



INSTITUT SUPÉRIEUR D'INFORMATIQUE DE
MODÉLISATION DE LEURS APPLICATIONS

1 rue de la Chebarde
TSA 60125
CS 600026
63 178 Aubière CEDEX



Unité de Formation et de Recherche
en Sciences et Techniques des
Activités Physiques et Sportives
Campus des Cézeaux
BP 104
63 172 Aubière CEDEX

Rapport de stage de 2ème année

Filière 2 – Génie Logiciel et Systèmes Informatiques

Conception et Réalisation d'un Logiciel d'Aide à la Gestion de
l'Emploi du Temps Individualisé
des Étudiants de Licence STAPS
(UFR STAPS Université Clermont Auvergne)

PRESENTE PAR : Ludovic Jozereau

TUTEURS ENTREPRISE : Philippe Vaslin et Vincent Thery

TUTEUR ISIMA : Annegret Wagler

DUREE DU STAGE : 5 mois

DATE DE SOUTENANCE : 18 juillet 2017

Remerciements

Je tiens à remercier Mr Philippe Vaslin, mon tuteur d'entreprise, pour son aide et son soutien au quotidien durant le stage ainsi que pour le temps qu'il a consacré à la résolution de problèmes de gestion.

Je tiens à remercier également Mr Vincent Thery et Mr Christophe Duhamel pour les conseils qu'ils m'ont apportés durant ce projet.

Je remercie enfin Mr Pascal Girodet pour sa collaboration dans la récupération et la normalisation des informations et des fichiers nécessaires au fonctionnement du logiciel.

Résumé

Ce projet a pour objectif la réalisation d'une **application web** permettant la création d'**emplois du temps** pour l'UFR **STAPS**. Elle remplacera de manière plus efficace le tableur **Excel** utilisé à l'heure actuelle pour réaliser cette tâche. En plus de cela, elle permettra aux utilisateurs de renseigner eux-mêmes certaines données utiles, afin de centraliser les données.

L'application devait être développée à l'aide du Framework **Symfony** en langage **PHP** et devait intégrer une base de données. J'ai choisi d'utiliser **MySQL** car il s'agit d'un système de gestion de base de données facilement utilisable avec le Framework imposé.

L'objectif principal était que l'application soit fonctionnelle à la fin du stage, c'est-à-dire qu'il devait être possible de générer une base d'emploi du temps complet. Rendre l'application esthétique était donc un travail non prioritaire qui n'a été effectué qu'à la fin du stage.

Mots clés : **application web, emploi du temps, Excel, MySQL, PHP, Symfony, STAPS**

Abstract

The aim of this project was to create a **web application** allowing the creation of **timetables** for the UFR **STAPS**. It should efficiently replace the **Excel** spreadsheet currently used to perform this task. Moreover, it should allow users to fill in themselves some useful data, in order to centralize data.

The application should be developed using the **Symfony** framework in **PHP** language and should integrate a database. I chose to use **MySQL** because it is a database management system easily usable with the imposed framework.

The main objective was that the application should be functional at the end of the internship that means it should be possible to generate timetable base. Making the aesthetic application was therefore a non-priority job that has only be done at the end of the internship.

Keywords: **web application, timetables, Excel, MySQL, PHP, Symfony, STAPS**

Glossaire

APSA : Activités Physiques Sportives et Artistiques.

Bundle : Ensemble de fichiers et de répertoires permettant d'implémenter une ou plusieurs fonctionnalités, si possible réutilisable(s) dans divers projets.

CM : Cours Magistraux. Type de cours permettant d'enseigner des cours théoriques à un grand nombre d'étudiants à la fois (100 à 300).

Créneau : Période d'une durée déterminée lors d'un jour donné.

EC : Élément Constitutif. Cours qui compose une UE, peut être composé d'APSA.

Framework : Ensemble de composants logiciels simplifiant la création de tout ou d'une partie d'un logiciel.

Layout : Organisation des éléments dans une page.

Parcours : Choix de spécialisation des cours pour un étudiant.

TD : Travaux Dirigés. Type de cours permettant d'appliquer les connaissances apprises pendant les cours théoriques ou d'aborder de nouvelles notions. Généralement en groupe moyen (40 étudiants) pour que l'enseignant puisse s'occuper des étudiants.

TP : Travaux Pratiques. Type de cours permettant d'apprendre via la mise en pratique dans le but d'illustrer ou de compléter la théorie. Généralement en groupe restreint (20 étudiants) à cause du matériel nécessaire.

UE : Unité d'Enseignement. Regroupement de cours constituée d'EC permettant la gestion de paquets de cours.

Table des matières

Remerciements

Résumé

Abstract

Glossaire

Table des matières

Table des figures

Table des annexes

Introduction.....	1
I. Contexte.....	2
I.1. UFR STAPS	2
I.2. Le problème	2
I.3. Organisation du travail	3
II. Analyse.....	6
II.1. Travaux précédents	6
II.2. Analyse complémentaire	7
II.3. Travail à réaliser	8
III. Outils utilisés	10
III.1. Langages	10
III.1.a. PHP	10
III.1.b. JavaScript	10
III.1.c. CSS.....	11
III.1.d. HTML	11
III.1.e. Rôle des langages dans le logiciel	12
III.2. Framework	12
III.2.a. JQuery	12
III.2.b. Bootstrap.....	13
III.2.c. Symfony.....	13
III.2.d. Doctrine.....	13
III.2.e. Twig.....	14
III.3. Motif MVC.....	15
IV. Conception.....	16
IV.1. L'interface utilisateur.....	16
IV.2. La base de données	17
IV.2.a. Utilisateur (User).....	18

IV.2.b. Événements & famille d'événements (Event & event_parent)	19
IV.2.c. Enseignements (EC, UE & APSA)	20
IV.2.d. Lieux de pratique	21
IV.2.e. Liens entre les tables.....	21
IV.3. L'algorithme de création.....	22
V. Réalisation	24
V.1. L'interface utilisateurs	24
V.2. L'algorithme de création	25
V.3. Fichiers de données.....	29
V.3.a. La maquette	29
V.3.b. Les APSA.....	29
V.3.c. La réservation des installations	30
V.3.d. La correspondance entre les installations et les APSA	31
VI. L'application résultante	32
VI.1. Présentation des résultats	32
VI.1.a. Page d'accueil.....	32
VI.1.b. Pages d'inscription et de connexion	33
VI.1.c. Page de l'agenda	33
VI.1.d. Page d'importation de la maquette	35
VI.1.e. Page du choix des cours	36
VI.1.f. Pages de génération de l'emploi du temps	37
VI.2. Utilisation de l'application.....	38
VI.3. Amélioration de l'application	40
Conclusion.....	41
Bibliographie	
Annexes	

Table des figures

Figure 1 : Diagramme de Gantt prévisionnel	3
Figure 2 : Diagramme de Gantt réel	4
Figure 3 : Processus de génération des emplois du temps (d'après [4])	6
Figure 4 : Nouveau processus global de génération des emplois du temps	7
Figure 5 : Liens entre les entités	8
Figure 6 : Type des liens entre les entités	9
Figure 7 : Fonctionnement classique d'une application web	12
Figure 8 : Exemple d'utilisation d'un <i>layout</i>	14
Figure 9 : Fonctionnement du motif Modèle-Vue-Contrôleur	15
Figure 10 : Quelques exemples de skeuomorphisme	17
Figure 11 : Table User	18
Figure 12 : Table Event	19
Figure 13 : Table Event_parent	19
Figure 14 : Table EC	20
Figure 15 : Table UE	20
Figure 16 : Table APSA	20
Figure 17 : Table Lieu	21
Figure 18 : Diagramme de classe des entités du logiciel	21
Figure 19 : Algorithme de génération des emplois du temps	28
Figure 20 : Page d'accueil (non connecté)	32
Figure 21 : Page d'accueil (côté administrateur)	32
Figure 22 : Page d'accueil (côté enseignant)	32
Figure 23 : Page d'inscription (non connecté)	33
Figure 24 : Page de connexion (non connecté)	33
Figure 25 : Page de l'agenda (côté administrateur)	34
Figure 26 : Page de l'agenda (côté enseignant)	35
Figure 27 : Page d'importation des données (côté administrateur)	36
Figure 28 : Page du choix des cours (côté enseignant)	36
Figure 29 : Page d'entrée des paramètres (côté administrateur)	37
Figure 30 : Page de génération de l'emploi du temps (côté administrateur)	38

Table des annexes

Annexe 1 : Diagramme de Gantt prévisionnel	I
Annexe 2 : Diagramme de Gantt réel	II
Annexe 3 : Exemple du fichier de la maquette	III
Annexe 4 : Exemple du fichier des APSA.....	IV
Annexe 5 : Exemple du fichier des réservations des installations	V
Annexe 6 : Exemple du fichier des correspondances APSA-Installations.....	VI

Introduction

Dans le cadre de mon stage de deuxième année à l'ISIMA, j'ai effectué un projet pour l'UFR Sciences et Techniques des Activités Physiques et Sportives (STAPS) de Clermont-Ferrand. Cet établissement doit, chaque année, organiser les différents cours durant la semaine et répartir les enseignants et les étudiants dans ces enseignements afin d'obtenir un emploi du temps complet. Cependant, cet emploi du temps doit répondre à certaines contraintes, telles que les volumes horaires des disciplines ou les disponibilités des lieux de pratique. Ainsi, l'administrateur s'occupant de cette tâche utilise le tableur MS Excel pour simplifier cette création, mais ce dernier a atteint ses limites et n'apporte plus l'aide nécessaire. L'administrateur passe alors beaucoup de temps à créer ces emplois du temps.

Pour soulager l'administrateur, Mr Philippe Vaslin et Mr Vincent Thery ont proposé la réalisation d'une application web capable de générer des emplois du temps de manière automatique grâce aux données entrées par les utilisateurs, comme les créneaux de réservations des différents lieux ou la maquette des disciplines. Ce projet a été démarré en 2015 par un ancien étudiant de troisième année à l'ISIMA, avant d'être poursuivi l'année suivante par un binôme d'étudiants anciennement en deuxième année à l'ISIMA, puis cette année, par un autre binôme d'étudiants de deuxième année. Mon travail consistait à finaliser et à concrétiser cette application web qui n'a jamais été aboutie depuis sa proposition, en prenant en compte l'évolution de la demande et des contraintes par rapport à l'initiation du projet.

Dans ce rapport, je présenterai tout d'abord le contexte du projet. Ensuite, j'exposerai mon analyse à partir des travaux précédents avant de présenter les outils utilisés puis d'expliquer la conception de l'application. Puis j'aborderai la réalisation du projet et je montrerai le résultat final du projet. Enfin, j'exposerai des points améliorables dans l'éventualité d'une reprise du développement de l'application.

I. Contexte

I.1. UFR STAPS

L'Unité de Formation et de Recherche (UFR) en Sciences et Techniques des Activités Physiques et Sportives (STAPS) est une composante de l'Université Clermont Auvergne (UCA). Située sur le campus universitaire des Cézeaux, elle a pour mission le développement de la recherche et des formations scientifiques, techniques et professionnelles dans le domaine des activités physiques, du mouvement et du sport [1].

Durant une année universitaire, les étudiants de l'UFR STAPS doivent suivre un ensemble de disciplines dépendantes de leur parcours et de certains de leur choix. Ces cours peuvent se décliner sous la forme de cours magistraux (CM) en grands effectifs, d'Activités Physiques Sportives et Artistiques (APSA) ou d'enseignements tels que des travaux pratiques (TP) ou des travaux dirigés (TD) en effectifs réduits. Pour valider leur année, ces étudiants doivent obtenir une note minimale et valider des unités d'enseignement (UE), chacune composée de plusieurs cours ou éléments constitutifs (EC). Un étudiant qui ne valide pas son année peut redoubler et retenter de la valider, mais n'a pas besoin de repasser les UE validées l'année précédente, rendant ces EC inutiles à suivre pour ce dernier.

I.2. Le problème

Autant de choix dans les parcours ou les enseignements génère des disparités dans le programme des étudiants, qui peuvent posséder des emplois du temps très différents. En plus de cela, les APSA nécessitent des installations particulières qui ne sont disponibles qu'à certains moments. Enfin, il existe de nombreuses contraintes, plus ou moins évidentes, dans l'élaboration d'un planning complet, comme l'impossibilité pour un étudiant ou un enseignant d'être présent à deux endroits simultanément, la limite horaire d'une journée ou la disponibilité des ressources, humaines ou matérielles, nécessaires pour un cours donné.

À tout cela, on peut encore ajouter les difficultés générées par certains responsables de diplôme ou de parcours dans l'élaboration de l'emploi du temps. En effet, certains de ces responsables proposent leur propre emploi du temps, consacrant ainsi leur temps à la création d'une petite partie de l'emploi du temps complet. De plus, ils adaptent leur partie, tout

naturellement, de sorte à ce que celle-ci réponde à leurs contraintes. Or, cela rend difficile, voire impossible, la compatibilité des différentes parties entre elles et fait perdre énormément de temps à la fois à ces responsables, mais aussi à l'administrateur chargé de faire l'emploi du temps complet. Toutes ces contraintes rendent ce problème NP-complet, c'est-à-dire un problème pour lequel on ne peut pas trouver de solution optimale dans des temps polynomiaux.

La création de l'emploi du temps global est donc une lourde tâche effectuée par une seule personne des semaines durant, pour l'obtention du planning utilisé pendant le semestre par les enseignants et les étudiants. Pourtant, cette tâche pourrait être automatisée par un programme informatique, ce qui ferait gagner énormément de temps à chacun des responsables et à l'administrateur, et pourrait potentiellement fournir un résultat mieux optimisé.

1.3. Organisation du travail

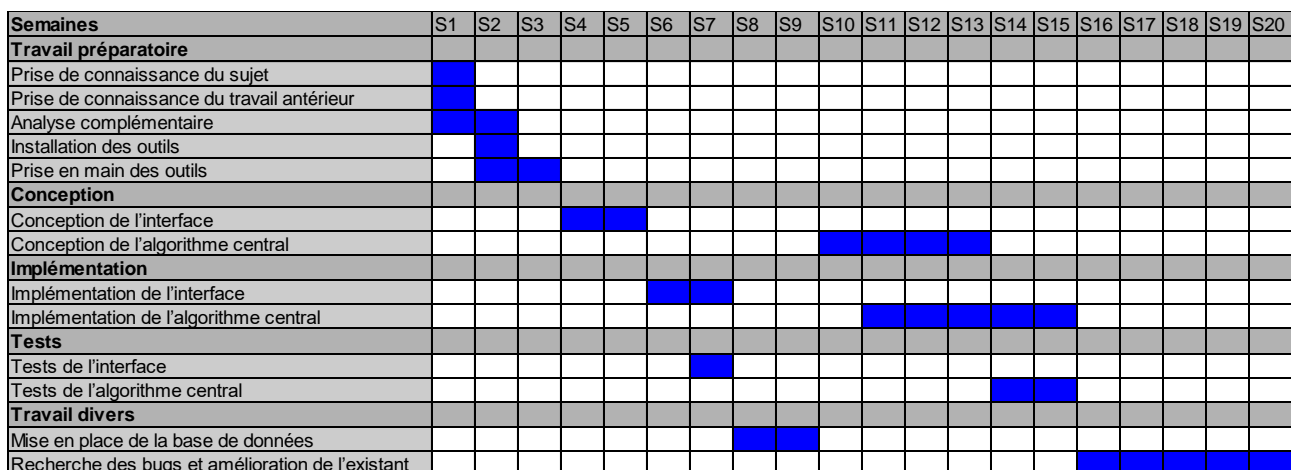


Figure 1 : Diagramme de Gantt prévisionnel

En premier lieu, j'avais prévu de lire les anciens rapports pour prendre connaissance du sujet et des anciens travaux, et avoir ainsi une base d'analyse à compléter. Ensuite, je m'étais réservé une petite période pour installer les outils dont j'aurais besoin et apprendre à m'en servir, à la suite de quoi j'allais commencer à concevoir l'interface avant de l'implémenter et de la tester. Après, je devais mettre en place la base de données pour concevoir l'algorithme central, à savoir l'algorithme de création des emplois du temps, que j'aurais implémenté progressivement, et testé. Enfin, dans le temps restant, je comptais corriger les bugs résiduels et améliorer certains points (Figure 1).

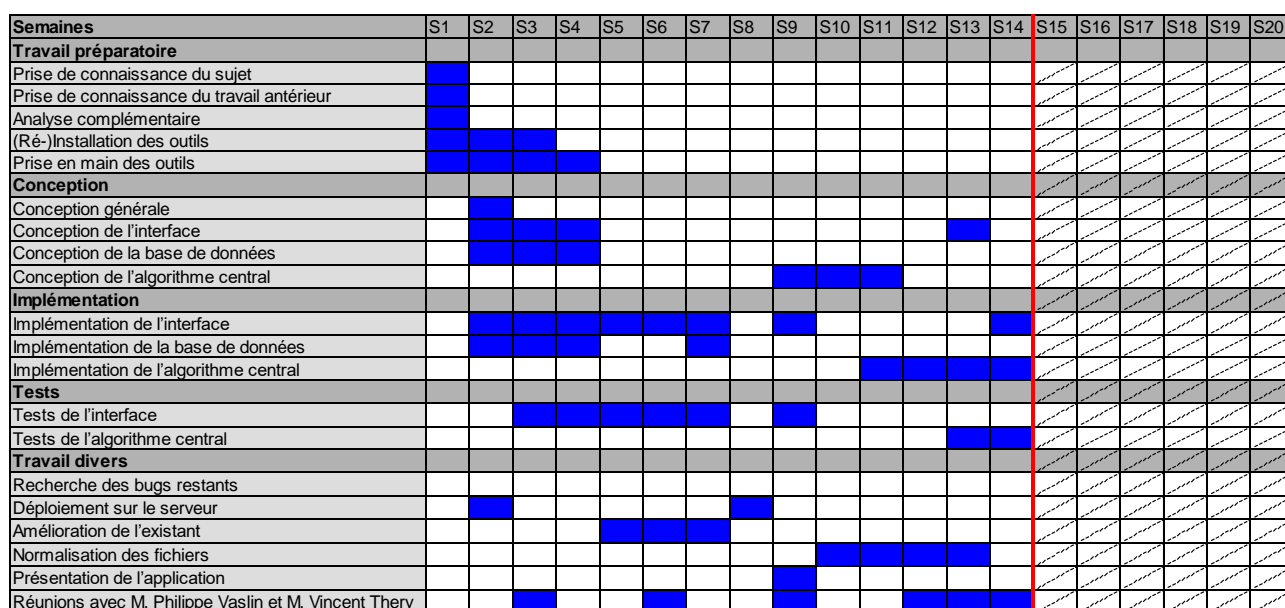


Figure 2 : Diagramme de Gantt réel

Cependant, le travail ne s'est pas exactement déroulé de cette façon. La prise de connaissance a été plus rapide que prévu, tout comme l'analyse complémentaire. En effet, l'analyse effectuée par le dernier binôme [2] m'a tout à fait convenue et je n'ai trouvé que peu à y redire.

A l'inverse, Symfony est un Framework très complet dont il m'a fallu plus de temps que je ne l'imaginais pour en faire un premier tour. De plus, des problèmes techniques m'ont obligé à installer plusieurs fois les outils, ce qui m'a un peu ralenti dans le démarrage. Jusqu'à la première réunion avec mes tuteurs (09/05/17), j'ai commencé à concevoir et implémenter l'interface et la base de données. En effet, de nombreuses fonctionnalités utilisent la base de données et il valait mieux la mettre en place progressivement.

Néanmoins, même si l'interface a été commencée avant la prévision, elle a été terminée plus tard car j'avais sous-estimé le temps de développement d'une interface à la fois fonctionnelle et ergonomique. Une fois cette interface terminée, j'ai effectué une présentation de l'avancement du logiciel à ses futurs utilisateurs (29/05/17), de manière à compléter le travail et leur montrer un aperçu du logiciel.

Après la modification de quelques éléments de l'interface, j'ai commencé la conception de l'algorithme de génération d'emplois du temps. Cela a été plutôt difficile car il me manquait certaines informations qui n'existaient pas de manière formelle au sein de l'UFR STAPS, à savoir les

différentes salles ainsi que les cours qui peuvent y être enseignés. Comme il fallait récupérer ces informations, créer des normes et valider ces fichiers, l'implémentation a duré un peu plus longtemps que prévu, et les tests ont pris du retard par rapport au planning initial. La réunion du 26/06/17 avec Mrs Philippe Vaslin, Vincent Thery et Pascal Girodet m'a permis de constater les avancées effectuées sur la récolte des informations nécessaires et la création des fichiers permettant l'importation de ces données dans le logiciel. J'ai ainsi pu avoir une vision plus précise des formats que j'allais obtenir et ai pu continuer à implémenter le logiciel jusqu'à avoir une première version capable de générer un emploi du temps. Suite à cette avancée et à ces premiers résultats réels, une seconde présentation du logiciel a été effectuée (07/07/17) afin de montrer les possibilités actuelles du logiciel (Figure 2).

II. Analyse

II.1. Travaux précédents

En trois années, le cahier des charges du logiciel a bien évolué. Ainsi, les deux premiers travaux [2] [3] apportaient des idées intéressantes mais peu utilisables telles quelles. De plus, la plupart des éléments utiles d'un travail antérieur avaient été repris et transformés dans le travail suivant.

Ainsi, seul le dernier travail effectué m'a réellement servi de base quant à l'analyse et à la conception du projet. Tout d'abord, cela m'a permis de bien délimiter le problème et les objectifs. En effet, ce projet étant très conséquent, la quantité d'informations reçues d'un coup pouvait intimider et rendre plus difficile le départ. Ce dernier travail [4] m'a débroussaillé le chemin ce qui m'a éclairci les idées et m'a aidé à démarrer.

De plus, ce travail [4] proposait des éléments de réflexion très intéressants, comme la séparation du projet en deux phases : la génération des emplois du temps et l'affectation des étudiants dans les groupes (CM, TD, TP) des différents cours. Cette séparation permet encore de simplifier le problème puisqu'elle subdivise la difficulté (Figure 3).

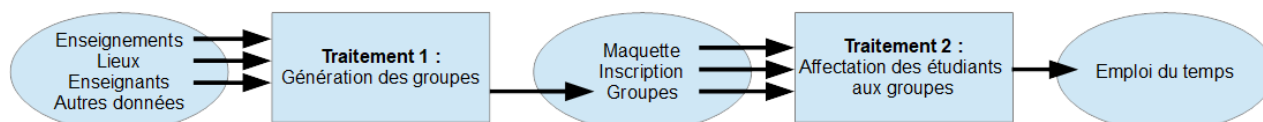


Figure 3 : Processus de génération des emplois du temps (d'après [4])

Un autre élément intéressant concerne la maquette, très conséquente et riche en informations, que ce travail propose de séparer en plusieurs fichiers plus petits et plus spécialisés, qui sont plus propices au traitement par le logiciel.

Enfin, l'algorithme utilisé pour l'affectation, très basique, m'a permis de comprendre en définitive que cette partie était bien plus simple que la première, sur laquelle était centré l'essentiel du problème.

Cependant, la vraie difficulté, à savoir la génération des emplois du temps, n'avait jamais été réellement traitée dans les travaux précédents.

II.2. Analyse complémentaire

En décomposant à mon tour le problème, j'ai procédé à l'analyse suivante. Pour commencer, le problème devait être réalisé étape par étape pour pouvoir avancer sans se perdre dans la quantité. Il fallait se fixer des objectifs proches et atteignables permettant d'avancer vers l'objectif principal. J'en suis venu alors à utiliser une méthode agile pour le développement, c'est-à-dire une méthode de développement dont les moyens principaux sont la mesure, l'ajustement du travail et la conversation avec le client de manière régulière afin de satisfaire ce dernier le mieux possible.

Ensuite, j'ai revu les différentes phases du projet et ai identifié précisément le problème. Je l'ai séparé en trois phases distinctes (Figure 4) : l'insertion des données, la création de l'emploi du temps, puis l'affectation des étudiants.

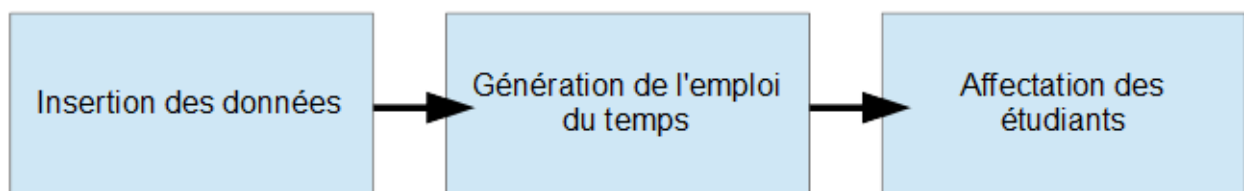


Figure 4 : Nouveau processus global de génération des emplois du temps

La résolution du problème principal se situe au milieu de ce processus, car dès que l'on a créé et placé des « boîtes » vides sur un créneau pour un enseignant, un lieu et un cours donnés, le problème est presque résolu : remplir ces « boîtes » d'étudiants n'est pas difficile. C'est même faisable par un algorithme basique, voire manuellement.

Ainsi, on distingue quatre entités principales : les enseignants, les lieux, les cours et les créneaux. Afin de résoudre le problème, il fallait réussir à relier toutes ces entités entre elles. En effet, c'est l'ensemble des liens qui constitue l'ossature de l'emploi du temps : il faut affecter un cours à un créneau dans un lieu précis, enseigné par un professeur particulier. Chaque lien représente une partie de la solution et/ou une donnée pour la résolution du problème (Figure 5).

	Enseignants	Créneaux	Lieux	Cours
Enseignants				
Créneaux	Disponibilités des enseignants			
Lieux	Disponibilités communes	Disponibilités des salles		
Cours	Cours qu'un enseignant peut dispenser	Affectation d'un cours à un créneau	Lieux où un cours peut être enseigné	

Figure 5 : Liens entre les entités

La collecte de l'ensemble de ces informations pour chaque cours permet d'obtenir l'emploi du temps complet.

II.3. Travail à réaliser

Pour pouvoir créer ces différents liens, il faut tout d'abord une interface permettant la collecte de données. En effet, une partie de ces liens peut être faite directement sans aucun traitement de données (Figure 6) :

- La récupération des disponibilités des enseignants peut se faire grâce à une interface où chaque professeur doit renseigner ses créneaux de disponibilité, comme un formulaire ;
- De la même manière, on peut récupérer les cours qu'un enseignant peut enseigner via une interface comme une liste ;
- La centralisation des disponibilités des salles et des réservations des installations peut se faire grâce à l'importation de fichiers tels que des tableurs Excel (MS) ;
- La correspondance entre cours et lieux peut se faire de la même manière grâce à l'importation de fichiers.

	Enseignants	Créneaux	Lieux	Cours
Enseignants				
Créneaux	Données			
Lieux	Traitement	Données		
Cours	Données	Traitement	Données	

Figure 6 : Type des liens entre les entités

Nous pouvons donc obtenir les deux-tiers des liens de manière brute, en créant une interface ergonomique permettant de renseigner ces informations. De plus, il fallait pouvoir aussi importer la maquette pour l'enregistrer dans la base de données, afin qu'elle fournisse la matière pour tout ce qui concerne les cours, à savoir les choix des enseignants, la correspondance avec les lieux et, plus tard, la génération de l'emploi du temps complet.

Il fallait aussi une interface permettant d'entrer les paramètres utiles à la génération de l'emploi du temps car il doit être possible de modifier certaines contraintes pour visualiser différents résultats. Il fallait donc créer un formulaire pour que l'utilisateur renseigne ces données.

Enfin, il fallait réaliser l'algorithme de création de l'emploi du temps, grâce auquel seront établis les deux derniers liens : Tout d'abord le lien enseignants-lieux en croisant les disponibilités de chacun pour obtenir un planning des possibilités, puis le lien cours-créneaux en affectant un cours donné à un créneau, et donc à un lieu et un enseignant. Enfin, l'affectation des étudiants aux différents cours.

L'algorithme devait donc fournir un emploi du temps complet de la période choisie, c'est-à-dire un emploi du temps indiquant pour chaque cours, son lieu d'enseignement, son horaire, son professeur et ses étudiants. Il devait respecter aussi les différentes contraintes : volume horaire du cours, ressources disponibles, simultanéité d'un cours pour un étudiant, etc.

III. Outils utilisés

III.1. Langages

III.1.a. PHP

Acronyme récursif de "Hypertext Preprocessor", le PHP est un langage de script orienté objet principalement utilisé pour produire un site web dynamique.

Il est exécuté du côté du serveur et n'oblige pas les visiteurs à posséder des logiciels ou plugins particuliers, tant que l'hébergeur prend en compte ce langage.

Lors de l'exécution d'une page PHP, le serveur renvoie généralement au client (le visiteur du site) une page web qui peut contenir du code HTML, XHTML, CSS, JavaScript, ... [5]



III.1.b. JavaScript

Le JavaScript est un langage informatique utilisé dans les pages web. Il consiste en la création de scripts sur une page HTML qui permettent à cette page d'être dynamique sans avoir besoin de la recharger. Généralement, cela permet d'augmenter l'ergonomie de l'interface utilisateur ou parfois l'esthétique de la page.

Il est exécuté du côté du client, par le navigateur lui-même.

La technique AJAX (Asynchronous Javascript And XML) utilise largement le JavaScript dans le but d'interagir sur la page de manière dynamique [6].



Le terme CSS est l'acronyme de "Cascading Style Sheets". Il s'agit d'un langage informatique utilisé pour la mise en forme des fichiers HTML ou XML grâce à du code permettant de gérer l'apparence d'une page web.

L'avantage de l'utilisation d'un fichier CSS pour la mise en forme d'un site réside dans la possibilité de faire des modifications par type de balise html, ce qui permet de modifier d'un coup tous les éléments de même type sans devoir le faire élément par élément [7].



Le langage HTML est utilisé pour la création de pages web. Il est l'acronyme de "HyperText Markup Language", ce qui peut se traduire par "langage de balisage d'hypertexte". En effet, ce langage permet de réaliser de l'hypertexte à partir d'une structure de balisage.

Ce n'est pas à proprement parler un langage de programmation mais plutôt un langage de description pour la mise en forme du contenu. Le balisage permet de placer les différents éléments repris par la page HTML. Pour visualiser ces pages, il est nécessaire d'utiliser un navigateur web pour visualiser ces pages.

Généralement, une page HTML est associée à d'autres langages comme le CSS ou le JavaScript [8].



III.1.e. Rôle des langages dans le logiciel

Chaque langage possède donc un rôle propre dans le fonctionnement du logiciel. Le PHP, exécuté du côté du serveur, va générer une page HTML en fonction de la demande du client. Cette page va être envoyée au client et affichée par son navigateur. De plus, le navigateur internet du client va exécuter du JavaScript pour modifier le fonctionnement de la page HTML, et du CSS pour modifier son apparence (Figure 7).

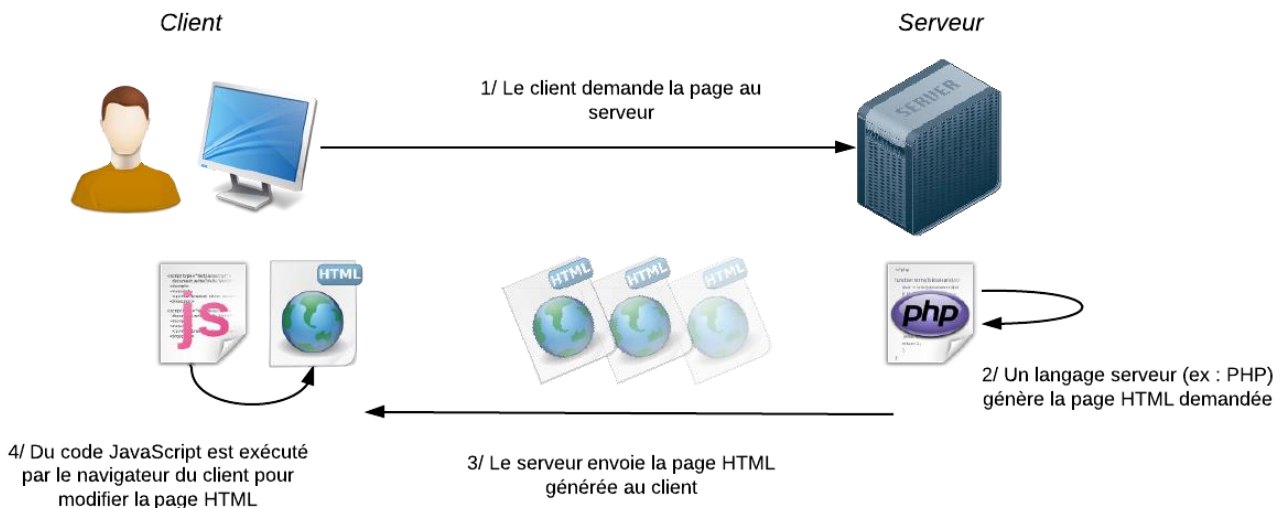


Figure 7 : Fonctionnement classique d'une application web

III.2. Framework

III.2.a. JQuery

JQuery est un Framework JavaScript sous licence libre facilite des fonctionnalités communes de JavaScript.

L'utilisation de cet ensemble de bibliothèques permet de gagner du temps de développement lors de l'interaction sur le code HTML d'une page web, l'AJAX ou la gestion des événements. JQuery possède par la même occasion l'avantage d'être utilisable par plusieurs navigateurs web (ex. Internet Explorer, Firefox, Chrome, Safari ou Opera).

JQuery s'utilise grâce à un fichier JavaScript qui peut être inférieur à 100 Ko lorsque le code est minimisé et compressé en Gzip [9].



III.2.b. Bootstrap

Bootstrap est une collection d'outils utile à la création du design (graphisme, animation et interactions avec la page dans le navigateur, etc. de sites et d'applications web. C'est un ensemble qui contient des codes HTML et CSS, des formulaires, boutons, outils de navigation et autres éléments interactifs, ainsi que des extensions JavaScript en option. C'est l'un des projets les plus populaires sur la plate-forme de gestion de développement GitHub [10].



III.2.c. Symfony

Symfony est un ensemble de composants PHP ainsi qu'un Framework MVC (cf. III.3) libre écrit en PHP. Il fournit des fonctionnalités modulables et adaptables qui permettent de faciliter et d'accélérer le développement d'un site web [11].



III.2.d. Doctrine

Doctrine est un ORM (couche d'abstraction à la base de données) pour PHP. Il s'agit d'un logiciel libre sous licence GNU LGPL.

Doctrine est l'ORM par défaut du Framework Symfony (depuis la version 1.3 de ce Framework). Cependant son utilisation dans le cadre d'un projet développé avec Symfony est optionnelle. De plus, il peut être utilisé avec de nombreux autres *Frameworks* tels que Zend Framework, CodeIgniter, FLOW3 ou encore Lithium [12].



Twig est un moteur de modèle (ou *templates*) PHP. Il s'agit du moteur défini de Symfony. Son intérêt réside dans l'utilisation du modèle MVC (cf. III.3), car il permet d'afficher des variables directement depuis les informations envoyées par le contrôleur.

Une autre fonctionnalité de Twig est la création de mises en forme (ou *layout*) pour la visualisation. On peut définir une disposition particulière pour une page, et la reprendre dans d'autres pages. Cela permet ainsi de définir une seule fois les éléments communs à tout le site et de les réutiliser simplement pour chaque page. Généralement, on utilise les *layouts* sur trois couches : la couche correspondant au site complet, celle correspondant au *bundle*, c'est-à-dire une fonctionnalité globale du site, puis celle correspondant à une fonctionnalité précise.



Par exemple, sur la Figure 8, on peut voir un *layout* global, (`::layout.html.twig`) qui contient un menu, une en-tête et un pied-de-page dans lequel on intègre un *layout* pour le *bundle* (`OCPlatformBundle::layout.html.twig`) et enfin, le *layout* le plus précis.

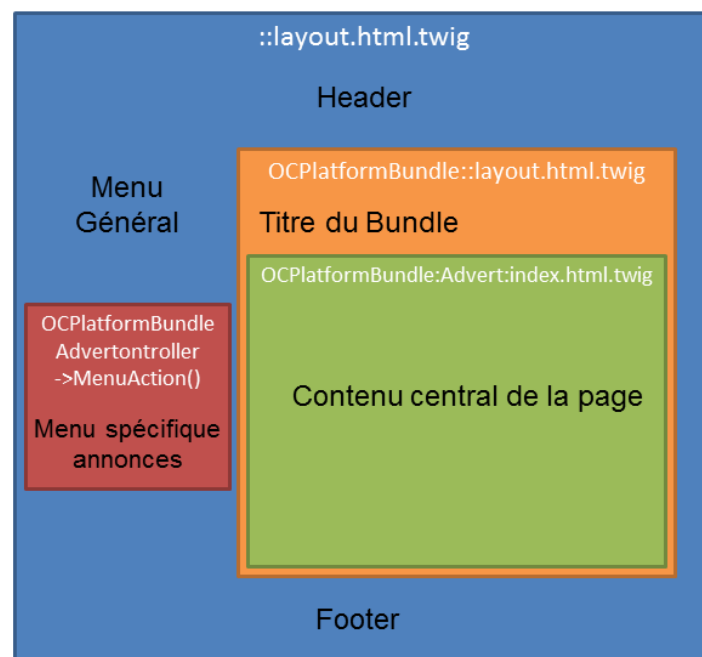


Figure 8 : Exemple d'utilisation d'un *layout*

Header : En-tête / *Footer* : Pied-de-page

III.3. Motif MVC

MVC est l'acronyme pour Modèle-Vue-Contrôleur. Il s'agit d'un modèle d'architecture (ou *design pattern*) ayant pour objectif la séparation des couches de présentation (UI : *User Interface*), métier (BLL : *Business Logic Layer*) et d'accès aux données (DAL : *Data Access Layer*). Cela permet d'avoir une dépendance minimale entre ces couches ; ainsi, les modifications effectuées sur n'importe quelle couche de l'application n'affectent pas les autres couches [13].

Chacune des couches possède un rôle différent (cf. Figure 9) :

- Modèle : c'est l'élément en contact avec les données et qui contient la logique de l'application ;
- Vue : c'est la partie visible de l'application, elle se charge de l'affichage des données que l'on lui fournit ;
- Contrôleur : c'est l'élément en charge du traitement des requêtes et de la redirection des données vers le modèle ou la vue.

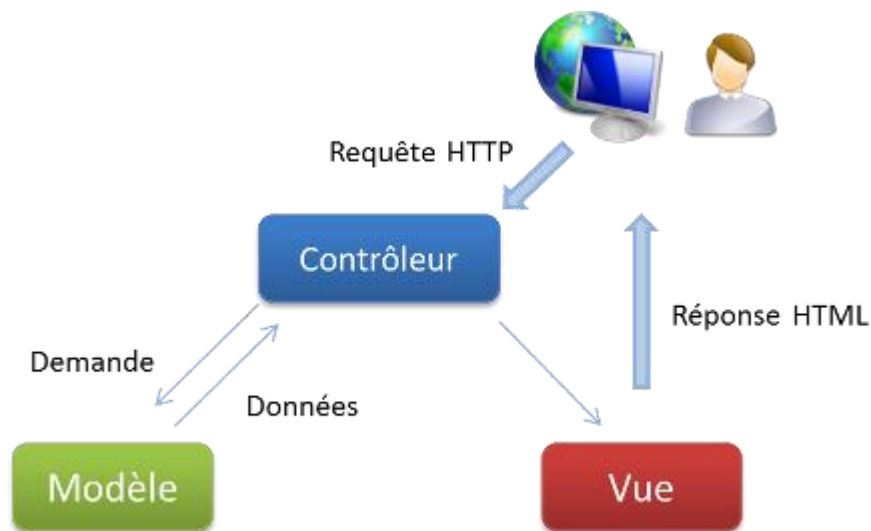


Figure 9 : Fonctionnement du motif Modèle-Vue-Contrôleur

IV. Conception

IV.1. L'interface utilisateur

C'est la partie du logiciel avec laquelle l'utilisateur sera amené à interagir. Elle doit donc être utilisable facilement par n'importe qui, quelle que soit son expérience en informatique. On parle d'interface ergonomique : c'est l'interface qui doit être adaptée et non l'utilisateur qui doit s'y adapter. Pour cela, il existe quelques règles à respecter, comme la règle des « trois clics », c'est-à-dire pouvoir accéder n'importe où dans le site depuis n'importe quelle page en 3 clics au maximum, ou encore avoir un menu clair et dégagé de superflu.

C'est dans cette optique d'ergonomie que j'ai opté pour certains choix au détriment d'autres. De plus, le renseignement des données doit être précis pour éviter tout problème dans le fonctionnement du logiciel : il faut donc guider l'utilisateur le plus possible afin d'éviter toute erreur de saisie.

Ainsi, j'ai commencé par concevoir la mise en page globale du site web, à savoir une barre de menu contenant le logo de l'UFR STAPS à gauche, et des liens de navigation à droite, un emplacement pour le corps de la page et un pied-de-page. Tous ces éléments sont communs à toutes les pages de l'application.

Ensuite, il fallait réfléchir aux meilleurs moyens de collecter des informations précises, ainsi qu'à celui de les visualiser, ce dernier dépendant de l'information elle-même. J'ai commencé par m'intéresser aux disponibilités des enseignants et j'ai opté pour un agenda. En effet, j'ai pensé qu'ajouter ses disponibilités via un formulaire s'affichant par un simple clic sur un agenda serait quelque chose d'intuitif. De plus, cette méthode utilise un principe de l'ergonomie, le skeuomorphisme (Figure 10), c'est-à-dire le fait de donner à une fonctionnalité l'apparence d'un objet ayant cette fonction dans la réalité (par exemple, donner la forme d'un micro à une application pour enregistrer des sons). Cette vue agenda possède aussi l'avantage de permettre l'affichage des disponibilités directement à l'ajout des événements, donnant un effet visuel très parlant.



Figure 10 : Quelques exemples de skeuomorphisme

Concernant les importations et exportations de fichier, j'ai estimé que des boutons seraient suffisamment explicites en les plaçant de manière organisée.

Enfin, le choix des cours pour les enseignants était quelque chose de très important à simplifier car c'est une action qui peut rapidement être ennuyeuse pour l'utilisateur. J'ai voulu faire un système de liste déroulante permettant de sélectionner les cours, mais aussi de filtrer les cours voulus en tapant directement au clavier. Cette méthode aide l'utilisateur à sélectionner ses cours en lui évitant un parcours redondant de tous les cours s'il connaît déjà ceux qu'il veut.

IV.2. La base de données

La conception de la base de données est quelque chose de très important. En effet, il faut que celle-ci soit à la fois facilement exploitable, mais aussi optimisée de sorte qu'elle ne contienne pas d'informations redondantes car une partie de la mémoire serait alors encombrée par des informations qui existent déjà.

Plusieurs tables ont dû être créées pour permettre le stockage des données utiles à l'algorithme.

IV.2.a. Utilisateur (User)

Il s'agit d'une table permettant la gestion des comptes utilisateurs (Figure 11). Initialement, cette table contient toutes les informations des utilisateurs, mais elle doit évoluer prochainement vers une table constituant un lien entre les utilisateurs du logiciels et les utilisateurs stockés dans la base de données de l'UCA.

CHAMP	TYPE	FACULTATIF ?	EXPLICATIONS
ID	Entier	NON	Identifiant
USERNAME	Chaîne de caractères	NON	Nom d'utilisateur
USERNAME_CANONICAL	Chaîne de caractères	NON	Nom d'utilisateur court
EMAIL	Chaîne de caractères	NON	E-mail
EMAIL_CANONICAL	Chaîne de caractères	NON	E-mail court
ENABLED	Booléen	NON	Activation du compte
SALT	Chaîne de caractères	OUI	Sel
PASSWORD	Chaîne de caractères	NON	Mot de passe (crypté)
LAST_LOGIN	Date - Temps	OUI	Date de la dernière connexion
CONFIRMATION_TOKEN	Chaîne de caractères	OUI	Jeton de confirmation
PASSWORD_REQUESTED_AT	Date - Temps	OUI	Date du dernier usage du mot de passe
ROLES	Chaîne de caractères	NON	Rôles du compte

Figure 11 : Table User

Le sel (*salt*) et le jeton de confirmation sont deux informations pour la sécurité :

- Le sel est une donnée servant à empêcher que deux informations différentes destinées à être hachée (méthode de cryptage) conduisent à la même empreinte ou résultat. On parle de salage. Le salage permet aussi de lutter contre les attaques par analyse fréquentielle (lorsque tente de décoder un message grâce aux fréquences d'apparition des lettres), les attaques utilisant des tables arc-en-ciel (structure de données servant à retrouver un mot de passe à partir de son empreinte), les attaques par dictionnaire (forcer un mot de passe en testant tous les mots du dictionnaire) et les attaques par force brute (forcer un mot de passe en essayant toutes les combinaisons possibles).

- Le jeton de confirmation est une information stockée lors de l’affichage de certains formulaires utilisée pour vérifier que l’utilisateur est bien passé par une page afin d’accéder à une autre page.

IV.2.b. Événements & famille d’événements (Event & event_parent)

Ces deux tables sont utilisées pour le stockage des événements à afficher sur l’agenda (Figure 12 et Figure 13). J’ai choisi de créer deux tables distinctes pour éviter de dupliquer inutilement les données des événements. En effet, dans le cas d’une table simple, chaque événement de l’agenda correspondrait à une ligne en base de données. Le fait d’avoir deux tables distinctes permet de regrouper les événements répétés sur une ligne, c’est-à-dire les événements hebdomadaires.

CHAMP	TYPE	FACULTATIF ?	EXPLICATIONS
ID	Entier	NON	Identifiant
EVENT_PARENT_ID	Entier	NON	Identifiant la famille d’événements
DATESTART	Date	NON	Date de début de l’évènement
DATEEND	Date	OUI	Date de fin de l’évènement

Figure 12 : Table Event

CHAMP	TYPE	FACULTATIF ?	EXPLICATIONS
ID	Entier	NON	Identifiant
USER_ID	Entier	OUI	Identifiant de l’utilisateur
LIEU_ID	Entier	Oui	Identifiant du lieu
CLASSNAME	Chaîne de caractères	NON	Type d’événement
TITLE	Chaîne de caractères	NON	Titre de l’événement
START	Temps	NON	Heure de début de l’événement
END	Temps	NON	Heure de fin de l’événement
DOW	Chaîne de caractères	NON	Jours de l’événement
ALLDAY	Booléen	NON	Evénement toute la journée ?
FREQUENCY	Chaîne de caractères	NON	Fréquence de l’événement

Figure 13 : Table Event_parent

IV.2.c. Enseignements (EC, UE & APSA)

Ces deux tables permettent le stockage des informations utiles extraites de la maquette (Figure 14, Figure 15 et Figure 16). Les informations sont importées de la maquette, et sont extraites pour remplir les listes des choix des cours pour les enseignants et pour générer l'emploi du temps.

CHAMP	TYPE	FACULTATIF ?
ID	Entier	NON
UE_ID	Entier	NON
INTITULE	Chaîne de caractères	OUI
NUMERO	Entier	OUI
EFFECTIF	Entier	NON
PARCOURS	Chaîne de caractères	OUI
HEURE	Entier	NON
TYPECOURS	Chaîne de caractères	NON

Figure 14 : Table EC

CHAMP	TYPE	FACULTATIF ?
ID	Entier	NON
INTITULE	Chaîne de caractères	NON
NUMERO	Entier	NON

Figure 15 : Table UE

CHAMP	TYPE	FACULTATIF ?
ID	Entier	NON
INTITULE	Chaîne de caractères	NON
CODE	Chaîne de caractères	NON

Figure 16 : Table APSA

IV.2.d. Lieux de pratique

Cette table permet de stocker les informations concernant les différents lieux de pratique des APSA (Figure 17).

CHAMP	TYPE	FACULTATIF ?
ID	Entier	NON
CODE	Chaîne de caractères	NON
LOCALITE	Chaîne de caractères	NON
INSTALLATION	Chaîne de caractères	NON
ZONEACTIVITE	Chaîne de caractères	NON
CAPACITE	Entier	NON

Figure 17 : Table Lieu

IV.2.e. Liens entre les tables

La Figure 18 montre le diagramme de classe qui représente les liens entre les tables de la base de données. Ainsi, la maquette est représentée via la composition entre les UE et les EC. Les associations avec les événements correspondant aux disponibilités des lieux et aux disponibilités des utilisateurs sont représentées par les liens avec EventParent.

Par ailleurs, les utilisateurs sont liés aux EC pour symboliser les choix d'enseignements, et les lieux sont associés aux EC pour représenter les salles dans lesquelles on peut dispenser ces EC.

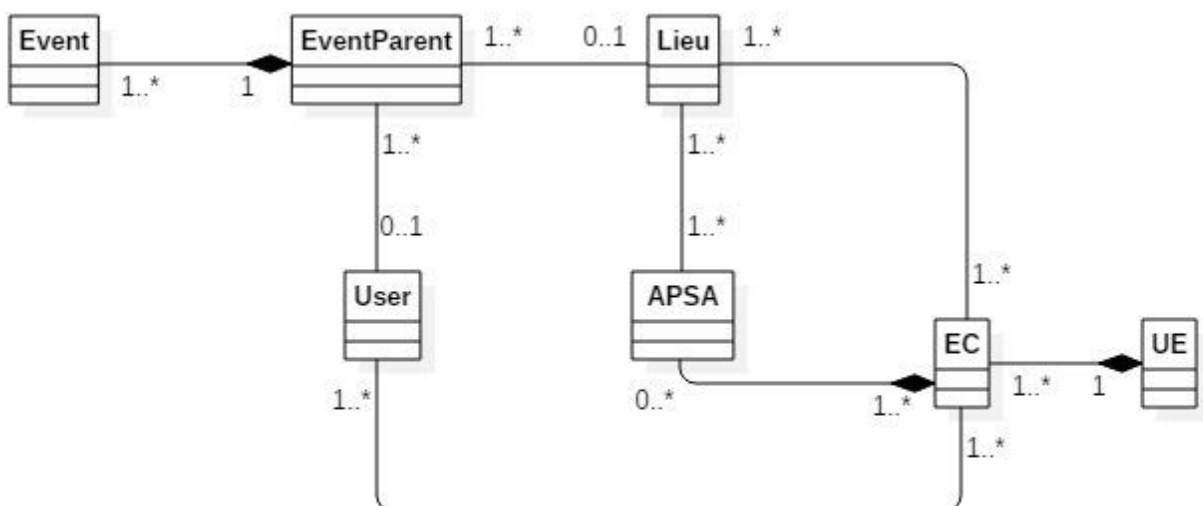


Figure 18 : Diagramme de classe des entités du logiciel

IV.3. L'algorithme de création

La conception de l'algorithme a été une phase plutôt complexe. Je savais vers où me diriger, mais pas d'où partir, ni comment y aller.

En fait, l'implémentation de l'interface m'a permis de savoir quelles étaient les données que je pourrais utiliser pour générer l'emploi du temps, mais il y avait besoin d'ajouter quelques degrés de liberté pour produire différents emplois du temps avec les mêmes données. Ainsi, il a fallu commencer par estimer les paramètres utiles. Le premier paramètre correspondait au choix du semestre pour lequel l'utilisateur voulait générer l'emploi du temps. Ce choix permettait de réduire le nombre d'EC à traiter, et donc de gagner du temps lors du traitement de l'algorithme.

Un autre paramètre a été mis à jour lors de la présentation de l'application (29/05/17), à savoir la durée d'un créneau qui est devenu indispensable suite au changement de la durée des cours qui doit passer de 2h à 1h30 à la rentrée 2017-2018.

Enfin, il était important pour le logiciel de connaître les capacités maximales des groupes de CM (300 étudiants), de TD (40 étudiants) et de TP (20 étudiants) ainsi qu'une estimation du nombre d'étudiants par année et par parcours.

Une fois ces paramètres fixés, on connaissait le point départ et le résultat souhaité de l'algorithme, ainsi que le moyen de s'y rendre : il ne restait plus qu'à définir le chemin. Pour cela, il existe plusieurs méthodes.

Une première méthode possible aurait consisté à partir d'un ancien emploi du temps et à en faire un nouveau en cherchant à s'approcher le plus possible de l'ancien. Cette solution possède l'avantage d'être plus économe en calculs et donc en temps d'exécution. Malheureusement, elle est difficile à mettre en place, non seulement à cause du changement de durée d'un créneau, mais aussi en cas de changement de la maquette elle-même, voire des salles et des enseignants disponibles.

Ainsi, je me suis dirigé naturellement vers une heuristique d'insertion, c'est-à-dire la création d'une solution au fil de l'eau. Pour augmenter l'optimalité de cette solution, j'ai rapidement décidé de trier les EC à placer selon un critère de contrainte : placer en premier les cours ayant le plus d'étudiants et le plus d'heures. Le principe est similaire à celui d'un déménagement, il faut placer

les grosses boîtes en premier dans le camion, car les plus petites arriveront à se glisser plus facilement dans les interstices. Ensuite, une fois l'ordre de traitement fixé, il suffisait de placer les différents cours sur différents créneaux pour chaque semaine de la génération.

Pour faire cela, j'ai tout d'abord commencé par me demander s'il valait mieux raisonner par semaine puis par cours, ou l'inverse. En effet, il serait préférable que les emplois du temps hebdomadaires soient similaires d'une semaine sur l'autre. J'ai commencé par essayer de générer pour chaque cours, un agenda regroupant les possibilités de créneaux pour toute la période de génération à partir de ceux définis pour chaque semaine. Cette méthode aurait permis de voir les créneaux en commun sur toute la période, et donc de placer ces cours aux mêmes moments chaque semaine. Cependant, cette approche comportait un problème : s'il n'y avait aucun créneau commun sur toute la période, le cours ne pouvait tout simplement pas être placé. *A contrario*, avec le second raisonnement, les emplois du temps hebdomadaires restent similaires car le traitement et l'ordre des cours est identique chaque semaine.

Pour la suite, j'ai fait en sorte de minimiser les conflits entre les cours à chaque choix. J'ai donc décidé de commencer par placer les cours n'ayant aucun degré de liberté sur les créneaux. En effet, pour chaque cours, on peut déterminer un nombre de groupes à placer par semaine en fonction des effectifs et des capacités. De plus, on peut déterminer le nombre de créneaux dans lesquels on peut placer ce cours. Dans certains cas, on doit placer le même nombre de groupes que de créneaux disponibles, ce qui conduit à une solution unique. En fixant ces cours dès le début, il est sûr de pouvoir les placer, alors qu'en les plaçant après d'autres choix, il existe le risque de prendre un créneau indispensable à ce cours, qui ne pourra plus être placé ensuite.

Pour les cours offrant davantage de possibilités, l'algorithme possède deux degrés de liberté : le premier se situe sur les choix des créneaux, le second sur les choix des couples enseignants-salles d'un créneau particulier. Pour minimiser les conflits, l'algorithme choisit les créneaux qui possèdent le moins de choix d'EC, puis les couples de lieux et d'enseignants dont chaque élément possède le plus petit catalogue de choix de cours.

V. Réalisation

Le projet s'est déroulé en deux parties principales, à savoir la création de l'interface utilisateurs et l'algorithme de création des emplois du temps.

V.1. L'interface utilisateurs

Il s'agit de la première partie importante du projet, sans laquelle il aurait été difficile de faire la suite. En effet, c'est ici que nous permettons aux utilisateurs de saisir ou d'importer les différentes données nécessaires au fonctionnement de l'algorithme de création. Les principaux utilisateurs de cette application seront l'administrateur en charge de la création des emplois du temps et les responsables des années et des parcours.

Depuis cette interface, les enseignants peuvent renseigner leurs disponibilités grâce à un agenda. Un clic sur l'agenda ouvre un *pop-up* permettant de créer un créneau de disponibilité en remplissant le formulaire avec les heures de début et de fin, la fréquence de ce créneau (simple ou hebdomadaire), les jours de la semaine associés à ce créneau et enfin, la période durant laquelle ce créneau existe. La validation de ce créneau entraîne son ajout dans la base de données. En sélectionnant par un clic un événement existant, on peut le supprimer ou le modifier, et choisir si cette opération doit s'appliquer à tous les événements de ce type ou à uniquement à celui qui est sélectionné.

Les enseignants peuvent aussi indiquer les cours dans lesquels ils souhaitent enseigner. Pour cela, ils peuvent sélectionner plusieurs UE dans une première liste qui sert à filtrer les EC. Ils peuvent ensuite choisir les EC dans une seconde liste et valider leurs choix qui seront sauvegardés dans la base de données.

L'administrateur a accès à davantage de fonctionnalités. Tout d'abord, certaines fonctionnalités lui permettent de superviser les données : il peut afficher les disponibilités des enseignants en filtrant par nom, par lieu ou par activité. De plus, il possède une vue d'ensemble lui reportant les potentielles erreurs qui peuvent exister, comme par exemple un manque de personnel pour un cours donné.

D'autres fonctionnalités lui permettent d'apporter certaines données nécessaires : il peut importer les données relatives aux disponibilités des lieux qui seront enregistrées et affichées sur l'agenda. Il peut aussi importer la maquette qui est utile à la fois pour l'algorithme, mais aussi pour renseigner les listes des cours dans lesquelles choisissent les enseignants. Ces données insérées doivent être complètes à chaque importation de la maquette car le logiciel se base sur les dernières informations fournies, effaçant les anciennes pour ne pas surcharger la mémoire inutilement. Cependant, pour éviter que les enseignants aient à ressaisir leurs cours à chaque changement de maquette, il est possible d'exporter les choix actuels et d'importer ce fichier une fois la maquette changée, réaffectant les cours en commun sur les deux maquettes aux professeurs les ayant sélectionnés.

Cette interface a pour but d'être ergonomique et simple à utiliser afin que n'importe quelle catégorie d'utilisateur puisse s'en servir facilement.

Une fois toutes les données insérées de manière correcte, l'administrateur peut générer l'emploi du temps en modifiant certains paramètres, grâce à l'algorithme de création.

V.2. L'algorithme de création

L'algorithme de génération des emplois du temps pioche dans les paramètres et les données que l'utilisateur lui a fourni afin de produire un emploi du temps le plus complet possible (cf. Figure 19).

Entrées :

Paramètres :

- Le semestre de génération de l'emploi du temps ;
- L'effectif estimé par licence et parcours ;
- La durée d'un cours ;
- Les capacités d'un groupe de CM, de TP et de TD ;
- Les années de licence de génération de l'emploi du temps.

Données :

- La liste des EC et des APSA;
- Les cours choisis par les enseignants ;
- Les disponibilités des enseignants ;
- Les lieux d'enseignement d'un cours ;
- Les disponibilités des lieux.

Algorithme :

Récupération des EC utiles à la génération de l'emploi du temps choisi

ON FILTRE LA LISTE DES EC POUR NE GARDER QUE CEUX DU SEMESTRE CHOISI ET DES ANNEES CHOISIES.

ON TRIE CETTE LISTE PAR NOMBRE D'APSA, NOMBRE DE PARCOURS, EFFECTIF MAXIMUM ET HEURES DANS LE SEMESTRE SELON UN ORDRE DECROISSANT.

Boucle principale

POUR CHAQUE SEMAINE :

Premier tour : Affectation des APSA

POUR CHAQUE EC :

POUR CHAQUE APSA :

POUR CHAQUE JOUR :

POUR CHAQUE CRENEAU :

ON RECUPERE LES ID DES INSTALLATIONS DANS LESQUELLES ON PEUT DISPENSER CETTE APSA ET QUI SONT DISPONIBLES A CE CRENEAU.

ON RECUPERE LES ID DES ENSEIGNANTS QUI PEUVENT DISPENSER CETTE APSA ET QUI SONT DISPONIBLES A CE CRENEAU.

ON FILTRE CES LISTES PAR RAPPORT AUX APSA DEJA PLACES A CE CRENEAU.

FIN POUR CHAQUE CRENEAU.

FIN POUR CHAQUE JOUR.

ON CALCULE LE NOMBRE DE GROUPES A CONSTITUER POUR CETTE APSA.

ON AFFECTE CHACUN DE CES GROUPES A UN CRENEAU, UNE INSTALLATION ET UN ENSEIGNANT. S'IL MANQUE L'ENSEIGNANT, ON LE REMPLACE PAR UN ENSEIGNANT « LAMBDA », QUI SERA A TROUVER PLUS TARD. DE MEME POUR LES INSTALLATIONS.

FIN POUR CHAQUE APSA.

FIN POUR CHAQUE EC.

Second tour : Affectation des EC sans APSA à solution unique

POUR CHAQUE EC :

POUR CHAQUE JOUR :

POUR CHAQUE CRENEAU :

S'IL Y A PAS D'EC/APSA DE LA MEME ANNEE ET DU MEME PARCOURS DEJA PLACE A CE CRENEAU :

ON RECUPERE LES ID DES SALLES DANS LESQUELLES ON PEUT DISPENSER CET EC ET QUI SONT DISPONIBLES A CE CRENEAU.

ON RECUPERE LES ID DES ENSEIGNANTS QUI PEUVENT DISPENSER CET EC ET QUI SONT DISPONIBLES A CE CRENEAU.

ON FILTRE CES LISTES PAR RAPPORT AUX EC DEJA PLACES A CE CRENEAU.

FIN SI.

FIN POUR CHAQUE CRENEAU.

FIN POUR CHAQUE JOUR.

ON CALCULE LE NOMBRE DE GROUPES A CONSTITUER ET LE NOMBRE DE CRENEAUX LIBRES DANS CETTE SEMAINE POUR CET EC.

S'IL Y A PLUS DE GROUPES A CONSTITUER QUE DE CRENEAU DISPONIBLES :

L'EC N'A PAS D'UNIQUE SOLUTION, ON PASSE AU COURS SUIVANT SANS L'AFFECTER.

SINON S'IL Y A AUTANT DE GROUPES A CONSTITUER QUE DE CRENEAUX DISPONIBLES :

L'EC A UNE UNIQUE SOLUTION, ON AFFECTE CHAQUE GROUPE DE L'EC A UN CES CRENEAUX.

SINON S'IL Y A MOINS DE GROUPES A CONSTITUER QUE DE CRENEAUX DISPONIBLES :

L'EC VA POSER PROBLEME, ON LE STOCKE DANS UNE NOUVELLE LISTE POUR LE TRAITER PLUS TARD.

FIN SI.

FIN POUR CHAQUE EC.

Troisième tour : Affectation des EC sans APSA à solution multiple

Partie 1 : Agenda des créneaux disponibles de ces EC

POUR CHAQUE JOUR :

POUR CHAQUE CRENEAU :

POUR CHAQUE EC :

SI LE COURS PEUT ETRE FAIT A CE CRENEAU :

ON STOCKE SON ID DANS UN TABLEAU POUR TRAITEMENT ULTERIEUR.

FIN SI.

FIN POUR CHAQUE EC.

FIN POUR CHAQUE CRENEAU.

FIN POUR CHAQUE JOUR.

Partie 2 : Affectation des EC en limitant les conflits

POUR CHAQUE EC :

POUR CHAQUE GROUPE DE L'EC A CONSTITUER :

PARMI LES CRENEAUX DISPONIBLES DE L'EC, ON CHOISIT CELUI SUR LEQUEL LE MOINS D'EC PEUVENT ETRE DISPENSES.

PARMI LES SALLES DISPONIBLES A CE CRENEAU, ON CHOISIT CELLE DANS LAQUELLE LE MOINS D'EC PEUVENT ETRE DISPENSES.

ON CHOISIT L'ENSEIGNANT DE LA MEME MANIERE.

ON AFFECTE L'EC AU CRENEAU CHOISI AVEC LA SALLE ET L'ENSEIGNANT (QUI SONT « LAMBDA » SI AUCUN N'A ETE TROUVE).

FIN POUR CHAQUE GROUPE.

FIN POUR CHAQUE EC.

Quatrième tour : Affectation des EC restants

POUR CHAQUE EC :

POUR CHAQUE GROUPE DE L'EC A CONSTITUER :

ON AFFECTE LE GROUPE DE L'EC AU CRENEAU LE MOINS REMPLI EN LUI ATTRIBUANT UNE SALLE ET UN ENSEIGNANT SI POSSIBLE, « LAMBDA » SINON.

FIN POUR CHAQUE GROUPE.

FIN POUR CHAQUE EC.

FIN POUR CHAQUE SEMAINE.

ON RETOURNE LES AFFECTATIONS STOCKEES.

Figure 19 : Algorithme de génération des emplois du temps

V.3. Fichiers de données

La plupart des données utilisées par le logiciel proviennent de différents fichiers importés. Certains de ces fichiers existaient dès le début du projet et étaient donc exploitables tels quels, tandis que d'autres étaient inexistantes et ont dû être normalisés afin d'être utilisables.

V.3.a. La maquette

Ce fichier existe depuis longtemps au sein de l'UFR STAPS. Il s'agit du fichier contenant toutes les informations à propos des UE et des EC. Le logiciel se sert d'une partie des informations du fichier pour générer l'emploi du temps et remplir la liste des choix de cours pour les enseignants.

Cependant, le fichier a été légèrement modifié afin d'ajouter l'information concernant les APSA existantes dans un EC. Par conséquent, pour être utilisable par le logiciel, le fichier doit contenir, au minimum les colonnes suivantes :

- « n°UE », qui doit être un nombre ;
- « Intitulé UE », qui doit être une chaîne de caractère ;
- « Intitulés EC », qui doit être une chaîne de caractère ;
- « APSA », qui doit être un intitulé court du fichier des APSA, séparés par une virgule suivie d'un espace s'il y en a plusieurs ;
- « CM », qui doit être un nombre ;
- « TD », qui doit être un nombre ;
- « TP », qui doit être un nombre.

De plus, le fichier doit contenir une feuille par licence et par semestre, ainsi qu'une feuille de présentation qui doit être la première. On obtient alors sept feuilles, dans l'ordre, « Général » puis « S1 » à « S6 ».

V.3.b. Les APSA

Ce fichier était inexistant et a dû être totalement créé et normalisé. De plus, il a fallu récolter la totalité des informations qui ont rempli ce fichier. Le logiciel se sert de ce fichier pour connaître les différentes APSA. Le logiciel peut utiliser ce fichier s'il contient les colonnes :

- « APSA », une chaîne de caractères pour le nom de l'APSA ;
- « Intitulé abrégé », une chaîne de caractères formatée pour savoir si l'APSA est pratique ou théorique, sa famille puis ce qu'elle est (exemple : La pratique de natation a pour intitulé abrégé la chaîne « Theo_Aqua_Nat »).

L'intitulé abrégé doit aussi exister pour les familles d'APSA, ce qui permet de pouvoir sélectionner toute une famille grâce à un seul intitulé.

V.3.c. La réservation des installations

Ce fichier existait en partie dès le début de projet en tant que récapitulatif des réservations des différentes installations sportives nécessaires aux APSA. Il a été repris et modifié pour être plus adapté à son rôle de fichier d'importation, et permet au logiciel d'enregistrer les différentes installations en base de données, ainsi que leurs périodes de réservation. Le fichier doit contenir les colonnes suivantes pour être utilisable par le logiciel :

- « Localité », chaîne de caractères indiquant la localité de l'installation ;
- « Nom installation », chaîne de caractères indiquant le nom de l'installation ;
- « Zone d'activité », chaîne de caractères indiquant la zone d'activité de l'installation ;
- « Code final », chaîne de caractères formaté permettant l'identification précise d'une installation ;
- « Capacité », nombre indiquant la capacité de l'installation ;
- « Jour souhaité », chaîne de caractères indiquant le jour de réservation ;
- « Horaires », chaîne de caractères formatée indiquant les horaires de la réservation. Doit être sous la forme « HHhMM à HHhMM » (exemple : « 8h30 à 11h30 ») ;
- « Début période », chaîne de caractères formatée indiquant la date de début de la période de réservation. Doit être sous la forme « jj/mm/aa » (exemple : « 07/09/17 ») ;
- « Fin période », chaîne de caractères formatée comme précédemment indiquant la date de fin de la période de réservation.

Chaque feuille du fichier doit contenir les réservations d'une année de licence.

V.3.d. La correspondance entre les installations et les APSA

Ce fichier était inexistant et a nécessité une collecte d'informations qui n'existaient pas de manière écrite. Son rôle est de fournir au logiciel une base utilisable pour connaître les différents lieux dans lesquels une APSA peut être enseignée.

Ainsi, le fichier se présente sous la forme d'un tableau à double entrée contenant les intitulés courts des APSA sur les lignes et les codes des installations dans les colonnes. Si une APSA peut être pratiquée dans une installation, un « X » est placé dans la case à l'intersection de cette APSA et de cette installation.

VI. L'application résultante

VI.1. Présentation des résultats

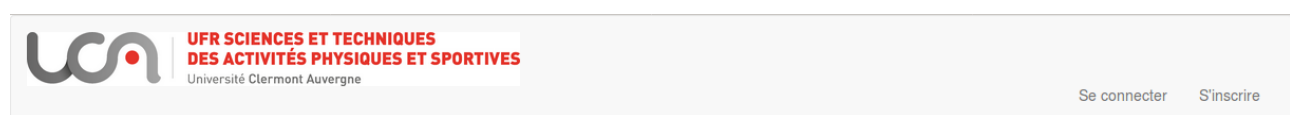
VI.1.a. Page d'accueil

Cette page est la page d'accueil. Elle n'a rien de particulier mais elle permet de voir le *layout* global de l'application. De plus, on remarque aussi qu'en fonction du rôle que l'on possède, on a accès à des pages différentes.

Une personne non connectée (Figure 20) peut s'inscrire ou se connecter au site.

L'administrateur (Figure 21) peut accéder aux pages de l'agenda, d'importation des données et de génération de l'emploi du temps.

Un enseignant (Figure 22) peut quant à lui accéder aussi à la page d'agenda, mais également aux pages de choix des cours.

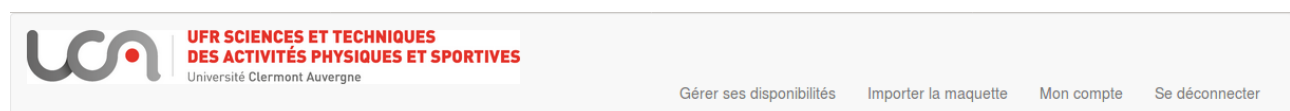


Ceci est la page d'accueil

Elle permettra à un utilisateur de savoir ce qu'est le site et de se connecter ou s'inscrire.

Dernière actualisation : 07/07/2017 à 13:31:43.

Figure 20 : Page d'accueil (non connecté)

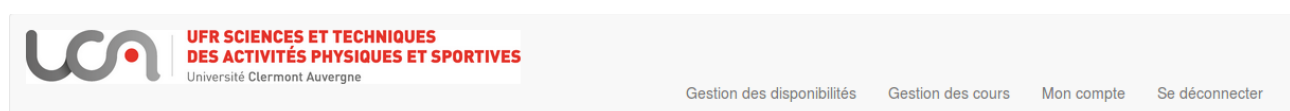


Ceci est la page d'accueil

Elle permettra à un utilisateur de savoir ce qu'est le site et de se connecter ou s'inscrire.

Dernière actualisation : 07/07/2017 à 13:44:25.

Figure 21 : Page d'accueil (côté administrateur)



Ceci est la page d'accueil

Elle permettra à un utilisateur de savoir ce qu'est le site et de se connecter ou s'inscrire.

Dernière actualisation : 07/07/2017 à 13:58:03.

Figure 22 : Page d'accueil (côté enseignant)

VI.1.b. Pages d'inscription et de connexion

Ces pages permettent à l'utilisateur de s'inscrire (Figure 23) et de s'authentifier (Figure 24).

UCA UFR SCIENCES ET TECHNIQUES
DES ACTIVITÉS PHYSIQUES ET SPORTIVES
Université Clermont Auvergne

Se connecter S'inscrire

Adresse e-mail

Nom d'utilisateur

Mot de passe

Répéter le mot de passe

Créer un compte

Dernière actualisation : 07/07/2017 à 13:44:11.

Figure 23 : Page d'inscription (non connecté)

UCA UFR SCIENCES ET TECHNIQUES
DES ACTIVITÉS PHYSIQUES ET SPORTIVES
Université Clermont Auvergne

Se connecter S'inscrire

Nom d'utilisateur Mot de passe ☐ Se souvenir de moi Connexion


Dernière actualisation : 07/07/2017 à 13:43:57.

Figure 24 : Page de connexion (non connecté)

VI.1.c. Page de l'agenda

Cette page possède une double fonction : elle permet à la fois une entrée d'informations, que ce soient les disponibilités des enseignants de leur côté ou l'import des disponibilités des salles pour l'administrateur, et elle permet aussi de visualiser toutes les informations d'événements. Ainsi, les enseignants peuvent visualiser leurs disponibilités et leur emploi du temps, tandis que l'administrateur peut voir l'agenda généré, les disponibilités des enseignants et les disponibilités des salles.

Pour plus de clarté, l'administrateur possède une fonctionnalité pour filtrer les différents événements pour n'afficher que ceux qui l'intéresse. Enfin, l'administrateur peut exporter ces événements et tous les effacer (Figure 25).



**UFR SCIENCES ET TECHNIQUES
DES ACTIVITÉS PHYSIQUES ET SPORTIVES**
 Université Clermont Auvergne

Gestion des disponibilités Importation des données Génération de l'emploi du temps Mon compte Se déconnecter

Tous ▾ Tous ▾
 Tous ▾ Tous ▾

Exporter
 Imprimer

Importer créneaux lieu (fichier ods, xls ou xml) :
 Parcourir... Aucun fichier sélectionné.
 Envoyer

Tout effacer

3 – 9 juil. 2017

Aujourd'hui ◀ ▶

	lun. 3/7	mar. 4/7	mer. 5/7	jeu. 6/7	ven. 7/7	sam. 8/7	dim. 9/7
07:00							
07:30							
08:00							
08:30							
09:00							
09:30							
10:00							
10:30							
11:00							
11:30							
12:00							
12:30							
13:00							
13:30							
14:00							

Figure 25 : Page de l'agenda (côté administrateur)

Si l'enseignant est responsable d'un parcours, il peut importer un fichier permettant de renseigner les disponibilités de plusieurs enseignants durant une période donnée (Figure 26).

Importer disponibilités enseignants :

Début de la période : Fin de la période (exclu) :

Aucun fichier sélectionné.

3 – 9 juil. 2017

Aujourd'hui



	lun. 3/7	mar. 4/7	mer. 5/7	jeu. 6/7	ven. 7/7	sam. 8/7	dim. 9/7
07:00							
07:30							
08:00							
08:30							
09:00							
09:30							
10:00							
10:30							
11:00							
11:30							
12:00							
12:30							
13:00							
13:30							
14:00							
14:30							
15:00							
15:30							
16:00							
16:30							
17:00							
17:30							
18:00							
18:30							
19:00							
19:30							
20:00							
20:30							
21:00							
21:30							
22:00							
22:30							
23:00							
23:30							
24:00							

Figure 26 : Page de l'agenda (côté enseignant)

VI.1.d. Page d'importation de la maquette

L'administrateur peut importer les différents fichiers grâce à cette page (Figure 27), à savoir les réservations des installations, les APSA, les correspondances entre les APSA et les installations, et enfin la maquette. Cependant, il faut noter que tout changement dans la maquette se répercute sur les choix faits par les enseignants et les efface. Par conséquent, l'administrateur peut exporter ces choix afin de les sauvegarder, puis il peut importer les choix sauvegardés avec la nouvelle maquette, afin de réattribuer les choix antérieurs aux bonnes personnes.

UFR SCIENCES ET TECHNIQUES
DES ACTIVITÉS PHYSIQUES ET SPORTIVES
Université Clermont Auvergne

Gestion des disponibilités Importation des données Génération de l'emploi du temps Mon compte Se déconnecter

Avant de charger de nouvelles données, vous pouvez exporter les choix d'EC actuels.
Exporter les choix actuels

Chaque fichier doit être au format ods, xls ou xml et doit être normalisé correctement.
Pour que l'emploi du temps généré soit le plus précis possible, il est conseillé de fournir le plus de données possibles.

****Réservations des installations :**
Parcourir... Aucun fichier sélectionné.

****APSA :**
Parcourir... Aucun fichier sélectionné.

****Correspondances APSA-installations :**
Parcourir... Aucun fichier sélectionné.

***Maquette :**
Parcourir... Aucun fichier sélectionné.

Importer les choix :
Parcourir... Aucun fichier sélectionné.

Charger les données

* : obligatoire
** : a fournir ensemble
Dernière actualisation : 07/07/2017 à 13:56:08.

Figure 27 : Page d'importation des données (côté administrateur)

VI.1.e. Page du choix des cours

Cette page plutôt simple permet aux enseignants de choisir les EC dans lesquels ils veulent enseigner (Figure 28). Pour cela, il suffit de choisir, dans la première liste, les UE qui contiennent les EC que l'on veut dispenser. Les EC s'ajouteront dans la liste des choix, juste en-dessous. Si l'EC sélectionné contient des APSA, on peut les choisir de la même manière dans la dernière liste.

La validation des choix s'effectue en cliquant sur le bouton.

UFR SCIENCES ET TECHNIQUES
DES ACTIVITÉS PHYSIQUES ET SPORTIVES
Université Clermont Auvergne

Gestion des disponibilités Gestion des cours Mon compte Se déconnecter

1. Choix des UE

2. Choix des EC

3. Choix des APS

Valider

Dernière actualisation : 07/07/2017 à 13:58:25.

Figure 28 : Page du choix des cours (côté enseignant)

VI.1.f. Pages de génération de l'emploi du temps

Cette page permet de renseigner les différents paramètres utiles à la génération de l'emploi du temps avant de pouvoir lancer le traitement de l'algorithme de création (Figure 29).

La seconde permet la visualisation et la modification de l'emploi du temps généré (Figure 30).

The screenshot shows the 'Génération de l'emploi du temps' page. At the top is the header for 'UFR SCIENCES ET TECHNIQUES DES ACTIVITÉS PHYSIQUES ET SPORTIVES' at 'Université Clermont Auvergne'. Navigation links include 'Gestion des disponibilités', 'Importation des données', 'Génération de l'emploi du temps' (active), 'Mon compte', and 'Se déconnecter'.

The main form is titled 'Placement des cours' and contains the following fields and options:

- ☐ Placer tous les cours ☐ Ne placer que les cours "sûrs"
- Durée d'un créneau: [text input]
- Date de début: [text input]
- Date de fin: [text input]
- Semestre: ☐ Semestre 1 ☐ Semestre 2
- Licence(s): ☒ L1 ☒ L2 ☒ L3
- Capacité d'un CM: [spin box with value 200]
- Capacité d'un TD: [spin box with value 40]
- Capacité d'un TP: [spin box with value 20]
- Effectif prévu pour la L1: [spin box with value 400]
- Effectif prévu pour la L2: [spin box with value 200]
- Effectif prévu pour la L3: [spin box with value 200]
- [Lancer la génération button]

Dernière actualisation : 07/07/2017 à 13:56:19.

Figure 29 : Page d'entrée des paramètres (côté administrateur)



Figure 30 : Page de génération de l'emploi du temps(côté administrateur)

VI.2. Utilisation de l'application

Pour que l'application fonctionne correctement, il est nécessaire de suivre quelques indications sans lesquelles il est fort possible de n'obtenir aucun résultat.

Pour commencer, il faut fournir à l'application le minimum des données dont elle a besoin, à savoir la maquette (importée depuis le fichier décrit en V.3.a), qui est la matière première de

l'emploi du temps. Cependant, en n'entrant que la maquette dans l'application, l'emploi du temps résultant n'aura aucune précision à cause de l'absence de contraintes.

Si l'on veut un emploi du temps qui soit un minimum organisé et exploitable, il faut fournir à l'application, en plus de la maquette, les informations suivantes :

- Les APSA (importées depuis le fichier décrit en V.3.b), qui permettent d'ajouter des éléments à placer ;
- Les installations et leurs disponibilités (importées depuis le fichier décrit en V.3.c) ainsi que les correspondances entre les installations et les APSA (importées depuis le fichier décrit en V.3.d), afin de placer correctement les APSA au sein de l'emploi du temps ;
- Les enseignants et leurs disponibilités ainsi que les correspondances entre les cours et les enseignants, pour pouvoir affecter des enseignants précis à des cours et affecter ces cours à des créneaux communs avec les enseignants.

Plus le nombre d'informations est important et plus le moule servant à faire l'emploi du temps sera précis. Ainsi, le logiciel fournira un emploi du temps d'autant plus rigoureux. Tous ces fichiers peuvent être importés par l'administrateur depuis la page « Importation des données » (cf. VI.1.d).

Une fois ces informations insérées dans l'application, il suffit de se rendre sur la page « Génération de l'emploi du temps » (cf. VI.1.f) sur laquelle se trouve un petit formulaire à remplir, servant à fournir quelques paramètres supplémentaires :

- La durée d'un créneau ;
- La période de la génération ;
- Le semestre de la génération ;
- Les années de licence dont on va générer l'emploi du temps ;
- Les capacités d'un groupe de CM, de TD et de TP ;
- Les effectifs prévisionnels par licence et par parcours.

Une fois le formulaire rempli, il suffit de cliquer sur le bouton « Lancer la génération » et d'attendre la fin du traitement. Une fois celui-ci terminé, l'emploi du temps complet est affiché et il est possible de filtrer son affichage pour voir uniquement l'emploi du temps d'une année de licence.

VI.3. Amélioration de l'application

Même si l'application est fonctionnelle comme demandé, la durée du stage n'a pas été suffisante pour faire tout ce que j'avais prévu et plusieurs points intéressants sont améliorables ou à ajouter.

Tout d'abord, l'interface graphique mérite une esthétique plus jolie. En effet, comme cette application va être régulièrement utilisée, il serait préférable de la rendre attrayante afin que tous les types d'utilisateurs ne soient pas rebutés lors de l'usage. De plus, il reste possible d'améliorer l'ergonomie du logiciel rien qu'en le rendant plus agréable à la vue.

Il est ensuite possible d'implémenter plusieurs nouvelles fonctionnalités qui augmenteraient l'utilité du logiciel, ou qui sont simplement pratiques. Ainsi, on pourrait implémenter un moyen de comptabiliser les heures effectuées par les différents enseignants, ce qui aiderait grandement à comptabiliser les heures de service grâce à la centralisation des données. On pourrait aussi implémenter un moyen d'imprimer directement les emplois du temps depuis le logiciel.

L'algorithme peut aussi être amélioré en ajoutant des méthodes heuristiques qui échangent différents créneaux de cours de manière à optimiser les placements des cours mais aussi les données qui concernent les enseignants et les installations.

Conclusion

L'objectif de mon stage était la réalisation d'un logiciel capable de générer un emploi du temps complet pour l'UFR STAPS de Clermont-Ferrand à partir de certaines données, dans le but de simplifier la tâche de l'administrateur responsable de ce travail ainsi que de fournir des résultats pouvant être optimisés.

L'application produite au cours de ce stage est actuellement capable de traiter ce problème de manière plus ou moins précise en fonction des données qui lui sont fournies. En effet, l'application peut tout à fait générer un emploi du temps en ne connaissant que la maquette, mais celui-ci ne sera rien de plus que l'ensemble des cours placés les uns à la suite des autres. Cependant, si d'autres données lui sont fournies, elle peut alors générer un emploi du temps beaucoup plus pertinent en prenant en compte des nouvelles contraintes telles que les disponibilités des enseignants ou des installations de pratique pour les APSA. De plus, il est possible de travailler directement sur l'emploi du temps généré afin de mettre à jour les créneaux, les salles et les enseignants par rapport à l'emploi du temps réel.

Ce résultat n'a pu être obtenu que grâce à la résolution des principaux problèmes rencontrés. Ceux-ci, qui concernaient les informations à apporter au logiciel, ont été résolus par la création de fichiers types et la récupération de certaines données qui resteront utilisables sans avoir besoin d'être revues entièrement chaque année.

Durant ce projet, j'ai eu l'opportunité d'apprendre à programmer dans plusieurs langages que je ne connaissais pas, ainsi qu'à utiliser le Framework Symfony qui m'était jusqu'alors inconnu. Cependant, ce que j'ai le plus appris concerne la culture d'entreprise et la gestion d'une entreprise, qui sont très dépendantes l'une de l'autre. En effet, mon projet étant au cœur de la gestion, j'ai pu voir différents comportements vis-à-vis de l'arrivée de ce logiciel et ainsi m'apercevoir de l'importance de la communication au sein d'une entreprise.

L'objectif principal de ce stage a été atteint ; néanmoins l'application résultante peut être encore améliorée avec l'intégration de fonctionnalités complémentaires. De plus, dans l'optique d'une utilisation plus générale que pour la génération des emplois du temps de l'UFR STAPS, il faudrait un peu adapter le programme ainsi que les informations à fournir.

Bibliographie

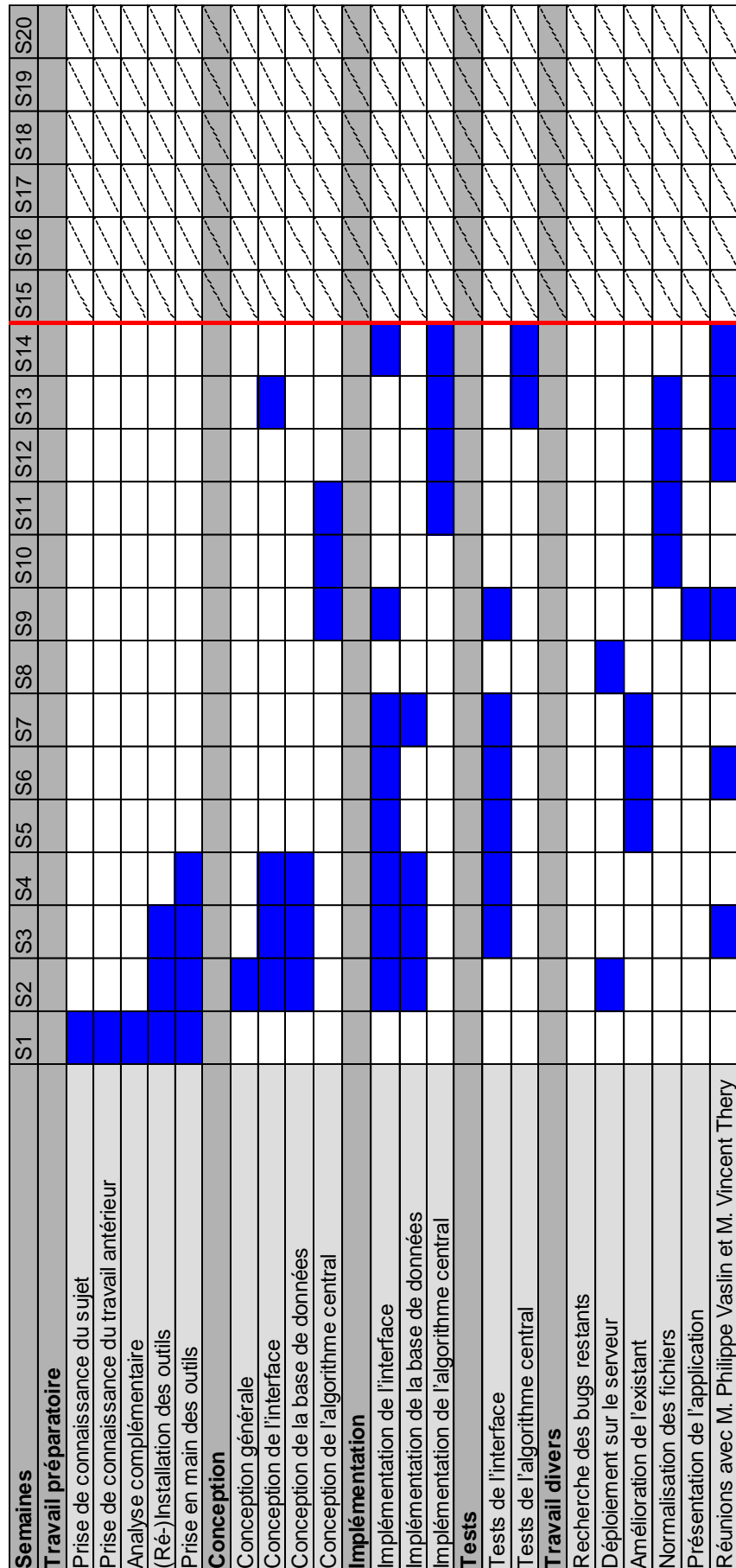
- [1] «Votre UFR | UFR STAPS - Sciences et techniques des activités physiques et sportives, Université Clermont Auvergne,» [En ligne]. Available: <http://staps.univ-bpclermont.fr/article127.html>. [Accès le 23 Mai 2017].
- [2] A. Pourcher-Portalier, «Emploi du Temps STAPS,» ISIMA, 2015.
- [3] F. Duché et B. Dusselier, «Développement d'une application web d'aide à la gestion de l'emploi du temps individualisé des étudiants de l'UFR STAPS,» ISIMA, 2016.
- [4] A. El Hachadi et Y. Kriouile, «Réalisation d'une application de génération des emplois du temps,» ISIMA, 2017.
- [5] «PHP - Définition,» [En ligne]. Available: <http://glossaire.infowebmaster.fr/php/>. [Accès le 23 Mai 2017].
- [6] «Javascript - Définition,» [En ligne]. Available: <http://glossaire.infowebmaster.fr/javascript/>. [Accès le 23 Mai 2017].
- [7] «CSS - Définition,» [En ligne]. Available: <http://glossaire.infowebmaster.fr/css/>. [Accès le 23 Mai 2017].
- [8] «HTML - Définition,» [En ligne]. Available: <http://glossaire.infowebmaster.fr/html/>. [Accès le 23 Mai 2017].
- [9] «jQuery - Définition,» [En ligne]. Available: <http://glossaire.infowebmaster.fr/jquery/>. [Accès le 23 Mai 2017].
- [10] «Bootstrap (framework) - Wikipédia,» [En ligne]. Available: [https://fr.wikipedia.org/wiki/Bootstrap_\(framework\)](https://fr.wikipedia.org/wiki/Bootstrap_(framework)). [Accès le 23 Mai 2017].
- [11] «Symfony - Wikipédia,» [En ligne]. Available: <https://fr.wikipedia.org/wiki/Symfony>. [Accès le 23 Mai 2017].
- [12] «Doctrine (ORM) - Wikipédia,» [En ligne]. Available: [https://fr.wikipedia.org/wiki/Doctrine_\(ORM\)](https://fr.wikipedia.org/wiki/Doctrine_(ORM)). [Accès le 23 Mai 2017].

- [13] «Dictionnaire des développeurs,» [En ligne]. Available: dico.developpez.com/html/3020-Conception-MVC-Model-View-Controller.php. [Accès le 06 Juin 2017].
- [14] «Stack Overflow - Where Developers Learn, Share & Build Their Careers - Stack Overflow,» [En ligne]. Available: <https://stackoverflow.com/>. [Accès le 7 Juillet 2017].
- [15] «PHP: Manuel PHP - Manual,» [En ligne]. Available: <http://php.net/manual/fr/>. [Accès le 7 Juillet 2017].
- [16] «Liste des forums,» [En ligne]. Available: <https://openclassrooms.com/forum/>. [Accès le 7 Juillet 2017].
- [17] «Forum du club des développeurs et IT Pro,» [En ligne]. Available: <https://www.developpez.net/forums/>. [Accès le 7 Juillet 2017].
- [18] «Documentation | FullCalendar,» [En ligne]. Available: <https://fullcalendar.io/docs/>. [Accès le 7 Juillet 2017].

Annexe 1 : Diagramme de Gantt prévisionnel

Semaines	S1	S2	S3	S4	S5	S6	S7	S8	S9	S10	S11	S12	S13	S14	S15	S16	S17	S18	S19	S20
Travail préparatoire																				
Prise de connaissance du sujet																				
Prise de connaissance du travail antérieur																				
Analyse complémentaire																				
Installation des outils																				
Prise en main des outils																				
Conception																				
Conception de l'interface																				
Conception de l'algorithme central																				
Implémentation																				
Implémentation de l'interface																				
Implémentation de l'algorithme central																				
Tests																				
Tests de l'interface																				
Tests de l'algorithme central																				
Travail divers																				
Mise en place de la base de données																				
Recherche des bugs et amélioration de l'existant																				

Annexe 2 : Diagramme de Gantt réel



||

[illegible]

Annexe 4 : Exemple du fichier des APSA

A1	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
	CIP	Composante	Date De Création	Nature	Semestre	Code Initial	Code (+1)	Libellé court initial (25 caractères?)	Statut ?	Type	Code _ (4)	Famille	Code _ (5)	APSA	Code _ (ex)	Intitulé abrégé (< 25 caractères)		
1																		
2	STAPS	STAPS	26/05/2017	Matière	1	Z116AM10	Z116AM100	A Athlétiques Théorie	O	Théorie	Theo	Activités Athlétiques	Athle	Course	COUR	Theo Athle	Z116AM100 Theo Athle	
3							Z116AM101				Theo	Athle	Athle			Theo Athle Cour	Z116AM101 Theo Athle COUR	
4							Z116AM102				Theo	Athle	Athle			Theo Athle Saut	Z116AM102 Theo Athle SAUT	
5							Z116AM103				Theo		Athle	Lancer	LANC	Theo Athle Lanc	Z116AM103 Theo Athle LANC	
6																		
7											Theo							
8	STAPS	STAPS	26/05/2017	Matière	1	Z116AM11	Z116AM110	A de natation Théorie	O	Théorie	Theo	Activités Aquatiques	Aqua	Natation	NAT	Theo Aqua	Z116AM110 Theo Aqua	
9							Z116AM111				Theo	Aqua	Aqua			Theo Aqua Nat	Z116AM111 Theo Aqua NAT	
10							Z116AM112				Theo	Aqua	Aqua	Sauvetage	SAUV	Theo Aqua Sauv	Z116AM112 Theo Aqua SAUV	
11							Z116AM113				Theo	Aqua	Aqua	Water-Polo	WP	Theo Aqua WP	Z116AM113 Theo Aqua WP	
12											Theo							
13											Theo							
14	STAPS	STAPS	26/05/2017	Matière	1	Z116AM12	Z116AM120	A sport de combat Théorie	O	Théorie	Theo	Sport combat	Comb	Judo	JUDO	Theo Comb	Z116AM120 Theo Comb	
15							Z116AM121				Theo		Comb			Theo Comb Judo	Z116AM121 Theo Comb JUDO	
16							Z116AM122				Theo	Comb	Karaté		KARA	Theo Comb Kara	Z116AM122 Theo Comb KARA	
17							Z116AM123				Theo	Comb	Boxe anglaise		BA	Theo Comb BA	Z116AM123 Theo Comb BA	
18							Z116AM124				Theo	Comb	Boxe française		BF	Theo Comb BF	Z116AM124 Theo Comb BF	
19											Theo							
20	STAPS	STAPS	26/05/2017	Matière	1	Z116AM14	Z116AM140	A de grd terrain Théorie	O	Théorie	Theo	Sports Co Gd Terrain	SCGT	Football	FOOT	Theo SCGT	Z116AM140 Theo SCGT	
21							Z116AM141				Theo		SCGT			Theo SCGT Foot	Z116AM141 Theo SCGT FOOT	
22							Z116AM142				Theo		SCGT	Rugby	RUG	Theo SCGT Rug	Z116AM142 Theo SCGT RUG	
23											Theo							
24	STAPS	STAPS	26/05/2017	Matière	1	Z116AM15	Z116AM150	A Petit terrain Théorie	O	Théorie	Theo	Sports Co Pt Terrain	SCPT	Basket-ball	BB	Theo SCPT BB	Z116AM150 Theo SCPT	
25							Z116AM151				Theo	SCPT	Handball		HB	Theo SCPT HB	Z116AM151 Theo SCPT HB	
26							Z116AM152				Theo	SCPT	Volley-ball		VB	Theo SCPT VB	Z116AM152 Theo SCPT VB	
27							Z116AM153				Theo		SCPT					
28											Theo							
29											Theo							
30	STAPS	STAPS	26/05/2017	Matière	1	Z116AM16	Z116AM160	APPN Théorie	O	Théorie	Theo	APPN	APPN	Canoe-kayak	CK	Theo APPN	Z116AM160 Theo APPN	
31							Z116AM161				Theo	APPN	APPN	Course Orientation	CO	Theo APPN CK	Z116AM161 Theo APPN CK	
32							Z116AM162				Theo	APPN	APPN	Escalade	ESC	Theo APPN CO	Z116AM162 Theo APPN CO	
33							Z116AM163				Theo	APPN	APPN	Ski alpin	SKA	Theo APPN ESC	Z116AM163 Theo APPN ESC	
34							Z116AM164				Theo	APPN	APPN	Ski de fond	SKF	Theo APPN SKA	Z116AM164 Theo APPN SKA	
35							Z116AM165				Theo	APPN	APPN			Theo APPN SKF	Z116AM165 Theo APPN SKF	
36											Theo							
37	STAPS	STAPS	26/05/2017	Matière	1	Z116AM20	Z116AM200	A Athlétiques Pratique	O	Pratique	Prat	Activités Athlétiques	Athle	Course	COUR	Prat Athle	Z116AM200 Prat Athle	
38							Z116AM201				Prat	Athle	Athle		SAUT	Prat Athle Saut	Z116AM201 Prat Athle SAUT	
39							Z116AM202				Prat	Athle	Athle	Lancer	LANC	Prat Athle Lanc	Z116AM202 Prat Athle SAUT	
40							Z116AM203				Prat							
41											Prat							
42											Prat							

v

A1	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
	Localité	Code 1	Nom installation	Code 2	Zone d'activité	Code 3	Code final	Capacité	Jour souhaité	Horaires	Début période	Fin période		Statut résa (AF = à faire, DD = demandé, OK = obtenu)				
1	Ceyrat	CEY	Arénium	ARTENIUM	Salle VIP 1	VIP1	CEY ARTENIUM VIP1	24					Colonne 1	AF				
2	Ceyrat	CEY	Arénium	ARTENIUM	Salle VIP 2	VIP2	CEY ARTENIUM VIP2	24						AF				
3	Ceyrat	CEY	Arénium	ARTENIUM	Surface Judo	JUDO	CEY ARTENIUM JUDO	24						AF				
4	Ceyrat	CEY	Arénium	ARTENIUM	Surface Karaté	KARATE	CEY ARTENIUM KARATE	24						AF				
5	Ceyrat	CEY	Arénium	ARTENIUM	Salle musculation	MUSCU	CEY ARTENIUM MUSCU	24						AF				
6	Ceyrat	CEY	Arénium	ARTENIUM														
7	Cézeaux	CEZ	Gymnase	GYMNASE	Gymnase	GYMN	CEZ GYMNASE GYMN	24 mardi		8H00 à 10H00	18/09/17	15/12/17		DD				
8	Cézeaux	CEZ	Gymnase	GYMNASE	Gymnase	GYMN	CEZ GYMNASE GYMN	24 mardi		10H00 à 12H00	22/01/18	15/12/17		DD				
9	Cézeaux	CEZ	Gymnase	GYMNASE	Gymnase	GYMN	CEZ GYMNASE GYMN	24 mercredi		8H00 à 10H00	18/09/17	15/12/17		DD				
10	Cézeaux	CEZ	Gymnase	GYMNASE	Gymnase	GYMN	CEZ GYMNASE GYMN	24 mercredi		10H00 à 12H00	18/09/17	15/12/17		DD				
11	Cézeaux	CEZ	Gymnase	GYMNASE	Gymnase	GYMN	CEZ GYMNASE GYMN	24 mercredi		8H00 à 10H00	18/09/17	15/12/17		DD				
12	Cézeaux	CEZ	Gymnase	GYMNASE	Gymnase	GYMN	CEZ GYMNASE GYMN	24 vendredi		10H00 à 12H00	18/09/17	15/12/17		DD				
13	Cézeaux	CEZ	Gymnase	GYMNASE	Gymnase	GYMN	CEZ GYMNASE GYMN	24 vendredi		10H00 à 12H00	18/09/17	15/12/17		DD				
14	Cézeaux	CEZ	Gymnase	GYMNASE	Gymnase	GYMN	CEZ GYMNASE GYMN	24 mardi		8H00 à 10H00	22/01/18	28/04/18		DD				
15	Cézeaux	CEZ	Gymnase	GYMNASE	Gymnase	GYMN	CEZ GYMNASE GYMN	24 mardi		10H00 à 12H00	22/01/18	28/04/18		DD				
16	Cézeaux	CEZ	Gymnase	GYMNASE	Gymnase	GYMN	CEZ GYMNASE GYMN	24 mercredi		8H00 à 10H00	22/01/18	28/04/18		DD				
17	Cézeaux	CEZ	Gymnase	GYMNASE	Gymnase	GYMN	CEZ GYMNASE GYMN	24 mercredi		10H00 à 12H00	22/01/18	28/04/18		DD				
18	Cézeaux	CEZ	Gymnase	GYMNASE	Gymnase	GYMN	CEZ GYMNASE GYMN	24 mercredi		8H00 à 10H00	22/01/18	28/04/18		DD				
19	Cézeaux	CEZ	Gymnase	GYMNASE	Gymnase	GYMN	CEZ GYMNASE GYMN	24 mercredi		10H00 à 12H00	22/01/18	28/04/18		DD				
20	Cézeaux	CEZ	Gymnase	GYMNASE	Gymnase	GYMN	CEZ GYMNASE GYMN	24 vendredi		10H00 à 12H00	22/01/18	28/04/18		DD				
21	Cézeaux	CEZ	Gymnase	GYMNASE	Gymnase	GYMN	CEZ GYMNASE GYMN	24						DD				
22	Cézeaux	CEZ	Gymnase	GYMNASE	Salle d'aggrès	AGRES	CEZ GYMNASE AGRES	24						AF				
23	Cézeaux	CEZ	Gymnase	GYMNASE	Salle gym au sol	GYMSOL	CEZ GYMNASE GYMSOL	24						AF				
24	Cézeaux	CEZ	Gymnase	GYMNASE	Salle de danse	DANSE	CEZ GYMNASE DANSE	24						AF				
25	Cézeaux	CEZ	Terrains	TERRAINS	Piste d'athlétisme	ATHLE	CEZ TERRAINS ATHLE	24 lundi		8H00 à 10H00	18/09/17	24/09/17		DD				
26	Cézeaux	CEZ	Terrains	TERRAINS	Piste d'athlétisme	ATHLE	CEZ TERRAINS ATHLE	24 lundi		10H00 à 12H00	18/09/17	24/09/17		DD				
27	Cézeaux	CEZ	Terrains	TERRAINS	Piste d'athlétisme	ATHLE	CEZ TERRAINS ATHLE	24 mardi		8H00 à 10H00	18/09/17	24/09/17		DD				</

Annexe 6 : Exemple du fichier des correspondances APSA-Installations

INSTALLATIONS SPORTIVES									
A	B	C	D	E	F	G	H	I	J
1	CEY_ARTENIUM_VIP1	CEY_ARTENIUM_VIP2	CEY_ARTENIUM_JUDO	CEY_ARTENIUM_KARATE	CEY_ARTENIUM_MUSCU	AUB_PLADNER_GYMN	AUB_PLADNER_SALLE 1		
2									
3									
4	Prat_Athle								
5	Prat_Athle_Cour								
6	Prat_Athle_Saut								
7	Prat_Athle_Lanc								
8									
9	Prat_Aqua								
10	Prat_Aqua_Nat								
11	Prat_Aqua_Sauv								
12	Prat_Aqua_WP								
13									
14	Prat_Comb								
15	Prat_Comb_Judo		X						
16	Prat_Comb_Kara			X					
17	Prat_Comb_BA								
18	Prat_Comb_BF								
19									
20	Prat_SCGT								
21	Prat_SCGT_Foot								
22	Prat_SCGT_Rug								
23									
24	Prat_SCPT								
25	Prat_SCPT_BB								
26	Prat_SCPT_HB					X			
27	Prat_SCPT_VB					X			
28									
29	Prat_APPN								
30	Prat_APPN_OK								
31	Prat_APPN_CO								
32	Prat_APPN_ESC								
33	Prat_APPN_SKA								
34	Prat_APPN_SKF								
35									
36									
37									
38									
39									
40									
41									