" pour poste ou "c" pour calcul. Ce fichier se décompose en 11 étapes distinctement séparées par des com-

mentaires.

C.5 Timemanager

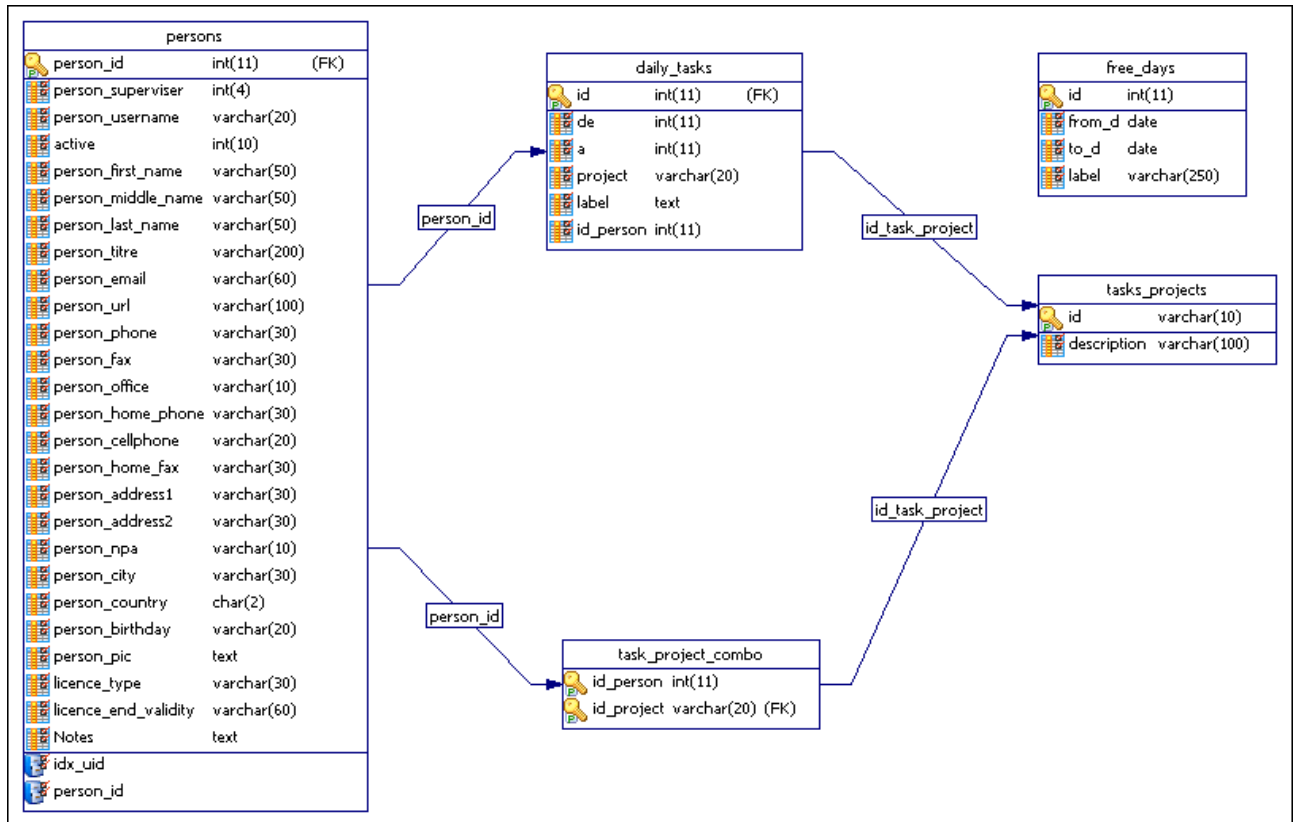


FIG. C.5 – Modélisation des tables pour le module timemanager.

Description des tables du module timemanager :

- `persons` : C'est la même table que celle du module `people`.
- `daily_tasks` : Table contenant toutes les informations sur le temps passé par projets

<code>id</code>	33	
<code>de</code>	1104737400	03/01/2005 08:30
<code>a</code>	1104743100	03/01/2005 10:05
<code>project</code>	ADM	Administration
<code>label</code>	Read email	
<code>id_person</code>	247	Rey Jonathan
- `free_days` : Table contenant tous les jours fériés, cette table est déjà décrite dans le module de gestion d'absences. Ces données sont utilisées quand la personne entre une longue période dans l'interface pour que l'application soit capable de comptabiliser le nombre de jours de travail.

- tasks_projects : Table contenant la description des acronymes.

id	ADM	
description	Administratives	tasks

- task_projects_combo : Cette table contient les tâches sur lesquelles nous travaillons, pour nous éviter à parcourir une grosse liste déroulante pour trouver notre tâches.

id_person	247	Rey Jonathan
id_project	ADM	Admin

Ce module comporte plusieurs fonctions, les données sont soit insérées dans un fichier texte qui est par la suite uploadé et traité instantanément par un script, soit pas l'interface qui offre trois différents formulaires. Les informations, qu'elles soient entrées via le formulaire ou uploadées, sont insérées dans la table et les heures sont converties en timestamp pour permettre une meilleure gestion. Il est également possible d'exporter les données dans le format initial ou d'éditer les 20 premières entrées d'une période en un seul clic.

Description des différentes pages utilisées par le module du Timemanager :

- index.php : Page principale du module, cette page traite les 4 formulaires se trouvant sur la page (le formulaire d'upload de fichiers, le formulaire pour entrer données normalement, le formulaire pour générer des horaires et le formulaire pour entrer les longues périodes comme les vacances). Les trois formulaires permettant d'entrer les données sont appelés par des fonctions, les vérifications sont effectuées sur cette page, mais une fonction insertTask est utilisée pour enregistrer les données.

Extrait d'un fichier texte à uploader :

2006.17.04

08:00-08:15 [ADM] Read email
 08:00-10:10 [ORG] Work on students module
 10:10-10:35 [COF] Coffee
 10:35-12:20 [ORG] Work on students module
 13:00-17:20 [ORG] Work on students module

...

2006.18.05-2006.23.05 [VAC] Vacances en espagne

- list.php : Page de statistiques, un graphique est un petit résumé sur le temps passé par tâches sont générés, deux différents filtres permettent de sélectionner une plage de dates. Les données propres aux heures sont confidentielles, par contre, le pourcentage passé sur une tâche, ainsi que son acronyme et le graphique sont consultables par le répondant de l'employé basé sur la même méthode que la vue hiérarchique dans le module

de gestion des personnes.

Un employé verra ces informations en consultant sa page de statistiques.

[SUMMARY] Total: 226 days or 1942.65 hours.

[SUMMARY] Mean of 8.6 hours per day.

[SUMMARY] Needed (8.40*226): 1898.4

[SUMMARY] Supp : 44.25

Tasks:

ADM : 297.3 (15.3%)

ML : 242.8 (12.5%)

SPE : 192.7 (9.92%)

CIN : 180.4 (9.29%)

OCV : 161.3 (8.3%)

...

Son chef verra ça :

Tasks:

ADM : (15.3%)

ML : (12.5%)

SPE : (9.92%)

CIN : (9.29%)

OCV : (8.3%)

...

- projects.php : Page d'ajout de tâches (acronymes), il n'est pas possible de les supprimer et si un acronyme n'a pas de description, la première personne qui le voit peut en ajouter une, après il n'est plus possible de la modifier. Cette page permet aussi de sélectionner les tâches présentes dans notre liste déroulante.
- graph_stats.php : Page générant le graphique, la librairie jpgraph est utilisée. La page est appelée simplement : graphs_stats.php?ADM=15.3&-ML=12.5&SPE=9.92... et avec une boucle le graphique est construit.

Il n'y a pas de moyen de tricher, car, pour accéder au module Timemanager, le login est indispensable.

C.6 Timechooser

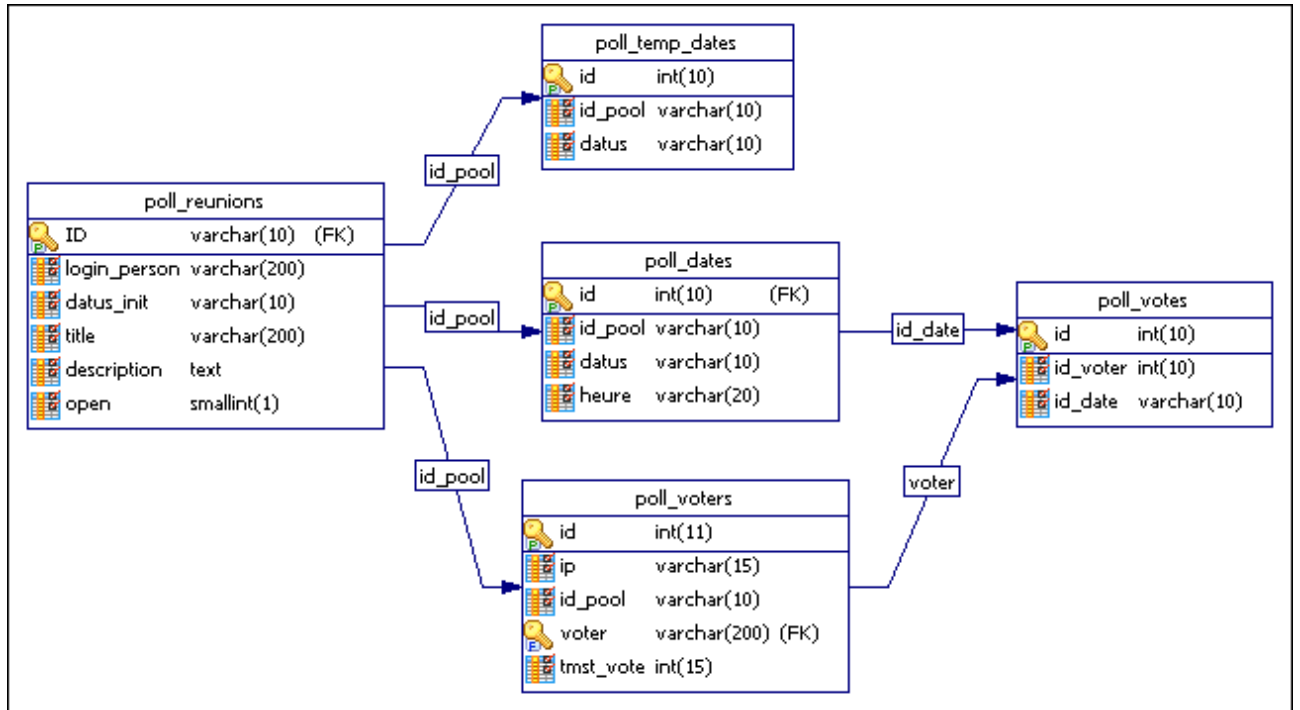


FIG. C.6 – Modélisation des tables pour le module timechooser.

Description des tables du module timemanager :

- poll_reunions : Table principale du module, elle contient les informations propre au poll (step 1 dans le formulaire). L'ID est un code de 10 lettres généré à partir d'un encodage md5 du timestamp du moment. Il n'est donc pas possible de trouver ce code au hasard.

ID	1602ceea1b	
login_person	jrey	
datus_init	2006-09-26	
title	biannual meeting	
description	Biannual IDIAP meeting...	
open		

- poll_dates : Table contenant les dates et les heures proposées par l'initiateur du poll

id	28	
id_pool	1602ceea1b	poll_reunions
datus	2006-09-09	
heure	08:30	

- poll_temp_dates : Table contenant les dates choisies à partir du calendrier à l'étape 2 du formulaire. Ces dates sont réutilisées par la suite dans

l'étape 3.

id	1	
id_pool	1602ceea1b	poll_reunions
datas	2006-09-27	

- poll_voters : Table qui contient les données propres aux voteurs pour un poll donné.

id	3
ip	192.33.221.194
id_pool	1602ceea1b
voter	Jonathan
tmst_vote	2006-09-27

- poll_votes : Table qui contient tous les votes en fonction de l'id du voteur et l'id de la date choisie.

id	1	
id_voter	2	table poll_voters
id_date	6	table poll_dates

Description des fichiers :

- admin.php : Page qui contient les formulaires de création pour les polls, cette page traite également les données des formulaires.
- index.php : Page d'accueil du module, si on est loggué les informations de nos polls sont affichées et un lien est affiché pour les éditer.
- view.php : Cette page traite les ajout et suppressions de votes, une fonction appelle le tabheader avec le résultat des votes et le formulaire de vote.

Annexe D

Organisation des fichiers

Chacun des modules de l'application a son propre dossier, par exemple le module de gestion d'absences se trouve dans le dossier "absences", le module de gestion du personnel se trouve dans le dossier "people". En plus de ces dossiers standards contenant les modules, il y a un dossier appelé "common" dans lequel on trouve une autre série de dossiers.

le dossier app

Le dossier app contient les petites applications utilisées par d'autres modules

Calend : Le calendrier, il sert de navigateur de date dans le module d'absences et il permet d'insérer facilement une date dans un champ texte dans la plupart des modules,

Palette : Palette de couleurs, il suffit de regarder l'exemple dans phone.php pour comprendre comment appeler la palette, une fonction javascript est nécessaire, cette petite application de palette n'est pas en PHP.

le dossier icons

Le dossier icons contient plusieurs icônes utilisées pas d'autres modules.



FIG. D.1 – Exemple d'icônes utilisées.

le dossier libs

Ce dossier contient les librairies utilisées par divers modules.

fpdf : librairie servant à générer les fichiers pdf à partir de code PHP. La syntaxe de codage n'est pas des plus simples, c'est pourquoi j'ai eu à développer une classe permettant de convertir le code html en instructions servant à générer le fichier pdf. Il existe une multitude de classes se basant sur cette librairie, il est possible de le télécharger sur <http://www.fpdf.org>. J'ai moi-même pris la classe `html_table` sur laquelle j'ai pas mal travaillé par la suite.

jpgraph : librairie servant à générer les graphiques au format jpg à partir de code PHP il existe une multitude d'exemples démontrant toutes les possibilités de diagrammes (Bâtons, camemberts, histogramme, polygones, nuages de points, diagramme en boîte à moustache...) Dans le projet ORGIDIAP, seuls les diagrammes en bâtons et camemberts sont utilisés.

le dossier scripts

Dossier qui contient des fichiers PHP communs comme `myfunctions.php`.

le dossier style

Dossier qui contient les divers fichiers de style utilisés par les modules.