

Projet STL

Formulaire Web de Vœux d'UE pour le Master Informatique

Jean ANAGBLA 3103825

Chafik NOUIRA 3403592

sous la supervision de : Antoine GENITRINI

Rapport de projet STL – Master 1 Informatique spécialité STL – Semestre 2 – UPMC

Mai 2016



Table des matières

1	Introduction	4
1.1	Travail demandé.....	4
1.2	Objectifs	4
1.3	Langages et technologies utilisées	5
1.4	Sources documentaires	5
2	Ancienne Version.....	5
2.1	Fonctionnalités offertes	5
2.2	Architecture.....	6
2.3	Bugs et failles détectés	7
2.4	Génération d'EDT	8
2.5	Gestion des redoublants	9
2.6	Défauts et limites de conception.....	9
3	Nouveau site	10
3.1	Bugs et failles corrigés.....	10
3.2	Scénario nominal d'utilisation du nouveau site	11
3.3	Nouvelles fonctionnalités	13
3.3.1	Introduction du parcours d'excellence.....	13
3.3.2	Vérification de l'email étudiant.....	13
3.3.3	Liens vers les pages des UE, spécialités et secrétariats	13
3.3.4	Coloration automatique des UE dans les EDT	13
3.3.5	PDF contrat pédagogique (dans l'email et en téléchargement)	13
3.3.6	Choix d'EDT idéal.....	14
3.3.7	Gestion du 2 ^{ème} semestre.....	14
3.4	Améliorations notables apportées.....	14
3.4.1	Redécoupage, redistribution des rôles et refactoring du code.....	14
3.4.2	Dynamisme, généricité et adaptabilité.....	14
3.4.3	Localité des données : gestion des coûts réseau et charge sur le serveur	16
3.4.4	Style	16
3.5	Choix d'implémentation.....	17
3.5.1	Base de données.....	17

3.5.2	Fichiers de données au lieu de tirer parti de la base de données	18
3.5.3	Poids, tri et classement des emplois du temps	18
4	Génération d'EDT	20
4.1	Principe de base.....	20
4.2	Améliorations apportées.....	21
4.2.1	Découpage et refactoring de la fonction edt()	21
4.2.2	Rôles et fonctionnement des nouvelles fonctions de génération d'EDT	21
4.2.3	Réduction de la taille du code et gain en lisibilité	23
5	Conclusion.....	24
6	Annexes.....	25



Projet STL

Formulaire Web de Vœux d'UE pour le Master Informatique

1 Introduction

1.1 Travail demandé

L'an dernier, en raison de contraintes temporelles, un site de choix d'UE du Master d'Informatique de l'UPMC a été rapidement mis en place. Il comportait un ensemble de formulaires interactifs demandant aux étudiants acceptés dans une spécialité du Master Informatique de l'UPMC de choisir un emploi du temps parmi l'ensemble de ceux possibles. Ces derniers sont générés en fonction des choix d'UE de l'étudiant, le nombre d'UE qu'il doit suivre, les contraintes d'horaires et de capacité des groupes de TD/TME ainsi que les UE obligatoires et les spécificités de chaque spécialité.

Minimaliste et comportant plusieurs bugs et failles de conception, l'ancien site nécessitait des améliorations s'étendant du côté algorithmique (au sein de la génération exhaustive des emplois du temps) jusqu'aux côtés design et ergonomie de l'application web, en passant par la qualité et la lisibilité du code, pour qu'il soit facilement maintenable dans le futur.

Pour résumer, les tâches à effectuer pour ce PSTL étaient donc les suivantes :

- Comprendre l'architecture et le fonctionnement de l'application existante.
- Nettoyage et factorisation du code.
- Amélioration de la gestion des contraintes (suivant les spécialités notamment).
- Amélioration du design.
- Nombreux tests et vérifications afin d'éliminer tout risque de bug (Application mise en place dès mai 2016).
- Déploiement sur le site du Master Informatique de l'UPMC.

1.2 Objectifs

Le principal but du projet est de pouvoir dimensionner à l'avance le nombre de groupes de TD/TME du master Informatique avant la rentrée.

Un autre objectif important est de pouvoir collecter des données sur les préférences des étudiants en leur demandant de choisir un emploi du temps idéal. Cette dernière amélioration permettra aux responsables des différentes spécialités du Master Informatique de modifier (si possible) les horaires de certains groupes de TD/TME si le besoin s'en fait sentir afin de satisfaire au mieux les besoins et les préférences des étudiants.



1.3 Langages et technologies utilisées

Les trois principaux langages utilisés pour réaliser le site sont :

- PHP comme langage de script utilisé côté serveur.
- JavaScript comme langage de script utilisé côté client.
- SQL comme langage de requête utilisé côté base de données.

Nous avons également utilisé CSS, notamment pour la coloration des UE des emplois du temps.

Quelques bibliothèques ont été nécessaires dans la réalisation du site actuel, parmi lesquelles :

- JQuery pour la modification dynamique des pages web et la gestion de l'interactivité avec l'utilisateur.
- PHPMailer : Une bibliothèque haut-niveau en PHP servant à envoyer des mails.
- FPDF : Bibliothèque PHP servant à créer des fichiers PDF contenant les informations voulues.

Nous avons aussi utilisé le Framework Ajax pour les requêtes asynchrones permettant de rafraîchir les effectifs des groupes d'UE en temps réel.

Enfin, nous utilisons MySQL comme système de gestion de bases de données.

1.4 Sources documentaires

Le site www.w3schools.com pour tout ce qui est développement web.

Les cours de l'UE **LI328 (Technologies du Web)** : <http://www-connex.lip6.fr/~denoyer/wordpress/20142015/technologies-du-web/>

Pour l'envoi de mail avec pièce jointe (PDF contenant les choix d'UE et d'emploi du temps), nous avons opté pour la bibliothèque PHPMailer qui est une bibliothèque PHP haut niveau et dont le site officiel est le suivant : <http://phpmailer.worxware.com>

Nous avons aussi opté pour la bibliothèque FPDF (*Free PDF*) pour la génération de fichiers PDF, la documentation de cette dernière se trouve sur le site : <http://fpdf.org>

2 Ancienne Version

2.1 Fonctionnalités offertes

L'ancienne version, étant développée rapidement, offre peu de fonctionnalités qui sont les suivantes :

- Choix d'UE du Semestre 1.
- Gestion d'un parcours classique formaté à 5 UE.
- Gestion des redoublants (Choix possibles entre 1 et 5 UE).
- Envoi de mail au format HTML contenant les UE et emploi du temps choisis.



2.2 Architecture

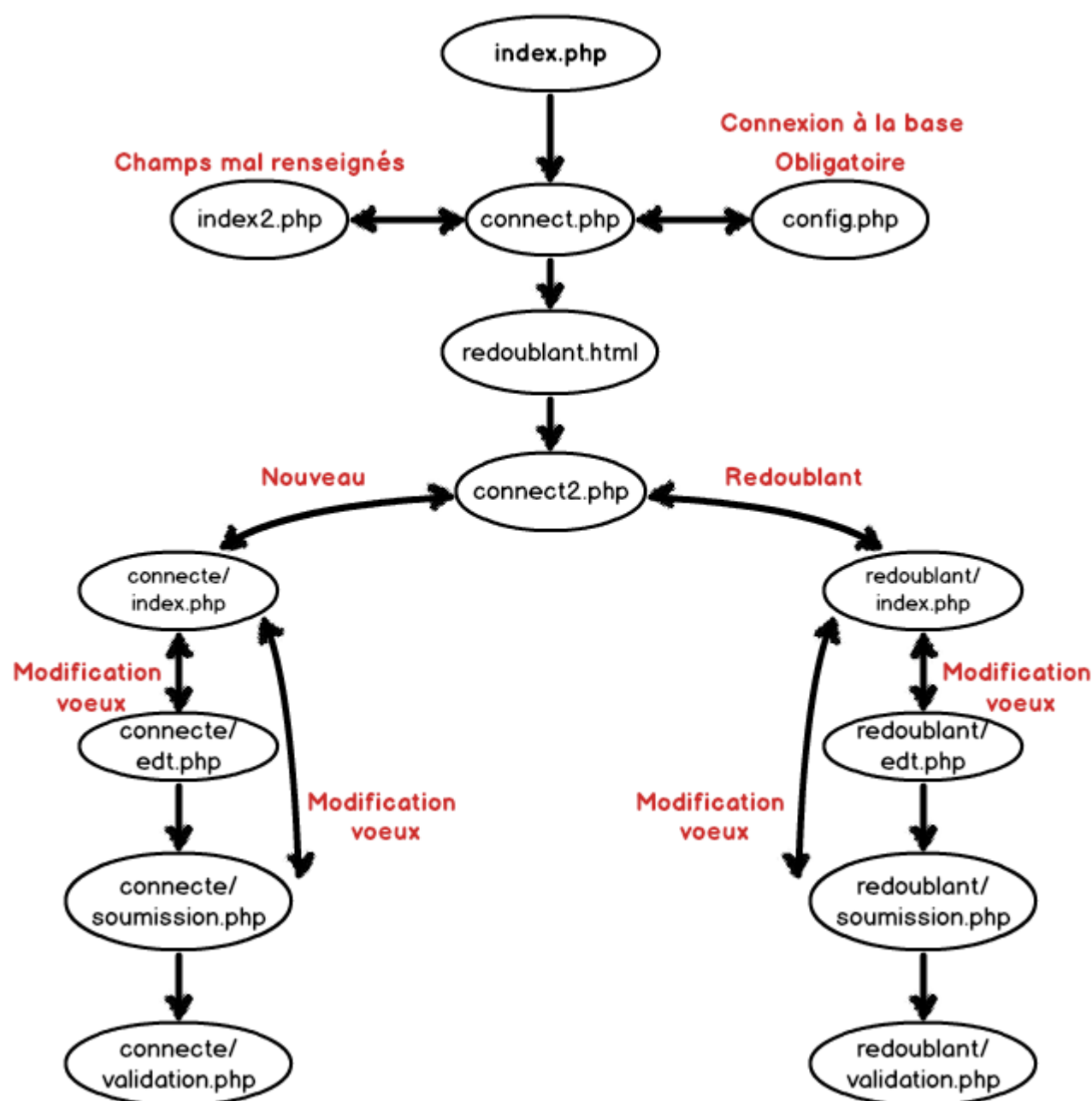


Figure 1.0 – Architecture de l'application existante

- **Scénario nominal de l'ancien site**

1. Saisie des informations de l'étudiant dans la page *index.php*.
2. Exécution du fichier *connect.php* qui vérifie dans un premier temps si les informations saisies par l'étudiant sont cohérentes avant de se connecter à la base en utilisant le fichier *config.php* afin de vérifier si l'étudiant a déjà validé ses choix et agir en conséquence. Lors de la vérification des informations saisies, si ces dernières sont incohérentes, l'étudiant est redirigé vers la page *index2.php* qui a le même rôle que la



page *index.php* sauf que cette fois-ci les champs sont pré-remplis avec les informations saisies précédemment par l'étudiant.

3. La page *redoublant.html* sert à indiquer si l'étudiant est redoublant ou pas.
4. La page *connect2.php* redirige l'étudiant vers la page correspondant à sa situation (*connecte/index.php* pour les nouveaux et *redoublant/index.php* pour les redoublants).
5. La page *index.php* (*redoublant ou connecte*) permet à l'étudiant de choisir les UE qu'il souhaiterait suivre parmi celles proposées.
6. Une fois le choix d'UE validé, la page *edt.php* sert à afficher les différents emplois du temps générés pour la combinaison d'UE correspondante. Cette page donne la possibilité de retourner en arrière afin de modifier le choix d'UE.
7. *soumission.php* affiche un récapitulatif contenant les informations de l'étudiant, la liste des UE et Groupes et l'emploi du temps qu'il a choisi. Elle donne la possibilité de retourner en arrière afin de modifier le choix d'UE.
8. Une fois les choix validés, la page *validation.php* indique à l'étudiant la fin de la procédure tout en affichant un dernier récapitulatif et en envoyant un mail à l'adresse de l'étudiant contenant ce même récapitulatif.

2.3 Bugs et failles détectés

- ***S'identifier avec des informations incohérentes (champs de formulaires invalides)***

L'ancienne version du site se contente de remplacer les caractères spéciaux et les espaces par des « underscores ». Dans la page *index.php*, lors de la soumission des informations du formulaire d'accueil par l'étudiant, une fonction JavaScript (*special_chars()*) est appelée afin de remplacer les caractères spéciaux dans le nom et prénom de l'étudiant par le caractère '_'. Ceci a pour effet de résulter un nom et prénom de la forme « ____ » si un étudiant s'amuse à ne mettre que des caractères spéciaux ou des espaces dans les champs nom et prénom. Il arrivait ainsi que la base de données contienne dans ses tables des cellules vides de sens comme la chaîne de caractères « ____ » par exemple, ou que l'impression du récapitulatif de l'inscription d'un étudiant ne contienne aucune information sur ce dernier.

- ***Numéros d'étudiant invalides***

De la même manière, l'ancienne version ne s'assure pas correctement de la validité des numéros d'étudiants saisis. Si un numéro d'étudiant renseigné est supérieur à 2000000, il est considéré comme valide.

- ***Utilisation d'une adresse mail non valide***

Des adresses email invalides ont pu être saisies par certains étudiants, qu'il était impossible de contacter par la suite. Ceci est dû au fait que le site effectue seulement une vérification du format de l'adresse mail entrée par l'étudiant en utilisant la fonction basique de PHP *filter_var(\$email, FILTER_VALIDATE_EMAIL)* et que certaines adresses mal formées sont acceptées par cette fonction.

Exemple : etudiant@upmc.fr est considérée comme une adresse mail valide.



- **Présence de « trous » dans les « id » des étudiants dans la table *ListeEtudiants***

Les « id » de la table *ListeEtudiants* représentent l'ordre d'inscription des étudiants. Une première insertion dans la base de données des informations de l'étudiant est faite au début du processus d'inscription et un numéro (id) est attribué à l'étudiant. A la fin du processus (lors de la validation des vœux) la ligne insérée en amont est complétée avec les vœux de l'étudiant. Les trous dans les id se produisent lorsqu'un étudiant décide de se déconnecter avant d'avoir validé ses vœux et qu'un second étudiant valide ses vœux avant la reconnexion de l'étudiant précédemment déconnecté. Car lorsque le premier étudiant se reconnecte pour terminer ses choix, la ligne correspondant à ses informations dans la table *ListeEtudiants* est supprimée et une nouvelle est rajoutée ensuite avec un nouvel id (l'id du dernier étudiant ayant validé ses vœux incrémenté de 1). L'ancien id attribué à l'étudiant reconnecté est ainsi perdu et constitue un trou dans les id de la table *ListeEtudiants*.

- **Possibilité de choisir n'importe quelles UE pour les redoublants**

Sur l'ancien site un nouvel étudiant a ses UE obligatoires cochées et grisées par défaut lors du choix d'UE, ce qui est la bonne chose à faire. Par contre pour un redoublant, les UE obligatoires ne sont pas cochées puisqu'elles peuvent être déjà validées. Ainsi le site ne fait que demander à l'étudiant s'il était redoublant ou non, puis en fonction de la réponse de l'étudiant, relâche toutes les contraintes sur les choix d'UE si l'étudiant est redoublant. Cela laisse une totale liberté d'action aux redoublants à qui il faut s'en remettre entièrement pour la cohérence des choix faits avec les contraintes de leur parcours. Par exemple, si un redoublant n'a validé aucune UE l'année précédente, il peut tout de même choisir 5 UE d'autres spécialités et ne pas cocher les UE obligatoires de sa spécialité, puis valider ces choix qui sont bien évidemment incohérents avec les contraintes de sa spécialité.

2.4 Génération d'EDT

Cette tâche est effectuée par la fonction ***edt(num, spe, ue1, ue2, ue3, ue4, ue5)*** faisant plus de 700 lignes et se trouvant dans le fichier ***edt.php***.

Pour commencer, elle initialise les séances des différents cours et TD/TME dans un tableau associatif ayant pour clé l'abréviation du nom de l'UE (Exemple : algav pour ALGorithmique AVancée) et pour valeur un tableau contenant l'horaire du cours dans la première case et les séances de TD/TME par paire dans les cases suivantes. L'indice d'une paire TD/TME dans le tableau correspondant à une clé n'est autre que le numéro du groupe de TD/TME de cette UE. La fonction exécute ensuite 5 boucles ***for***. Dans la première, elle rajoute les séances correspondant aux cours des UE choisies par l'étudiant, si au moins deux séances se chevauchent, aucun emploi du temps n'est possible, sinon, s'il n'y a aucun chevauchement, la première séance de TD/TME de la première UE de la liste est rajoutée à l'emploi du temps et les 4 autres boucles sont exécutées ensuite afin d'énumérer les différents combinaisons possibles de groupes, si 2 séances de TD/TME ou cours se chevauchent, l'emploi du temps est ignoré et un ***break*** est exécuté afin de passer à la combinaison suivante. Si l'emploi du temps



est valide (aucun break n'est exécuté jusqu'à la 5^{ème} boucle), le poids de ce dernier est calculé afin de déterminer la liste dans laquelle il sera ajouté.

Il y a 3 listes d'emplois du temps :

- listeEDT1 : Contient les emplois du temps de couleur VERTE, cela veut dire qu'ils ne contiennent pas de groupes de TD/TME presque ou très remplis.
- listeEDT2 : Contient les emplois du temps de couleur ORANGE, cela veut dire qu'ils contiennent certains groupes de TD/TME qui sont presque remplis.
- listeEDT3 : Contient les emplois du temps de couleur ROUGE, cela veut dire qu'ils contiennent certains groupes de TD/TME remplis.

Une fois les listes remplies, la même fonction procède à l'affichage des emplois du temps générés dans les 3 listes respectives en écrivant dans le DOM de la page courante.

2.5 Gestion des redoublants

Dans cette version, les redoublants sont gérés de la même façon que les nouveaux, à quelques détails près. En effet, certains redoublants ont moins de cinq UE à valider, or, pour utiliser la fonction de génération des emplois du temps décrite dans la section précédente, il faut 5 UE. Afin de pouvoir l'utiliser sans devoir en créer une autre, 4 UE imaginaires ont été rajoutées au tableau associatif contenant les horaires de cours et TD/TME des UE. Les UE imaginaires sont **sup1, sup2, sup3, et sup4**, et bien évidemment, les séances de leurs cours et TD/TME ne se chevauchent avec aucune séance des UE réelles, puisqu'elles ont lieu à partir de minuit (le lundi, mardi, mercredi et jeudi).

Vu que la seule utilité des UE **supX** est de pouvoir utiliser la fonction **edt(num, spe, ue1, ue2, ue3, ue4, ue5)**, elles ne sont jamais affichées dans les emplois du temps générés.

2.6 Défauts et limites de conception

- **Duplication de l'architecture**

Le code gérant les nouveaux étudiants (**répertoire 'connecte'**) est entièrement dupliqué pour gérer les redoublants (**répertoire 'redoublant'**), une tâche qui aurait pu se faire autrement puisque la seule différence entre un redoublant et un nouveau (non-redoublant) est le fait qu'un nouveau est obligé de choisir 5 UE, tandis qu'un redoublant peut choisir entre 1 et 5 UE.

- **Génération d'emplois du temps avec un nombre fixe d'UE**

La fonction de génération d'emplois du temps ne prend pas plus de cinq UE en paramètres, ce qui signifie qu'elle ne peut être utilisée pour d'autres parcours (comme le Parcours d'Excellence par exemple). De plus, cette dernière fait plus de 700 lignes de code, ce qui rend sa compréhension et son utilisation un peu difficiles.

- **Duplication de la page *index.php***

Lors de la saisie d'informations invalides par l'étudiant (ou simplement en laissant vides certains champs), ce dernier est redirigé vers la page **index2.php** afin de pouvoir renseigner correctement les informations demandées. Ce contrôle de saisie aurait pu se faire avec un



script JavaScript vérifiant les informations renseignées et affichant les messages d'erreur directement sur la page *index.php* sans avoir besoin de dupliquer cette dernière.

3 Nouveau site

3.1 Bugs et failles corrigés

- ***Champs de formulaires invalides : contrôles de saisie des formulaires***

L'ensemble des bugs liés aux champs de formulaires invalides, détectés dans l'ancienne version, a été corrigé à l'aide de patterns matching stricts. À Chaque champ de formulaire correspond maintenant une expression régulière bien précise qui vérifie la validité des saisies des étudiants. Si une saisie ne répond pas aux exigences lors des vérifications, un message d'erreur est affiché à proximité du champ invalide dans le corps de la page. Il est impossible de poursuivre l'inscription tant que tous les champs d'un formulaire ne sont pas valides.

- ***Présence de « trous » dans les « id » des étudiants dans la table ListeEtudiants***

Pour corriger ce bug il a suffi d'unifier les insertions dans la base de données à la fin du processus de choix des vœux d'un étudiant (au moment de la validation de ses vœux). En plus de corriger le bug, ce procédé présente l'avantage de réduire la charge sur le serveur (nous y reviendrons dans la suite).

- ***Possibilité de choisir n'importe quelles UE pour les redoublants***

Si lors du remplissage du formulaire d'accueil un étudiant précise qu'il est redoublant, il devra remplir un formulaire supplémentaire où il devra renseigner les UE qu'il a déjà validées. En effet, la nouvelle version du site dispose d'une page consacrée aux redoublants, leur permettant de renseigner les UE qu'ils ont déjà validées (*saisie_ues_valides.php*), avant de passer au choix des UE qu'ils souhaitent suivre. En fonction de ces renseignements, le nombre exact d'UE lui restant à valider est calculé et transmis à la page suivante (*choix_ues.php*). Les UE que l'étudiant aura désignées comme déjà validées ne lui seront plus proposées lors du choix des UE à suivre pour le semestre à venir. À noter qu'il n'est pas possible de renseigner plus de 4 UE déjà validées, ce qui n'aurait pas de sens quel que soit le parcours de l'étudiant. De plus, tout comme pour les nouveaux étudiants, un redoublant ne peut valider ses choix d'UE tant qu'ils ne respectent pas les contraintes de sa spécialité.



3.2 Scénario nominal d'utilisation du nouveau site

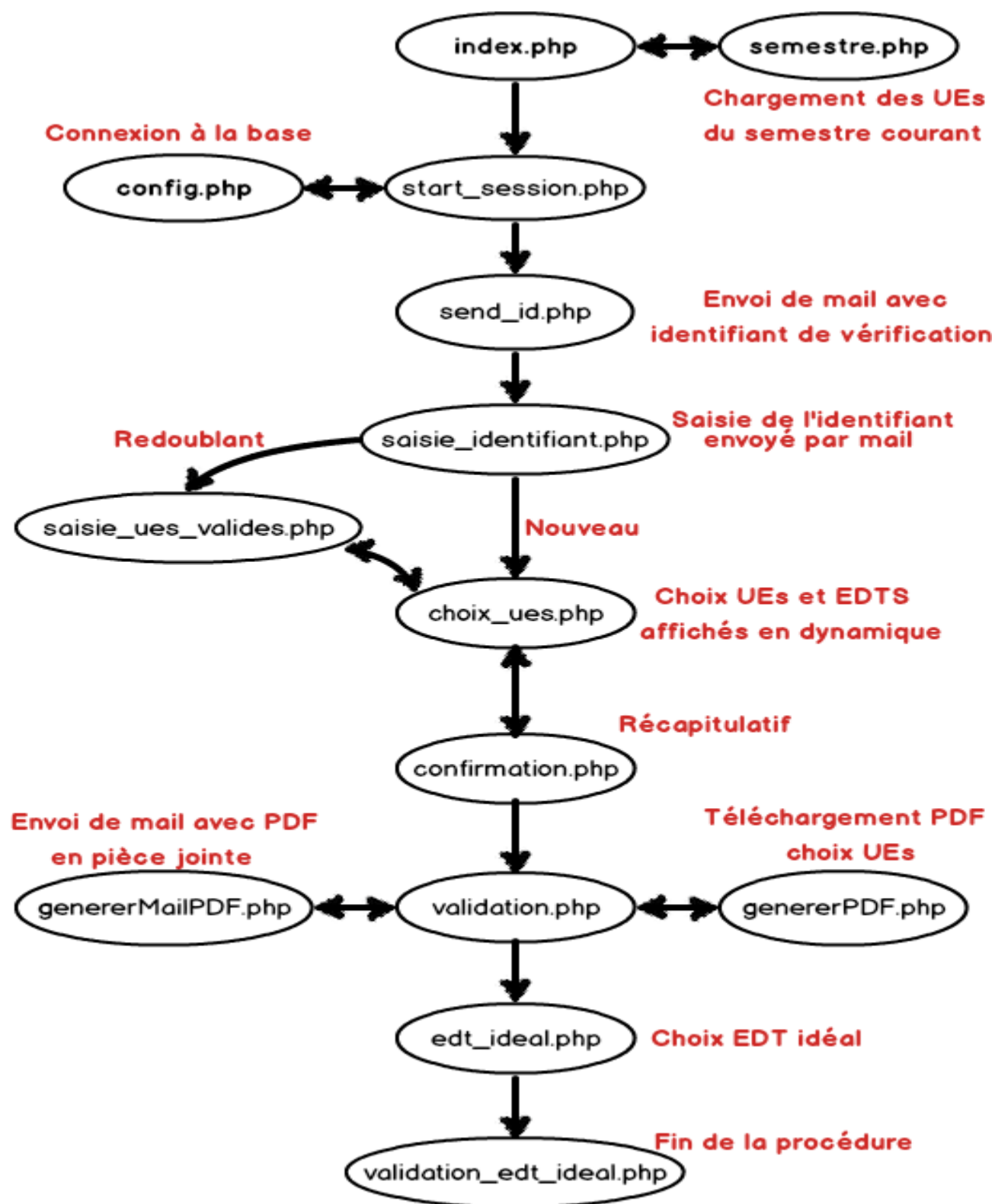


Figure 2.0 – Scénario nominal d'utilisation du nouveau site



1. La page *index.php* commence par faire un appel au fichier *semestre.php* afin de charger la liste d'UE correspondant au semestre courant.
La même page permet la saisie des informations de l'étudiant.
2. Une fois que les informations saisies sont validées, c'est la page *start_session.php* qui est exécutée, cette dernière utilise le fichier *config.php* afin de se connecter à la base de données et vérifier si l'étudiant a déjà effectué ses vœux.
3. Si l'étudiant n'a pas déjà effectué ses vœux, un mail contenant un identifiant de session lui est envoyé grâce au script de génération aléatoire d'identifiants *send_id.php* avant de le rediriger vers la page *saisie_identifiant.php* où il devra saisir l'identifiant qu'il a reçu.
4. Si l'étudiant est redoublant, une page supplémentaire lui est dédiée (*saisie_ues_valides.php*), cette dernière lui permettra de saisir les UE qu'il a déjà validées l'année précédente, ces UE n'apparaîtront pas parmi celles qu'il peut choisir pour le semestre courant.
5. La page *choix_ues.php* permettra à l'étudiant de choisir les UE qu'il souhaiterait suivre pendant le semestre courant. Une fois que l'étudiant a sélectionné le nombre d'UE maximum qu'il doit suivre, les emplois du temps sont affichés dynamiquement dans la même page.
6. Après avoir choisi l'emploi du temps à l'étape précédente, un récapitulatif des informations relatives à l'étudiant ainsi qu'à ses choix d'UE et EDT est affiché dans la page *confirmation.php*, cette dernière sert également à valider les choix.
7. Après validation par l'étudiant de l'étape précédente, les informations relatives à ses choix d'UE et d'EDT sont enregistrées dans la base de données (table *ListeEtudiants*) et un récapitulatif définitif est affiché dans la page *validation.php*. En même temps, un script PHP (*generationPDFMail.php*) est exécuté afin d'envoyer à l'étudiant un mail avec comme pièce jointe un fichier PDF contenant les informations du récapitulatif affiché. Un bouton 'Version Imprimable' sert à télécharger directement le récapitulatif au format PDF et un autre bouton 'EDT Idéal' permet de passer à la page de choix d'un emploi du temps idéal.
8. La page *edt_ideal.php* réaffichera les emplois du temps possibles pour les UE choisies par l'étudiant, ce dernier sera amené à choisir un emploi du temps idéal. Les données collectées à cette étape seront enregistrées dans la base de données (table *edt_ideal*) et permettront aux responsables des différentes spécialités de modifier (si possible) certains horaires pour satisfaire au mieux les préférences des étudiants.
9. Affichage d'un message de confirmation de fin de procédure dans la page *validation_edt_ideal.php*.

****Un étudiant peut à tout moment revenir en arrière et modifier ses choix tant qu'il ne les a pas validés dans la page confirmation.php, grâce à un bouton retour.***

****Notons qu'à chaque page PHP correspond un script javascript de même nom rendant la page localement interactive (coté client) et permettant en même temps la vérification et le contrôle des informations renseignées.***



3.3 Nouvelles fonctionnalités

3.3.1 Introduction du parcours d'excellence

Pour éviter des démarches trop longues le jour des inscriptions, le nouveau site prévoit un champ permettant de renseigner si un étudiant souhaite intégrer le parcours d'excellence de l'UPMC. Si un étudiant choisit d'intégrer le parcours d'excellence, le site l'autorisera à choisir jusqu'à 6 UE et l'aidera à trouver un emploi du temps qui lui convienne.

3.3.2 Vérification de l'email étudiant

Un simple pattern matching aurait été insuffisant pour contrôler la conformité du champ « email ». En effet, la convention de nommage sur les adresses email est très permissive et de fait une adresse email valide et conforme aux standards de l'IETF (<https://tools.ietf.org/html/rfc3696>) peut prendre de diverses formes. Pour pallier cette largesse, nous avons choisi d'envoyer aux étudiants, en plus d'une vérification simpliste via pattern matching de l'email saisi, un email de vérification contenant une clé sans laquelle il est impossible de poursuivre le processus de choix des UE. Chaque email envoyé à un étudiant contient une clé unique de vérification de 32 caractères générée en hachant le *timestamp UNIX* actuel selon l'algorithme *md5*. Cela est fait dans le but de vérifier la validité des adresses email renseignées par les étudiants et éviter à l'administration des difficultés de communication avec les étudiants le long de la période des inscriptions.

3.3.3 Liens vers les pages des UE, spécialités et secrétariats

Soucieux de faciliter le processus de choix de ses UE à un étudiant, un clic sur le nom d'une UE ouvre un onglet à l'url contenant les informations et descriptions spécifiques à l'UE. Dans le même esprit, une barre de navigation contient les liens vers les pages d'accueil de chaque spécialité et une page interne au site contenant les informations permettant de contacter chaque secrétariat.

3.3.4 Coloration automatique des UE dans les EDT

La coloration des UE se fait automatiquement grâce à un fichier CSS dédié « **ue.css** » définissant pour chaque UE un code couleur unique lui permettant de se différencier visuellement d'autres UE. À Chaque code couleur est associé un sélecteur «class» du nom de l'UE correspondante. Les UE des EDT générés sont des balises HTML de classe le nom de l'UE affichée. Ainsi la coloration est automatisée dans toutes les pages affichant les EDT obtenus par génération.

3.3.5 PDF contrat pédagogique (dans l'email et en téléchargement)

Lors de la validation des choix d'UE et d'emploi du temps par l'étudiant, un mail avec pièce jointe au format PDF contenant toutes les informations relatives à son inscription (nom, prénom, spécialité, n° étudiant, rang, UE choisies, EDT choisi) lui est envoyé. Le même fichier PDF peut être téléchargé depuis la page **validation.php** en cliquant sur le bouton 'Version Imprimable'.



3.3.6 Choix d'EDT idéal

À l'issue de la validation de ses choix d'UE et d'EDT, et après avoir téléchargé le fichier PDF contenant les informations relatives à son inscription, l'étudiant doit cliquer sur le bouton 'EDT Idéal' afin de choisir un emploi du temps qui lui semblerait idéal. Cette étape est destinée à des fins purement statistiques et permettra aux responsables des différentes spécialités du Master Informatique d'ajuster, si possible, les horaires de certains groupes de TD/TME afin de s'adapter aux besoins des étudiants.

3.3.7 Gestion du 2^{ème} semestre

L'ancien site avait été conçu uniquement pour le choix des UE du semestre 1 du M1. Le site actuel est en mesure de proposer un choix d'UE sur tous les deux semestres du M1 et pourrait même être dupliqué pour gérer le choix des UE du M2.

3.4 Améliorations notables apportées

3.4.1 Redécoupage, redistribution des rôles et refactoring du code

Le code de l'ancienne version était très condensé et peu organisé, certaines fonctions faisant plus de 700 lignes de code, l'ancienne version comportait beaucoup de redondances et de duplication de code. Une part non négligeable de notre travail a été de découper ces grosses fonctions en de plus petites et de les répartir sur plusieurs fichiers représentant des rôles bien distincts, puis pour certaines fonctions de les réécrire sans en changer le sens, ni le fonctionnement.

3.4.2 Dynamisme, généricité et adaptabilité

L'une des spécificités du nouveau site est qu'il est très dynamique. En effet, chaque page réagit et se modifie dynamiquement en réponse aux actions de l'utilisateur ou en fonction de certains paramètres temporels :

- ***What Time is it ?***

À chaque connexion d'un utilisateur, la page d'accueil vérifie la date courante et détermine le semestre en cours, ce qui aura pour effet de proposer automatiquement à l'étudiant, un choix d'UE pertinent sans intervention humaine entre les semestres.

- ***Modification dynamique du DOM pour l'affichage des messages d'erreur***

Les messages d'erreur sont incrustés dans le corps des pages web sans générer de fenêtres bloquantes (comme le faisait l'ancienne version).

- ***Vérification dynamique des choix d'UE***

Lors du remplissage du formulaire d'accueil, il est demandé à l'étudiant de préciser s'il est redoublant ou non et s'il souhaite effectuer un parcours magistère ou non (*à noter qu'un*



redoublant ne peut pas postuler au parcours d'excellence). À partir de ces renseignements, un nombre maximal d'UE pouvant être choisies par l'étudiant est automatiquement défini. Le nombre d'UE choisies est contrôlé en temps réel : à chaque clic sur une UE (*sélection ou désélection*) le nombre d'UE choisies est mis à jour. Lorsque le nombre maximal d'UE pouvant être choisies par l'étudiant est atteint, et que le choix d'UE est cohérent avec les contraintes de la spécialité de l'étudiant, les autres UE sont grisées et il est impossible de sélectionner une UE supplémentaire sans avoir désélectionné au moins une UE déjà choisie. Si les choix ne sont pas cohérents, un message d'erreur est affiché dans le corps de la page afin d'avertir l'étudiant de l'incohérence de ses choix d'UE.

- ***Génération dynamique d'emplois du temps***

À chaque sélection de sa $n^{\text{ième}}$ UE par un étudiant, les emplois du temps disponibles en fonction des UE choisies par l'étudiant sont générés et affichés dynamiquement à la suite de la page de choix des UE sans rechargement ni redirection vers une nouvelle page. De plus, les indications (instructions sur le choix d'UE) aux étudiants sont dynamiquement modifiées.

- ***Bouton « Valider » intelligent***

Afin d'éviter à l'étudiant de devoir défiler jusqu'en bas de la page pour valider son choix d'emploi du temps, nous avons choisi de rendre le bouton « **valider** » amovible. Celui-ci se déplace d'emploi du temps en emploi du temps au gré des choix d'un étudiant.

- ***Mise à jour en temps réel des places disponibles dans les groupes d'UE***

À chaque sélection de sa $n^{\text{ième}}$ UE par un étudiant (où n représente le nombre d'UE que l'étudiant a le droit de choisir), une requête Ajax est envoyée au serveur pour récupérer l'état actuel de remplissages des groupes des UE choisies par l'étudiant. Ceci est fait dans le but de trier et de colorer en temps réel les emplois du temps selon leurs taux de remplissage au moment du choix des UE : les emplois du temps avec un fort taux de remplissage seront mis au plus bas de la liste des emplois du temps disponibles, à l'inverse ceux avec un faible taux seront propulsés en haut de la liste. Dans le même temps, si un emploi du temps contient un groupe trop rempli (effectif supérieurs à 25 étudiants préinscrits), cet emploi du temps est coloré en rouge pour indiquer qu'il est risqué de le choisir. Une note en début de page explique à l'étudiant le sens des couleurs associées aux emplois du temps.

- ***Encapsulation des données et traitements génériques***

Le nouveau site a été conçu dans le but de s'adapter facilement aux changements divers notamment aux changements de noms ou horaires des UE, changement des informations administratives concernant les secrétariats, etc. et de fait de nécessiter à l'avenir peu de modifications du code source. Il est ainsi par exemple possible et aisé de modifier sans avoir à toucher au code source des fonctions qui les traitent, les fichiers de données suivants :

- **Calendrier.js** définissant les UE de chaque spécialité et leurs horaires.
- **Semestre.php** définissant les UE obligatoires, recommandées et libre de chaque spécialité pour chaque semestre, ainsi que la liste des spécialités existantes.
- **MSN.php** définissant les informations de contact des secrétariats.



- **Gestion automatique de nouvelles contraintes**

Une fois le site achevé pour le semestre 1, il nous a été confié de nous charger du semestre 2. Cependant le semestre 2 comportait des contraintes spécifiques à chaque spécialité et non prises en compte par l'architecture initiale du site. De plus, notre encadrant prévoyait d'intégrer des contraintes imprévues provenant des responsables de chaque spécialité. Afin de sauvegarder le caractère générique du site, nous avons choisi de ne pas modifier l'architecture existante mais plutôt d'ajouter un fichier **sem2constraints.js** qui contient des fonctions de vérification appelées à chaque validation d'un choix d'emploi du temps par un étudiant. Ces fonctions sont de la forme **checkConstraintName(choix)** où choix représente l'emploi du temps choisi par un étudiant. Il est possible d'ajouter ou d'enlever des contraintes dans le fichier de contraintes sans entraver le fonctionnement global du site. Si lors d'un choix d'UE par un étudiant certaines contraintes ne sont pas respectées, un message d'erreur sera affiché. Ce procédé s'est inspiré de l'une des méthodes de conception d'un logiciel sûrs étudiées dans l'UE CPS : **la conception par contrat**.

Par exemple au second semestre, un étudiant de la spécialité STL doit choisir en plus de l'UE Projet STL, au moins 3 UE de sa spécialité parmi les 5 proposées (en tenant compte des UE éventuellement validées par l'étudiant).

Exemple : Un étudiant en STL sans UE validées ayant choisi PC2R, CPS, SRCS et IHM se verra afficher un message d'erreur indiquant le non-respect de la contrainte ci-dessus.

3.4.3 Localité des données : gestion des coûts réseau et charge sur le serveur

Le site web du choix des UE a été pensé comme une application locale, et ce dans le but de réduire l'utilisation des ressources réseau de l'étudiant en même temps que la charge sur le serveur web hébergeant le site. Chaque chargement d'une page contient tous les fichiers nécessaires pour traiter et stocker les données relatives aux étudiants et leurs choix d'UE sur la machine de l'utilisateur. Grâce à JavaScript et à l'utilisation de cookies de session, un étudiant peut choisir son emploi du temps en interagissant avec les pages web sans transmettre de données au serveur web (sauf rafraîchissement du remplissage des groupes).

Chaque donnée générée par l'utilisateur du site est stockée localement sur la machine du « client » dans des variables de session. À la fin du processus de choix des UE à suivre, toutes les données nécessaires à l'enregistrement de l'étudiant et de ses choix sont transmises en une fois grâce à des requêtes SQL à la base de données du site.

3.4.4 Style

Afin d'améliorer l'aspect visuel du site ainsi que l'organisation des éléments des pages, nous avons opté pour Bootstrap qui est un framework CSS créé par les développeurs de **Twitter** et comportant plusieurs styles et éléments complémentaires (boutons esthétiques, barres de navigations, etc.). En plus d'améliorer l'affichage, ce framework est *cross-browser*, c'est-à-dire que la présentation d'une page est similaire quel que soit le navigateur utilisé. Il prend également en compte la contrainte de tailles d'écrans très variées liés aux différents types



d'appareils utilisés (Smartphones, tablettes, ordinateurs...). Un dernier avantage de ce framework est qu'il est « côté client », ce qui veut dire que les fichiers liés à Bootstrap seront chargés et exécutés sur la machine du client, ce qui réduira la charge sur le serveur.

3.5 Choix d'implémentation

3.5.1 Base de données

Pour la nouvelle version, nous avons utilisé la base de données de l'application existante en lui apportant quelques modifications.

Rôles des différentes tables et champs :

- **ListeEtudiants**

C'est la table contenant les informations des étudiants (numéro, nom, prénom, mail, spécialité) ainsi que les informations relatives à leurs choix d'UE. Un champ 'ueX' (avec X le numéro de l'UE), contient l'abréviation du nom de l'UE (Exemple : algav pour ALGorithmique AVancée) et un champ ueXgpe (avec X le numéro de l'UE) contient le numéro de groupe de TD/TME correspondant à l'UE n°X.

- **ANDROIDE, BIM, DAC, STL, etc.**

Ces tables représentent les différentes spécialités et sont toutes de la même forme, elles contiennent 2 champs :

- numetu : Numéro de l'étudiant.
- rang : Rang de passage lors de la journée des inscriptions pédagogiques.

- **Master**

Cette table contient les mêmes champs que les tables des différentes spécialités décrites ci-dessus : numéros des étudiants de toutes les spécialités confondues ainsi que leurs rangs (ordre d'inscription sur le site). Elle sert entre autres à effectuer des calculs statistiques servant aux administrateurs du site.

- **UEGroupes**

Cette table contient les groupes de TD/TME des différentes UE ainsi que le nombre d'étudiants qui les ont choisis (effectifs de chaque groupe).

Les modifications apportées à la base de données sont les suivantes:

- Ajout des champs ue6 et ue6gpe dans la table *ListeEtudiants* afin de gérer les étudiants suivant le parcours d'excellence.
- Ajout de la table *edt_ideal* afin de stocker les informations relatives aux choix des emplois du temps idéaux par les étudiants (Ses champs ont les mêmes rôles que ceux de la table *ListeEtudiants*).

Ci-dessous une représentation graphique de la structure de la base de données.



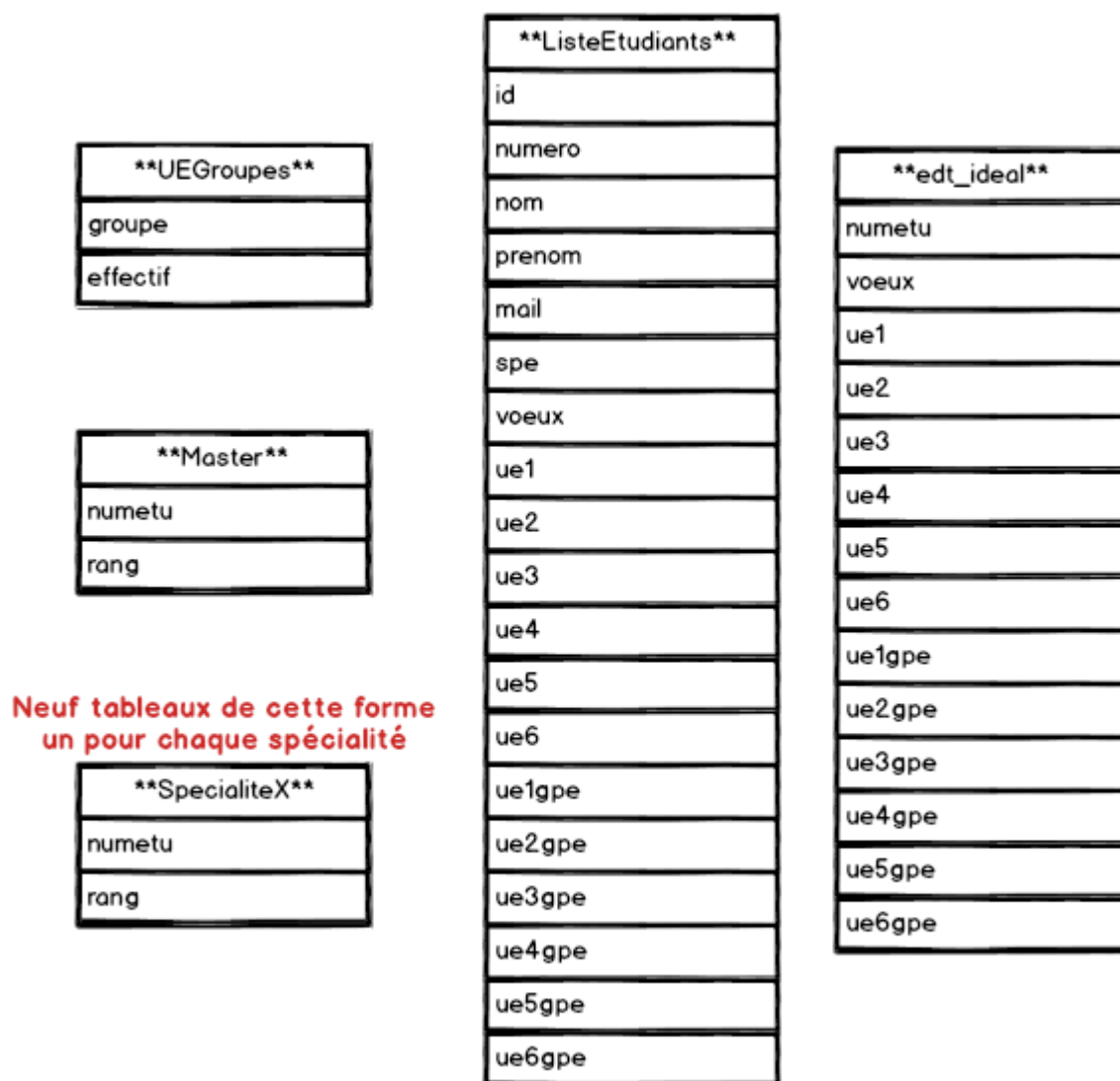


Figure 3.0 – Représentation de la structure de la base de données du site

3.5.2 Fichiers de données au lieu de tirer parti de la base de données

L'ancien site s'appuyait déjà sur des fichiers de données pour stocker par exemple les correspondances des UE avec leurs horaires respectives, la liste des spécialités ou l'ensemble des UE proposées par chaque spécialité du master. Nous avons choisi de conserver ce mode de fonctionnement, afin de réduire le nombre de requêtes vers la base de données et ainsi limiter la charge sur le serveur web du site.

3.5.3 Poids, tri et classement des emplois du temps

Les emplois du temps sont rangés en deux grandes classes:

- **La classe 1 ou « classe des EDTVerts »** n'est composée que d'emplois du temps dont tous les groupes contiennent suffisamment de place.
- **La classe 2 ou « classe des EDTRouges »** contient des emplois du temps dont certains groupes sont presque pleins voire déjà pleins.



Le poids d'un emploi du temps représente le coefficient permettant de déterminer sa classe d'appartenance, mais aussi son classement (positionnement) au sein d'une même classe. Le poids d'un EDT est calculé selon la moyenne géométrique des effectifs libres des groupes qui le composent. Cette moyenne s'obtient en calculant la racine n-ième du produit de n nombres. Le poids d'un EDT est donc obtenu en faisant la racine n-ième du produit du nombre de places restantes dans chaque groupe où n représente le nombre de groupes de l'EDT. Quant à elle, la moyenne arithmétique dite moyenne «ordinaire» est la somme des valeurs numériques (d'une liste) divisée par le nombre de ces valeurs numériques.

Pourquoi ne pas simplement avoir fait la moyenne arithmétique des effectifs libres des groupes ?

Expliquons-cela avec un exemple simple :

L'effectif d'un groupe de quelque UE que ce soit est plafonné à 25 étudiants (effectif maximal). Supposons qu'un étudiant ait le choix entre deux emplois du temps composés de groupes dont les effectifs respectifs sont les suivants :

Choix Edt1 :

Groupe1 : 24, Groupe2 : 24, Groupe3 : 24, Groupe4 : 24, Groupe5 : 0

Ce qui implique en termes de nombre de places restantes par groupe :

➔ Groupe1 : 1, Groupe2 : 1, Groupe3 : 1, Groupe4 : 1, Groupe5 : 25

Choix Edt2 :

Groupe1 : 19, Groupe2 : 19, Groupe3 : 19, Groupe4 : 19, Groupe5 : 20

Ce qui implique en termes de nombre de places restantes par groupe :

➔ Groupe1 : 6, Groupe2 : 6, Groupe3 : 6, Groupe4 : 6, Groupe5 : 5

Les moyennes arithmétiques des effectifs restants de ces deux emplois du temps sont **égales** :
 $(1+1+1+1+25)/5 = (6+6+6+6+5)/5 = 29/5 = 5,8$

Par contre, les moyennes géométriques de ces mêmes emplois du temps donnent des **résultats complètement différents** :

Choix Edt1 : $(1*1*1*1*25)^{1/5} = 25^{1/5} = 1,9$

Choix Edt2 : $(6*6*6*6*5)^{1/5} = 6480^{1/5} = 5,7$

La moyenne arithmétique n'est pas assez précise pour permettre de faire ressortir les caractéristiques individuelles (taux de remplissage) de chaque groupe composant un emploi du temps. L'objectif étant de favoriser les emplois du temps dont les groupes sont chacun le moins rempli possible, le choix d'une moyenne géométrique était plus approprié, la moyenne arithmétique permettant de ne connaître que le taux de remplissage global (de tous les groupes) d'un emploi du temps.

Une fois les emplois du temps classés et triés, ces derniers sont affichés dynamiquement dans la page **choix_ues.php** verticalement (l'un en dessous de l'autre) en commençant par la classe des EDT Verts en haut suivie par la classe des EDT Rouges en dessous.

Au sein d'une même classe, l'emploi du temps ayant le poids maximum est affiché tout en haut, et celui ayant le poids minimum tout en bas (Tri décroissant selon le poids).

Bien entendu, il y a suffisamment d'espace dans la page **choix_ues.php** pour afficher horizontalement plusieurs emplois du temps à la suite, mais nous avons choisi un affichage vertical afin de privilégier le choix des emplois du temps ayant un poids élevé. Cela a pour but de favoriser l'équilibrage des effectifs des différents groupes de TD/TME.



4 Génération d'EDT

4.1 Principe de base

Dans cette section, nous allons présenter le pseudo-code représentant les étapes importantes de la fonction `edt()` dont le comportement est détaillé dans la section '**2.4 Génération d'EDT**'. Afin d'alléger le pseudo-code, une fonction fictive '**traitement_repetitif()**' rassemblant les instructions répétées dans chaque boucle est utilisée. Cette dernière se résume de la manière suivante : pour chaque horaire de TD-TME de l'UE courante, si cet horaire est disponible (absent du tableau des horaires constituant l'emploi du temps en construction) alors l'ajouter au tableau, sinon, arrêter la boucle courante (**break**).

```
function procedure edt(UEs) : returns List<EDT>
    tri(UEs);
    EDT := ∅ ;
    for ue1 := 1 to nb_horaires_ue1 do
        EDT := ajout_horaires_cours(UEs) ;
        if EDT.length != 5 then break ;
        EDT := traitement_repetitif();
        for ue2 := 1 to nb_horaires_ue2 do
            EDT := traitement_repetitif();
            for ue3 := 1 to nb_horaires_ue3 do
                EDT := traitement_repetitif();
                for ue4 := 1 to nb_horaires_ue4 do
                    EDT := traitement_repetitif();
                    for ue5 := 1 to nb_horaires_ue5 do
                        EDT := traitement_repetitif();
                        pds := poids(EDT);
                        if categorie(pds) == 1 then
                            ListeVerte.push(EDT, pds);
                        else if categorie(pds) == 2 then
                            ListeOrange.push(EDT, pds);
                        else
                            ListeRouge.push(EDT, pds);
```

***À la fin de l'insertion des horaires de la 5ème ue on obtient un tableau «EDT» de la forme : EDT = [['nom_ue1', groupe_ue1], ['nom_ue2', groupe_ue2], ['nom_ue3', groupe_ue3], ['nom_ue4', groupe_ue4], ['nom_ue5', groupe_ue5]] qui représente un emploi du temps généré et où groupe_ueN représente le numéro de groupe de la N-ième UE.**

La suite de la fonction `edt()` est une duplication d'instructions d'affichage pour chaque liste d'emplois du temps (`listeEDTn`) triée selon le taux de remplissage de ses groupes. En tout, la fonction `edt()` tenait dans l'ancienne version sur pas moins de **748 lignes** de code.



4.2 Améliorations apportées

4.2.1 Découpage et refactoring de la fonction `edt()`

Un défaut majeur de la fonction `edt(num, spe, ue1, ue2, ue3, ue4, ue5)` est qu'elle est structurellement formatée pour un étudiant qui choisit exactement 5 UE. Elle ne s'adapte donc pas par exemple au parcours d'excellence (choix de plus de 5 UE) ou au parcours de redoublant (choix de moins de 5 UE). Pour traiter le cas des redoublants, l'ancienne version faisait du bourrage en ajoutant des UE fictives parmi les UE réellement choisies par un étudiant redoublant, et ce pour pouvoir respecter le format fixe à 5 UE.

Il aurait été coûteux et surtout peu efficace de continuer sur cette voie en passant à un format à 6 ou 7 UE intégrant le parcours d'excellence. La solution : **GÉNÉRICITÉ**.

La fonction `edt(num, spe, ue1, ue2, ue3, ue4, ue5)` a été partagée en 2 fonctions génériques :

- `edt(choix)` s'adaptant à un nombre variable d'UE choisies (choix est un tableau non contraint d'UE).
- `h_edt(ue)` où le paramètre « ue » représente une UE en particulier.

4.2.2 Rôles et fonctionnement des nouvelles fonctions de génération d'EDT

La fonction générique `edt(choix)` a pour rôle d'initialiser des structures (tableaux et listes) génériques, ne tenant pas compte d'un nombre fixe d'UE composant le choix d'un étudiant quel que soit son parcours, de vérifier la compatibilité des horaires de cours des UE choisies par l'étudiant, d'appeler la fonction `h_edt(ue)` pour chaque UE choisie par l'étudiant et enfin d'appeler la fonction d'affichage des emplois du temps générés. En Gros elle prépare le terrain à la fonction `h_edt(ue)`.

Indications utiles à la compréhension du pseudo-code

- La fonction `getCalendrier()` utilisée au début sert à récupérer la liste complète des horaires (Cours et TD/TME) des UE du semestre courant.
- La fonction `sort_choix_by_nbGroups(choix)` effectue un tri croissant des UE choisies en fonction de leur nombre de groupes de TD/TME.
- La fonction `Cours(horairesUE[ue])` retourne l'horaire du cours d'une 'ue' à partir de la liste complète de ses horaires (cours, TD, Tme).



Pseudo-code de la fonction *edt(choix)*

```
function procedure edt(choix) : returns List<EDT>
    horairesUE := getCalendrier();
    TMP := ∅; EDT := ∅; listeEDT := ∅; NB_LISTEDT := 2;
    for nb1 := 0 to NB_LISTEDT do
        listeEDT[nb1] := ∅;
    sort_choix_by_nbGroups(choix);
    for i := 0 to choix.length-1 do
        if TMP.contains(Cours(horairesUE[choix[i]])) == false then
            TMP.push(Cours(horairesUE[choix[i]]));
    if TMP.containsAll(choix) then
        h_edt(choix.indexOf(choix[0]));
    for i := 0 to listeEDT.length do
        print_edts(listeEDT[i]);
```

Note : la fonction d’affichage (*print_edts*) sait reconnaître les classes d’edts et assigner à chaque classe une couleur significative (claire pour les edts ne présentant pas de risques de dépassement de capacité, foncée sinon).

La fonction ***h_edt(ue)*** se charge de poursuivre la génération d’EDT sur les horaires des groupes de TD/TME :

Pseudo-code de la fonction *h_edt(ue)*

La fonction ***h_edt(ue)*** est en réalité le cœur de la production des emplois du temps. Elle représente la véritable nouvelle implémentation de la fonction ***edt(num, spe, ue1, ue2, ue3, ue4, ue5)*** de l’ancienne version du site. Cette fonction, au lieu de réaliser une construction des emplois du temps grâce à un nombre prédéfini de boucles imbriquées sur *n* niveaux où *n* représente un format fixe du nombre d’UE maximum qu’un étudiant peut choisir, construit récursivement les emplois du temps en essayant d’agencer les horaires des différentes UE choisies. À chaque niveau de récursion, on essaie d’ajouter une UE supplémentaire et ses horaires aux UE ajoutées au niveau des récursions précédents à l’emploi du temps en construction.

Indications utiles à la compréhension du pseudo-code

- Le paramètre ***ue*** représente le numéro de l’ue courante.
- La variable ***TMP*** sert à mémoriser les horaires déjà occupés (par un cours, TD ou TME).
- Les fonctions ***Td(horairesUE[ue])*** et ***Tme(horairesUE[ue])*** retournent respectivement les horaires de TD et TME d’une ‘ue’ à partir de la liste complète de ses horaires (cours, TD, TME).
- La fonction ***chevauchement(ue)*** retourne ***true*** si les horaires de TD/TME de l’ue passée en paramètre sont présents dans le tableau ***TMP***, ***false*** sinon.



- La fonction **bind([ue, groupNum])** retourne une chaîne de caractères composée du nom de l'UE et du numéro de groupe selon le format nomUE-groupNum (Exemple : DLP-2).
- On parcourt tous les groupes de TD/TME pour chaque UE choisie par l'étudiant.
- **GroupNum** représente le numéro du groupe associé à un horaire de TD/TME.
- **EDT** gère les collisions d'horaires en mémorisant les horaires déjà occupés.
- Un nouvel horaire se chevauchant avec un autre déjà présent dans **EDT** est ignoré et on passe à l'horaire suivant.
- Si aucun horaire de groupe n'a pu être ajouté, on arrête la génération d'EDT, sinon on ajoute au tableau **EDT** les groupes qui ont des horaires disponibles au format (nomUE-groupNum). Exemple : IL-3.
- Au dernier niveau de recursion, la liste des horaires construite constitue un emploi du temps généré. Un poids (coefficient) lui est affecté en fonction du taux de remplissage de ses groupes, puis il est ajouté à la classe d'emplois du temps adéquate.
- On supprime les horaires de TD/TME de l'UE courante (**TMP.remove(...)**) pour préparer le niveau de récursion suivant.

```
function procedure h_edt(ue)
    next_ue := ue+1;
    if ue < choix.length then
        for groupNum := 1 to horairesUE[choix[ue]].length do
            while (groupNum < horairesUE[choix[ue]].length
                    AND chevauchement(choix[ue]) == true) do
                h := h + 1;
            if groupNum == horairesUE[choix[ue]].length then break ;
            TMP.push(Td(horairesUE[choix[ue]])[groupNum]);
            TMP.push(Tme(horairesUE[choix[ue]])[groupNum]);
            EDT.push(bind([choix[ue], groupNum]));
            if ue == choix.length-1 then
                listeEDT[categorie(poids(EDT))].push(poids(EDT), EDT);
            h_edt(next_ue); // Appel récursif pour l' ue suivante
            TMP.remove(Td(horairesUE[choix[ue]])[groupNum]);
            TMP.remove(Tme(horairesUE[choix[ue]])[groupNum]);
            EDT.remove(bind([choix[ue], groupNum]));
```

4.2.3 Réduction de la taille du code et gain en lisibilité

Les deux nouvelles fonctions de génération des emplois du temps jouent exactement le même rôle que l'ancienne fonction **edt()** mais leur écriture est simplifiée et de fait plus facilement compréhensible. En tout, les nouvelles fonctions de génération occupent 68 lignes de codes. Les nouvelles fonctions d'affichages quant à elles font au total 54 lignes de code. En tout, pour générer puis afficher les emplois du temps, l'ensemble des fonctions utilisées occupe 122 lignes de code contre 748 pour l'ancienne fonction **edt()**, ce qui représente un gain (en lignes de code) de plus de **83%**. Cette réduction de la taille du code en plus des commentaires ajoutés améliore la lisibilité du code et facilite sa compréhension.



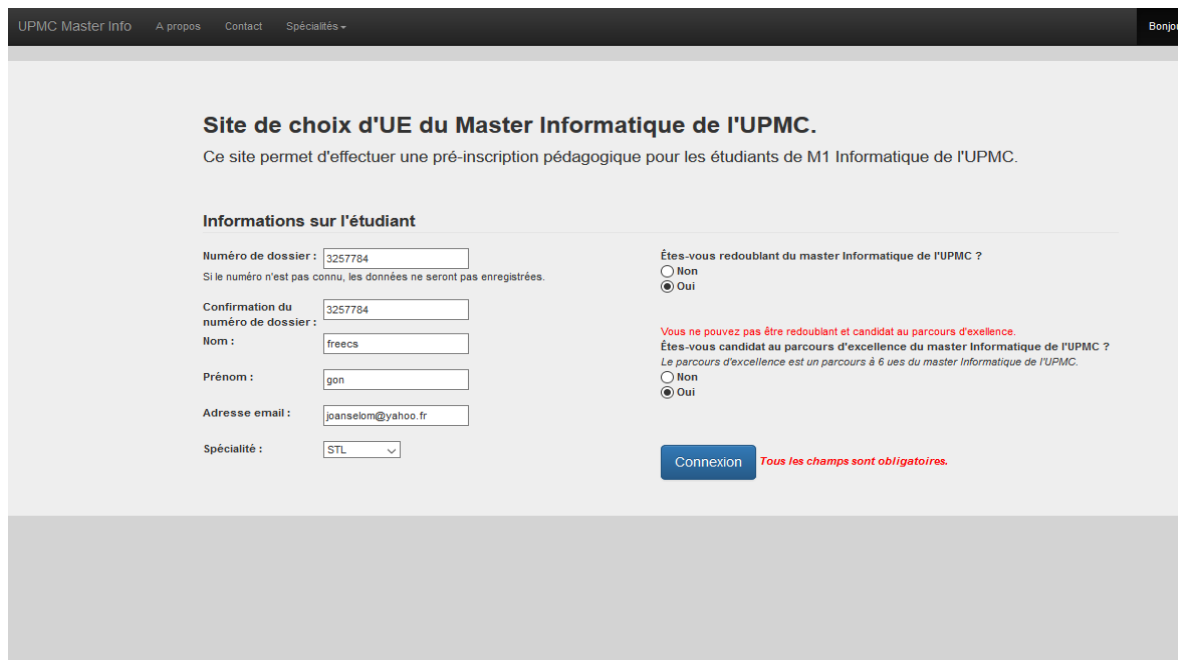
5 Conclusion

Ce PSTL, en plus des connaissances acquises sur le plan technique, nous a surtout appris à conduire de bout en bout un projet sur le long terme (plusieurs mois) tout en gérant en parallèle les tâches liées aux autres UE du semestre. En plus d'affiner notre capacité à gérer intelligemment notre temps, nous avons appris à mettre en place des processus de communication et d'échange efficaces non seulement entre binômes mais aussi avec notre responsable de projet. Ce projet fût aussi l'opportunité de réaliser un outil qui contribuera réellement à faciliter l'organisation des inscriptions au M1 des étudiants du master informatique de l'UPMC, ce qui représente pour nous une petite fierté. Il pourrait même être utilisé pour la gestion des inscriptions au M2 pour le S3 ou d'autres formations ayant des contraintes similaires pour le choix d'UE.



6 Annexes

Exemple imagé d'utilisation du site



UPMC Master Info A propos Contact Spécialités + Bonjour

Site de choix d'UE du Master Informatique de l'UPMC.

Ce site permet d'effectuer une pré-inscription pédagogique pour les étudiants de M1 Informatique de l'UPMC.

Informations sur l'étudiant

Número de dossier :	<input type="text" value="3257784"/>	Êtes-vous redoublant du master Informatique de l'UPMC ? <input type="radio"/> Non <input checked="" type="radio"/> Oui
Si le numéro n'est pas connu, les données ne seront pas enregistrées.		
Confirmation du numéro de dossier :	<input type="text" value="3257784"/>	Vous ne pouvez pas être redoublant et candidat au parcours d'excellence. Êtes-vous candidat au parcours d'excellence du master Informatique de l'UPMC ? Le parcours d'excellence est un parcours à 6 ues du master Informatique de l'UPMC. <input type="radio"/> Non <input checked="" type="radio"/> Oui
Nom :	<input type="text" value="freecs"/>	
Prénom :	<input type="text" value="gon"/>	
Adresse email :	<input type="text" value="janselom@yahoo.fr"/>	
Spécialité :	<input type="text" value="STL"/>	

Tous les champs sont obligatoires.



● edtpstl@gmail.com
À joanselom@yahoo.fr

Bonjour gon.

Votre identifiant de session est : 3968d8973ba1129ea57bcb6513dfc785

← Répondre ← Répondre à tous → Transférer *** Plus

UPMC Master Info A propos Contact Spécialités+

Bonjour gon

Vérification de l'email étudiant

Identifiant de vérification

3ba1129ea57bcb6513dfc785

Vérifier

Veuillez saisir l'identifiant qui vous a été envoyé à l'adresse "joanselom@yahoo.fr".

Vous n'avez pas reçu de mail ? Vérifiez vos spams ou

Renvoyer un email

Retour

UPMC Master Info A propos Contact Spécialités+

Bonjour gon

Saisie des UEs Validées

Retour

Informations sur l'étudiant (Unités d'enseignement validées)

Sélectionnez les UEs que vous avez déjà validées :

☐ AAGB
☒ DLP
☐ MOBJ
☐ SIGNAL

☐ ALGAV
☐ ELECAN1
☐ MODEL
☐ VLS11

☐ ARCH11
☐ IL
☐ MOGPL

☐ ARES
☐ LRC
☐ NOYAU

☐ BIMA
☐ MAPSI
☐ PR

☐ COMPLEX
☐ MLBDA
☐ RTEL

Valider

Si vous n'avez validé aucune UE, cliquez directement sur Valider.



Choix des UEs

Dans la liste ci-dessous, les **UE obligatoires** sont grisées.

Il vous reste alors à compléter le choix d'UE afin de générer les emplois du temps disponibles.

Chaque spécialité indique certaines **UE recommandées**. La [description complète du master est consultable ICI](#).

Un clic sur un nom d'UE ouvre la page de sa description dans un nouvel onglet.

[Retour](#)

Choix des unités d'enseignement

Sélectionnez les ues que vous souhaitez suivre, les emplois du temps seront générés ci-dessous :

UEs obligatoires :

☒ **ALGAV**

UEs recommandées :

☐ IL ☐ LRC ☐ MLBDA ☐ NOYAU ☐ PR

Autres UEs :

☐ AAGB ☐ ARCH11 ☐ ARES ☐ BIMA ☐ COMPLEX ☐ ELECAN1 ☐ MAPSI ☐ MOBJ ☐ MODEL ☐ MOGPL ☐ RTEI ☐ SIGNAL ☐ VLSI1

Choix des UEs

Tous les emplois du temps compatibles avec les UE que vous avez choisies sont affichés ci-dessous.

Ils sont répertoriés en 2 classes suivant le taux de remplissage actuel des groupes de TD/TME.

Les emplois du temps de **classe 1 (contours VERT)** sont ceux, dont les groupes contiennent encore de la place.

Les emplois du temps de **classe 2 (contours ROUGE)** sont ceux, dont certains groupes sont presque pleins voire pleins.

Attention: En choisissant un emploi du temps **Rouge**, il est probable que vos vœux soient modifiés à la rentrée.

Nous vous rappelons qu'il s'agit de vœux d'UE et d'emploi du temps, et quels que soient vos choix, il est possible qu'ils soient modifiés lors de la pré-rentrée, pour des raisons pédagogiques ou de contraintes de remplissage de groupes.

[Retour](#)

Choix des unités d'enseignement

Sélectionnez les ues que vous souhaitez suivre, les emplois du temps seront générés ci-dessous :

UEs obligatoires :

☒ **ALGAV**

UEs recommandées :

☒ IL ☒ LRC ☒ MLBDA ☐ NOYAU ☐ PR

Autres UEs :

☐ AAGB ☐ ARCH11 ☐ ARES ☐ BIMA ☐ COMPLEX ☐ ELECAN1 ☐ MAPSI ☐ MOBJ ☐ MODEL ☐ MOGPL ☐ RTEI ☐ SIGNAL ☐ VLSI1

Il y a 32 emplois du temps de classe 1

	lundi	mardi	mercredi	jeudi	vendredi
08:30 - 10:30	MLBDA-1		IL	ALGAV-2	
10:45 - 12:45	MLBDA-1	ALGAV		ALGAV-2	IL-2
13:45 - 15:45		LRC-3	LRC		IL-2
16:00 - 18:00		LRC-3	MLBDA		

	lundi	mardi	mercredi	jeudi	vendredi
08:30 - 10:30	MLBDA-1		IL	ALGAV-2	
10:45 - 12:45	MLBDA-1	ALGAV		ALGAV-2	
13:45 - 15:45		LRC-3	LRC	IL-3	
16:00 - 18:00		LRC-3	MLBDA	IL-3	



Choix des UEs

Tous les emplois du temps compatibles avec les UE que vous avez choisies sont affichés ci-dessous.

Ils sont répertoriés en 2 classes suivant le taux de remplissage actuel des groupes de TD/TME.

Les emplois du temps de **classe 1 (contours VERT)** sont ceux, dont les groupes contiennent encore de la place.

Les emplois du temps de **classe 2 (contours ROUGE)** sont ceux, dont certains groupes sont presque pleins voire pleins.

Attention: En choisissant un emploi du temps **Rouge**, il est probable que vos vœux soient modifiés à la rentrée.

Nous vous rappelons qu'il s'agit de vœux d'UE et d'emploi du temps, et quels que soient vos choix, il est possible qu'ils soient modifiés lors de la pré-rentrée, pour des raisons pédagogiques ou de contraintes de remplissage de groupes.

[Retour](#)

Choix des unités d'enseignement

Sélectionnez les ues que vous souhaitez suivre, les emplois du temps seront générés ci-dessous :

UEs obligatoires :

☒ ALGAV

UEs recommandées :

☒ IL ☒ LRC ☒ MLBDA ☐ NOYAU ☐ PR

Autres UEs :

☐ AAGB ☐ ARCHIT ☐ ARES ☐ BIMA ☐ COMPLEX ☐ ELECANAI ☐ MAPSI ☐ MOBJ ☐ MODEL ☐ MOGPL ☐ RTEI ☐ SIGNAL ☐ VLSII

Il y a 32 emplois du temps de classe 1



Vendredi	lundi	mardi	mercredi	jeudi	vendredi
08:30 - 10:30	MLBDA-1		IL	ALGAV-2	
10:45 - 12:45	MLBDA-1	ALGAV		ALGAV-2	IL-6
13:45 - 15:45		LRC-3	LRC		IL-6
16:00 - 18:00		LRC-3	MLBDA		



	lundi	mardi	mercredi	jeudi	vendredi
08:30 - 10:30	MLBDA-1		IL	ALGAV-2	
10:45 - 12:45	MLBDA-1	ALGAV		ALGAV-2	
13:45 - 15:45		LRC-3	LRC	IL-6	
16:00 - 18:00		LRC-3	MLBDA	IL-6	



	lundi	mardi	mercredi	jeudi	vendredi
08:30 - 10:30			IL	ALGAV-2	
10:45 - 12:45		ALGAV		ALGAV-2	MLBDA-4
13:45 - 15:45		LRC-3	LRC	IL-6	MLBDA-4
16:00 - 18:00		LRC-3	MLBDA	IL-6	



	lundi	mardi	mercredi	jeudi	vendredi
08:30 - 10:30			IL	ALGAV-3	
10:45 - 12:45		ALGAV		ALGAV-3	MLBDA-4
13:45 - 15:45	LRC-1		LRC	IL-6	MLBDA-4
16:00 - 18:00	LRC-1		MLBDA	IL-6	



	lundi	mardi	mercredi	jeudi	vendredi
08:30 - 10:30			IL	ALGAV-2	
10:45 - 12:45		ALGAV		ALGAV-2	MLBDA-4
13:45 - 15:45	LRC-1		LRC	IL-6	MLBDA-4
16:00 - 18:00	LRC-1		MLBDA	IL-6	

Il y a 23 emplois du temps de classe 2



	lundi	mardi	mercredi	jeudi	vendredi
08:30 - 10:30				IL	
10:45 - 12:45		ALGAV			MLBDA-4
13:45 - 15:45	LRC-1	ALGAV-1	LRC	IL-6	MLBDA-4
16:00 - 18:00	LRC-1	ALGAV-1	MLBDA	IL-6	



	lundi	mardi	mercredi	jeudi	vendredi
08:30 - 10:30			IL	LRC-1	
10:45 - 12:45		ALGAV		LRC-1	MLBDA-4
13:45 - 15:45		ALGAV-1	LRC	IL-6	MLBDA-4
16:00 - 18:00		ALGAV-1	MLBDA	IL-6	



	lundi	mardi	mercredi	jeudi	vendredi
08:30 - 10:30	MLBDA-1			IL	
10:45 - 12:45	MLBDA-1	ALGAV			IL-6
13:45 - 15:45	LRC-1	ALGAV-1	LRC		IL-6
16:00 - 18:00	LRC-1	ALGAV-1	MLBDA		



UPMC Master Info
A propos
Contact
Spécialités -

Bonjour gon

Confirmation de l'Emploi du temps choisi

Retour

Les vœux ne seront enregistrés que lorsque vous les aurez validés.

	lundi	mardi	mercredi	jeudi	vendredi
08:30 - 10:30	MLBDA-1		IL	ALGAV-2	
10:45 - 12:45	MLBDA-1	ALGAV		ALGAV-2	IL-6
13:45 - 15:45		LRC-3	LRC		IL-6
16:00 - 18:00		LRC-3	MLBDA		

Valider cet emploi du temps
La page peut prendre un certain temps à se charger. Merci de ne pas la recharger!

UPMC Master Info
A propos
Contact
Spécialités -

Bonjour gon

Récapitulatif de votre pré-inscription

Dossier n° : 3257784
Étudiant : gon FREEC S
Le 29.04.16 à 03:02:14
Master Spécialité STL
Rang : 3

IMPORTANT : Vous devez télécharger la Version Imprimable et l'apporter le jour des inscriptions pédagogiques. Vous devez ensuite choisir votre Emploi du Temps idéal en cliquant sur "EDT Ideal" pour finaliser votre pré-inscription.

UE	Groupe de TD/TME
ALGAV	2
LRC	3
MLBDA	1
IL	6

	lundi	mardi	mercredi	jeudi	vendredi
08:30 - 10:30	MLBDA-1		IL	ALGAV-2	
10:45 - 12:45	MLBDA-1	ALGAV		ALGAV-2	IL-6
13:45 - 15:45		LRC-3	LRC		IL-6
16:00 - 18:00		LRC-3	MLBDA		

EDT Ideal
Version Imprimable



UPMC Master Info
A propos
Contact
Spécialités

Bonjour gon

Récapitulatif de votre pré-inscription

Dossier n° : 3257784
Étudiant : gon FREECs
Le 29.04.16 à 03:02:14
Master Spécialité STL
Rang : 3

IMPORTANT : Vous devez télécharger la Version Imprimable et l'apporter le jour des inscriptions pédagogiques. Vous devez ensuite choisir votre Emploi du Temps idéal en cliquant sur "EDT Idéal" pour finaliser votre pré-inscription.

UE	Groupe de TD/TME
ALGAV	2
LRC	3
MLBDA	1
IL	6

	lundi	mardi	mercredi	jeudi	vendredi
08:30 - 10:30	MLBDA-1		IL	ALGAV-2	
10:45 - 12:45	MLBDA-1	ALGAV		ALGAV-2	IL-6
13:45 - 15:45		LRC-3	LRC		IL-6
16:00 - 18:00		LRC-3	MLBDA		

EDT Idéal
Version Imprimable

Ouverture de m1_s1_voeux_3257784.pdf

Vous avez choisi d'ouvrir :

m1_s1_voeux_3257784.pdf

qui est un fichier de type : Adobe Acrobat Document
à partir de : http://localhost

Que doit faire Firefox avec ce fichier ?

☐ Ouvrir avec Adobe Acrobat Reader DC (défaut)

☒ Enregistrer le fichier

☐ Toujours effectuer cette action pour ce type de fichier.

OK
Annuler

UPMC Master Info
A propos
Contact
Spécialités

Bonjour gon

Emploi du Temps Idéal

A des fins purement statistiques, nous aimerions savoir quel aurait été votre emploi du temps idéal. Cela ne modifiera en rien votre pré-inscription!

Il y a 32 emplois du temps de classe 1

☐

	lundi	mardi	mercredi	jeudi	vendredi
08:30 - 10:30	MLBDA-1		IL	ALGAV-3	
10:45 - 12:45	MLBDA-1	ALGAV		ALGAV-3	IL-6
13:45 - 15:45		LRC-3	LRC		IL-6
16:00 - 18:00		LRC-3	MLBDA		

☐

	lundi	mardi	mercredi	jeudi	vendredi
08:30 - 10:30	MLBDA-1		IL		
10:45 - 12:45	MLBDA-1	ALGAV			
13:45 - 15:45	LRC-1	ALGAV-1	LRC	IL-6	
16:00 - 18:00	LRC-1	ALGAV-1	MLBDA	IL-6	

☐

	lundi	mardi	mercredi	jeudi	vendredi
08:30 - 10:30	MLBDA-1		IL		
10:45 - 12:45	MLBDA-1	ALGAV			IL-6
13:45 - 15:45	LRC-1	ALGAV-1	LRC		IL-6
16:00 - 18:00	LRC-1	ALGAV-1	MLBDA		

☐

	lundi	mardi	mercredi	jeudi	vendredi
08:30 - 10:30	IL-1		IL	MLBDA-3	
10:45 - 12:45	IL-1	ALGAV		MLBDA-3	
13:45 - 15:45	LRC-1	ALGAV-1	LRC		
16:00 - 18:00	LRC-1	ALGAV-1	MLBDA		

UPMC Master Info
A propos
Contact
Spécialités

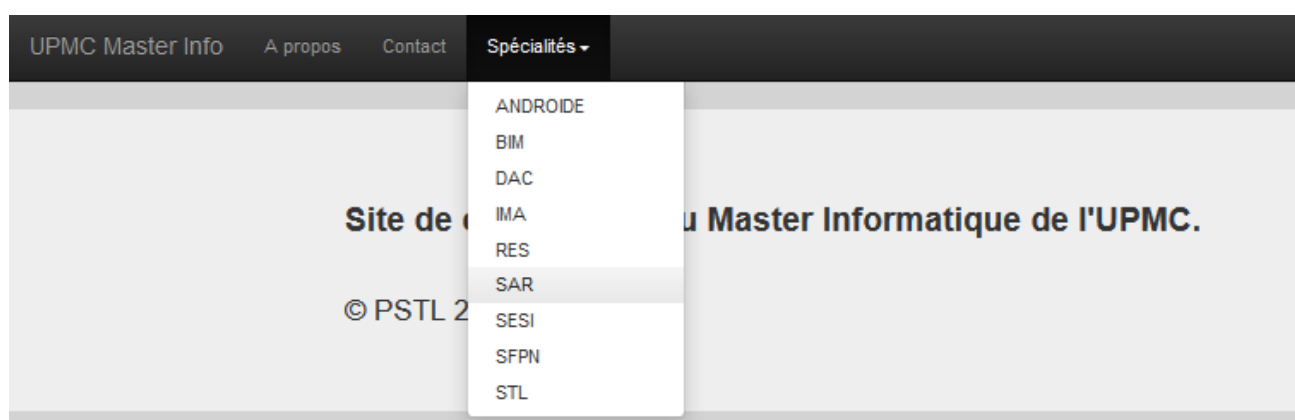
Bonjour gon

Pré-inscription terminée, vous pouvez à présent quitter le site.

Toute demande de modification se fera directement auprès des responsables de la spécialité le jour des inscriptions pédagogiques.



Adresses mail et numéros des secrétariats du Master Informatique de l'UPMC	
Spécialié ANDROIDE Secrétaire : Nicole Nardy Tél. :01 44 27 73 32 Courriel : master.info.androide@upmc.fr	Spécialié SAR Secrétaire :Mlle Aliénor Le Conte Tél. :01 44 27 31 20 Courriel :Alienor.LeConte@ufr-info-p6.jussieu.fr
Spécialié BIM Secrétaire :Aliénor Le Conte Tél. :01 44 27 31 20 Courriel :master.info.bim@upmc.fr	Spécialié SESI Secrétaire :Jennyta Bara Tél. :01 44 27 72 78 Courriel : master.info.sesi@upmc.fr
Spécialié DAC Secrétaire :Geraldine Bompard Tél. :01 44 27 82 56 Courriel : master.info.dac@upmc.fr	Spécialié SFPN Secrétaire :Katia Pytel Tél. :01 44 27 53 82 Courriel : master.info.sfpn@upmc.fr
Spécialié IMA Secrétaire :Géraldine Bompard Tél. :01 44 27 62 05 Courriel : master.info.ima@upmc.fr	Spécialié STL Secrétaire :Emilie Auger Tél. :01 44 27 53 80 Courriel :master.info.stl@upmc.fr



UPMC - Master Informatique - Spécialité STL - Voeux M1-S1 de 3257784

UPMC - Master Informatique <edtpstl@gmail.com>


À joanselom@yahoo.fr, edtpstl@gmail.com, nouirachafik@hotmail.com

Aujourd'hui à 3h03

Bonjour gon,

Veuillez trouver ci-joint un fichier PDF contenant toutes les informations relatives à votre pré-inscription. Vous devez l'imprimer et l'apporter le jour des inscriptions pédagogiques, la date et la salle vous seront communiquées par mail.

Cordialement,
Master Informatique de l'UPMC - Spécialité STL



m1_s1_vo...pdf

Répondre

Répondre à tous

Transférer

Plus

Cliquez pour Répondre, Répondre à tous ou Transmettre

Envoyer

Tt B I A

m1_s1_voeux_3257784.pdf

1 / 1

UPMC

SORBONNE UNIVERSITÉS

Voeux M1-S1 de 3257784

Rang : 3

Spécialité : STL

Ce fichier doit être imprimé et présenté le jour des inscriptions.

Dossier n° : 3257784

Étudiant : gon FREECS

Créé le vendredi 29 avril 2016 à 03:02:15

Liste des UEs	Groupe de TD/TME
ALGAV	2
LRC	3
MLBDA	1
IL	6

	Lundi	Mardi	Mercredi	Jeudi	Vendredi
8:30 - 10:30	MLBDA-1		IL	ALGAV-2	
10:45 - 12:45	MLBDA-1	ALGAV		ALGAV-2	IL-6
13:45 - 15:45		LRC-3	LRC		IL-6
16:00 - 18:00		LRC-3	MLBDA		



Manuel administrateur

Ce manuel indique où trouver les informations spécifiques du projet (scripts & données).

 Les fichiers PHP se trouvent à la racine du projet :

- ❖ **MSN.php** contient les adresses mail des secrétariats de chaque spécialité.
- ❖ **effectifs.php** contient le script de rafraîchissement des effectifs Ajax coté serveur.
- ❖ **genererMailPDF.php** contient le script de génération du PDF et d'envoi du mail récapitulatif avec pièce jointe.
- ❖ **genererPDF.php** contient le script de génération du PDF pour un téléchargement direct (clic sur le bouton 'version imprimable').
- ❖ **semestre.php** est principalement un fichier de données. Il contient la liste complète des UE de chaque semestre, toutes spécialités confondues, ainsi que la liste des UE proposées par chaque spécialité du master pour chaque semestre. Il permet aussi de définir les périodes de disponibilité du site pour chaque semestre.
- ❖ **send_id.php** contient le script de génération et d'envoi par mail d'un identifiant unique pour chaque connexion d'un étudiant.
- ❖ **start_session.php** contient un script qui sert principalement à démarrer une session (redirection vers la génération d'identifiant de session), ou au contraire à refuser l'accès à la page de vérification de l'identifiant si par exemple le mail saisi par l'étudiant n'est pas conforme ou si celui-ci a déjà validé son inscription.
- ❖ **uerl.php** est un fichier de données contenant les liens (url) vers les pages de description de chaque UE.

 Les fichiers JavaScript se trouvent dans le répertoire « js » du projet :

- ❖ **calendrier.js** est le fichier de données qui centralise les horaires des groupes de TD/TME pour chaque UE de chaque semestre.
- ❖ **choix_ues.js** contient le script associé à la page choix_ues.php.
- ❖ **descriptions.js** centralise tous les messages d'instructions servant à aider l'étudiant dans la compréhension du fonctionnement du site.
- ❖ **edt.js** contient le script de génération des emplois du temps. Il s'appuie sur :
 - **edt_utils.js** qui contient les fonctions auxiliaires à la génération (tri, poids, etc.).
 - **edt_print.js** qui contient les fonctions d'affichage des emplois du temps.
- ❖ **edt_ideal.js** contient le script associé à la page edt_ideal.php
- ❖ **effectif.js** contient le script de rafraîchissement des effectifs Ajax coté client.
- ❖ **index.js** contient les scripts associés à la page index.php et contient notamment les scripts de vérification des champs de formulaire.
- ❖ **saisie_ues_valides.js** contient le script associé à la page saisie_ues_valides.php.
- ❖ **sem2constraints.js** est le fichier où il faut rajouter la définition des fonctions de vérification des contraintes spécifiques à chaque spécialité. Ensuite dans le fichier choix_ues.js/function generateEDTs(nb_suivi) ajouter l'appel de la fonction définie.
- ❖ **jquery-latest.js** est la version de la bibliothèque jQuery utilisée.
- ❖ **Utils.js** contient un ensemble de fonctions utilitaires.



- ✚ Les fichiers CSS se trouvent dans le répertoire « css » du projet :
 - ❖ **choix_ues.css, confirmation.css, edt_ideal.css, index.css, saisie_identifiant.css, saisie_ues_valides.css, validation.css** contiennent le CSS des pages PHP homonymes.
 - ❖ **Maincss.css** contient le CSS principal partagé par toutes les pages PHP.
 - ❖ **Ue.css** contient les codes couleurs associés aux UE dans les emplois du temps.
- ✚ Les fichiers SQL se trouvent dans le répertoire « sql » du projet :
 - ❖ **MasterVoeuxS1.sql** contient la définition des tables de la base de données.
- ✚ Pour ce qui est de la génération des pages HTML, les serveurs de page sont :
choix_ues.php, confirmation.php, edt_ideal.php, index.php, saisie_identifiant.php, saisie_ues_valides.php, validation.php, validation_edt_ideal.php.
- ✚ Rôle des autres répertoires :
 - ❖ Le répertoire **fpdf** contient la bibliothèque *FPDF (Free PDF)*.
 - ❖ Le répertoire **phpmailer** contient la bibliothèque d'envoi de mail avec pièce jointe.
Nb : Cette bibliothèque n'est pas utile à l'envoi des mails d'identification de session.
 - ❖ Le répertoire **edtmp** sert à contenir localement sur le server les documents générés pendant les sessions d'inscription.

✚ Dépôt GitHub

Lien vers le dépôt GitHub contenant les sources de la version finale de notre projet :

https://github.com/Shaficks/PSTL_EDT

