

Rapport du Projet Gestion Étudiants

WebSite: Liste des Etudiants

Rechercher ...

Rechercher

Affichage initial

Liste des Etudiants

Choisir le fichier ☐ liste.json

Affiche-moi le contenu du fichier

Num	Nom	Prénom	Clas	INFO2	S4	Ue41	Ue42	S3	Ue31	Ue32	Ue33	INFO1	S2	Ue22	Ue21	S1	Ue11	Ue12	detail
					18	10	8	28	10	12	6		30	14	16	30	17	13	
					12.19	12.80	11.58	12.49	12.12	10.95	14.41		10.36	10.91	9.81	11.49	9.99	12.98	
2017_1	bisson	julien*			11.76	11.54	11.97	12.84	12.14	11.43	14.93		10.36	10.91	9.81	13.09	11.71	14.46	+
2017_2	dachy	remy			12.33	13.22	11.45	12.83	12.12	11.43	14.93		10.36	10.91	9.81	10.95	9.42	12.49	+
2017_14	bisson	julien			11.76	11.54	11.97	10.13	12.12	7.54	10.73		10.36	10.91	9.81	13.09	11.71	14.46	+
2017_3	dachy	remy			12.33	13.22	11.45	12.83	12.12	11.43	14.93		10.36	10.91	9.81	10.95	9.42	12.49	+
2017_4	dachy	remy			12.33	13.22	11.45	12.83	12.12	11.43	14.93		10.36	10.91	9.81	10.95	9.42	12.49	+

TUTEUR DU PROJET : HSU TIENTE
Groupe

Florian Flahaut
Bastien Cornru

Stephan Loison
Alexis Delarre

Table des matières

I] Introduction	1
1) La gestion de notes des étudiants de l'IUT de Lens	1
2) Principe	1
II] Équipe	2
1) Composition	2
2) Répartition des tâches	2
III] Le projet	5
1) Objectif	5
2) Contexte	5
3) Planning	5
IV] L'analyse de l'existant et la mise en place des idées	6
1) La ligne de vie du projet existant	6
2) L'amélioration de l'existant par des outils	7
3) Tout refaire ou repiocher dans le code existant ?	7
4) Les outils	8
V] Synthèse	9
1) Objectifs	9
2) Arborescence	10
3) Initialisation du projet	11
4) Automatisation	12
5) Difficultés rencontrées	13
6) Bilan de chacun	14
VI] Technique	16
1) Introduction	16
2) Les vérifications	16
3) Le code	17



I] Introduction

1) La gestion de notes des étudiants de l'IUT de Lens

En cette première année de **la licence professionnelle DIOC**, et surtout, en ce premier projet tuteuré du semestre cinq, nous avons été choisis pour **améliorer** et **développer** le projet de gestion de notes des étudiants **préalablement créé par des DUT2 de l'année 2016-2017**.

Nous avons donc repris un projet développé, et nous **avons installé un système de développement collaboratif**.

2) Principe

Le fonctionnement complet de ce projet repose sur deux grandes étapes pouvant être divisées en plusieurs petites étapes.

Dans un premier temps nous avons 2 fichiers CSV comportant des informations qui, par la suite, vont nous permettre de générer des fichiers Excel.

Ces fichiers Excel vont être distribués à chaque professeur en fonction de leur matière et de leurs élèves en début de semestre.

En fin de semestre, le professeur en charge d'utiliser le projet récupère chaque fichier Excel et applique ainsi la seconde partie.

La seconde partie consiste à créer, à partir des fichiers Excel récupérés (comportant les moyennes de chaque élève dans la matière du professeur), un fichier JSON (comportant les informations de chaque élève leurs moyennes/absences/validations des semestres etc) que l'on va introduire dans une page web et qui va donc nous sortir en affichage les notes des semestres de l'élève sélectionné ainsi qu'un suivi de cet élève.

II] Équipe

1) Composition

Équipe :

Florian Flahaut

E-mail: florian-flahaut@hotmail.fr

Tel: 06-04-53-19-47

Bastien Cornu

E-mail: bast-du-62110@hotmail.fr

Tel: 07-70-43-08-68

Stephan Loison

E-mail:

Tel: 06-46-02-74-98

Alexis Delarre

E-mail: delarre.alexis@gmail.com

Tel: 06-49-49-09-60

2) Répartition des tâches

Bastien :

- Analyse de l'architecture du projet ainsi que des différents programmes.
- Création d'un schéma de l'architecture du projet.
- Création d'une branche nommée "testing" afin d'y mettre mon travail.
- Création d'un programme qui va permettre de convertir des fichiers XLSX en CSV.
- Création d'un programme nommé "initialiser.php". Ce programme va permettre de prendre les informations dans les fichiers d'étudiants et matières au format CSV de manière à créer tous les fichiers excel dont nous avons besoin. Le semestre voulu devra être passé en paramètre. Les vérifications sur le paramètre sont implémentées ainsi que la vérification des fichiers existant pour le semestre demandé.
- Création d'un programme en bash qui va permettre de créer toute l'architecture nécessaire pour le bon fonctionnement du programme. Un paramètre devra être passé qui sera l'année. ex(16_17).



Stephan :

Alexis :

- Analyse du code déjà présent et, après réflexion, j'ai préféré repartir de zéro.
- Mon but étant de convertir des données de chaque étudiant contenues dans les feuilles du fichier Bilan, pour en ressortir une liste en JSON. Après que le fichier Bilan soit rempli par les notes des professeurs, je prends le fichier bilan pour en ressortir une liste de tous les étudiants avec leurs notes et leurs informations personnelles. (Pour arriver à convertir les données des feuilles du fichier bilan.xlsx, j'utilise la classe PHPExcel)
- A ce stade : la conversion des données se trouvant dans le fichier bilan.xlsx en une liste en JSON est faite. Dans la liste se trouve l'information pour l'avatar qui n'est pas encore terminé, il manque à mettre tout en minuscule pour simplifier la gestion des noms d'avatars.

Florian :

- Analyse du code déjà présent.
- Création d'une branch AutomatisationExcel : mon but étant d'automatiser le travail de la secrétaire en fin de semestre (après avoir reçu tous les fichiers Excel complétés par les professeurs avec les moyennes de chaque élève). À partir de ces fichiers : SE, RX, Algo etc et du fichier bilan généré en début de semestre, je dois intégrer chaque fichier Excel de chaque matière dans le fichier Excel Bilan et compléter les données dans certaines feuilles du fichier Excel Bilan afin de, par la suite, convertir le fichier excel Bilan en un fichier json prêt à l'emploi.
- Création d'une branch permettant de fixer un bug : lors de l'ouverture du fichier Excel Bilan les données ne se mettant pas à jour, il faut modifier l'ordre des feuilles du fichier Excel Bilan pour actualiser.



- Création d'une branch composer permettant d'avoir un minimum de bases pour le rendu final du projet, le fichier composer.json contient les dépendances vers PHPExcel qui nous permettent de gérer des fichiers Excel avec du PHP grâce à des fonctions déjà implémentées.
- Création des README.md des branch : master, composer, testing, AutomatisationExcel
- Rapport



III] Le projet

1) Objectif

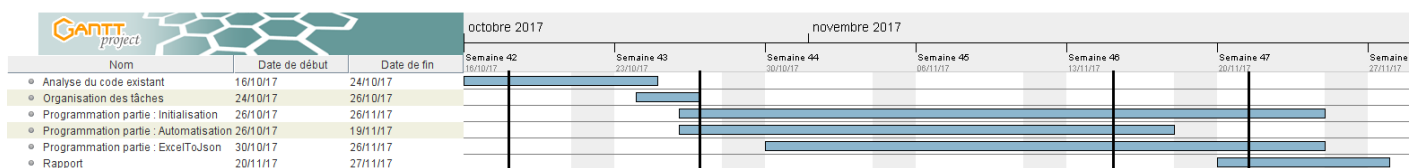
L'objectif de ce projet est de rendre l'application existante mieux élaborée et de permettre aux prochains développeurs une **facilité de compréhension** et de développement (avec l'ajout de commentaires dans le code, des fichiers README pour l'exécution, une adaptation du code) rendant ainsi le projet plus professionnel.

2) Contexte

Ce projet est encadré par les professeurs **T.Hsu** incluant les matières de Programmation **PHP/Javascript** de la licence DIOC de l'IUT de Lens. Ce projet a pu voir le jour grâce aux connaissances acquises pendant les TD/TP de **PHP**, au cours de **GIT** et à celles acquises en autodidacte (utilisation de **PHPExcel**).

Le travail demandé a été réalisé par quadrinôme : **Florian Flahaut, Bastien Cornu, Stephan Loison, Alexis Delarre.**

3) Planning



Pour ce projet, nous avons disposé **d'un peu plus de 1 mois** pour reprendre le code existant, **l'analyser**, le **comprendre** et **l'améliorer**. Nous n'avons pas forcément repris le code existant dans son intégralité mais nous sommes inspirés de celui-ci dans le but d'avoir l'algorithmique plus aisément.

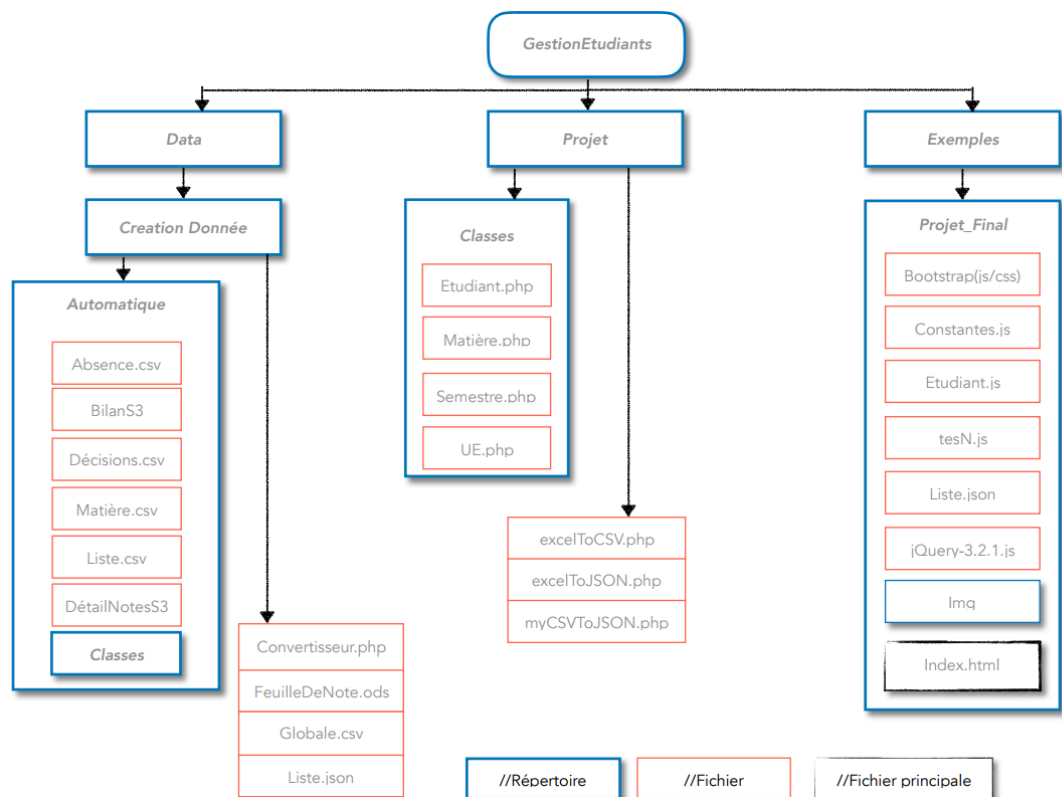


IV] L'analyse de l'existant et la mise en place des idées

1) La ligne de vie du projet existant

Au début de notre projet, nous devons explorer la totalité des fichiers existant pour observer le fonctionnement de celui-ci. Le projet n'était pas optimisé pour une compréhension facile alors la création d'un fichier PDF contenant les répertoires et les fichiers importants était non négligeable.

Voici donc, la vision globale du projet à son début. Cette architecture nous a permis de comprendre qu'il y avait plusieurs parties à ce projet.



Après cela, il fallait donc explorer chaque fichier php, essayer de les commenter afin de rendre le code plus lisible et plus clair. Cette phase d'analyse



du code existant nous a pris 1 semaine, comprenant ainsi, dans cette semaine, une réunion qui nous a permis de véritablement savoir quoi faire, comment le faire et le réaliser de manière propre.

2) L'amélioration de l'existant par des outils

À partir de ce moment nous nous sommes attribués des tâches bien définies. **Chaque phase étant bien séparée**, nous pouvions travailler de **manière autonome** sur chacune d'elles sans avoir besoin que la personne s'occupant de la phase d'avant ne doive fournir un code pour que nous puissions continuer notre code. Nous avons donc commencé par se créer, pour chaque phase, une branch sur la forge de l'université de l'Artois. Nous avons donc séparé les phases en 3 branch :

_ **La branch testing** est la branch de l'initialisation : conversion des fichiers CSV en Excel.

_ **La branch AutomatisationExcel** est la branch d'automatisation du travail de la secrétaire.

_ **La branch ExcelToJson** est la branch de conversion des fichiers Excel en JSON

Chaque branch peut ainsi travailler **indépendamment** des autres, sans possibilité de **'casser'** le code des autres développeurs.

3) Tout refaire ou repiocher dans le code existant ?

L'une des questions importantes était celle de la reprise du code précédent. Après avoir commenté une bonne partie du code, nous nous sommes demandé si nous devions ou non reprendre des morceaux de codes de l'ancien projet. Après quelque temps de réflexion, nous voulions repartir sur des bases structurées et nous avons donc repris la ligne conductrice du projet tout en

éloignant le code existant, soit : la lecture des fichiers CSV des étudiants, la lecture des fichiers CSV des matières, l'ouverture de répertoire afin d'enregistrer les données.

4) Les outils

Lors de ce projet, l'utilisation de certains outils étaient indispensable : l'utilisation de PHPStorm ainsi que l'utilisation de GIT. Nous avons donc utilisé les connaissances vues en cours pour rendre le projet le plus propre possible, par exemple nous avons créé différentes branches pour travailler chacun de notre côté, pouvoir récupérer le code peut-importe sur quelle machine à partir du moment que ou nous avons nos identifiants.



V] Synthèse

1) Objectifs

L'objectif ici, est de décrire le fonctionnement de manière générale, nous allons donc suivre le déroulement du projet.

Avant tout, nous devons exécuter **deux étapes** dans ce projet : l'une en **début de semestre et l'autre en fin de semestre**. La première va permettre de générer des documents pour les professeurs, la seconde va regrouper ces documents et les afficher pour parcourir ces données plus facilement.

Lors des premières réunions nous nous sommes mis d'accord sur une architecture du projet final, voici une représentation de cette architecture complétée avec les fichiers :

```
/ ----->
  /16_17->
    |
    | /Admin->
    | |
    | | ListeInfo*_S*_1617.csv
    | | Liste_Matière_Info*_S*_1617.csv
    | | automatiser.php
    | | initialiser.php
    | |
    | | /S1
    | | /S2
    | | /S3----->
    | | |
    | | | /excel->
    | | | |
    | | | | APA_Info*_S*_1617.xlsx
    | | | | Absences_Info*_S*_1617.xlsx
    | | | | ...
    | | |
    | | | /csv
    | | | /json
    | |
    | | /S4
    |
    | ArchitectureProjet.pdf
    | fileTreeCreation.sh
    | README.txt
```

Cette structure permet de bien séparer les différentes étapes de cette phase 1.

Ensuite, grâce à cette arborescence nous allons pouvoir placer les fichiers importants dans le répertoire /Admin : les fichiers CSV, les fichiers php.



2) Arborescence

Dans un premier temps nous allons parler de l'arborescence, puis ensuite, de la phase d'initialisation.

Nous avons le répertoire 16_17 ainsi que les répertoires Admin, S1, S2, S3, S4 qui se créent grâce au fichier fileTreeCreation.sh dans la racine du projet. Ce fichier .sh permet la création de l'arborescence selon les paramètres entrés en argument.

Ici la commande : `sh fileTreeCreation.sh 16_17` permet de générer l'arborescence ainsi :

```
/16_17->
|
|      /Admin
|      /S1---->
|      |      /csv
|      |      /excel
|      |      /json
|      /S2
|      /S3
|      /S4
```

Grâce à ce fichier .sh nous pouvons donc générer à chaque début d'année un répertoire qui regroupera chaque semestre de l'année en cours avec, dans chaque répertoire de chaque semestre, les répertoires /csv, /excel, /json qui par la suite vont posséder des fichiers générés grâce aux différentes phases.

Ensuite, il faut remplir ces répertoires de manière correcte, dans le répertoire /Admin, les fichiers CSV contenant les étudiants et les matières en rapport avec ses étudiants devront apparaitre. Les répertoires /S1, /S2, /S3, /S4 et leurs sous-répertoires devront rester vides jusqu'à l'initialisation.

Lorsque les fichiers CSV et le fichier initialiser.php se trouvent dans le répertoire /Admin nous pouvons alors exécuter la commande :

```
php initialiser.php S3
```

qui va créer dans le répertoire /16_17/S3/excel/ un ensemble de fichiers Excel associant les étudiants à leurs matières.

3) Initialisation du projet

Nous arrivons ainsi à la partie du fichier initialiser.php. Cette partie étant la plus longue, nous allons donc passer en revue les exécutions de ce fichier php en ayant une vue générale.

Dans un premier temps les fonctions readCSVEtudiant et readCSVMatieres vont s'exécuter. Ces dernières permettant de stocker dans des tableaux les données contenues dans les fichiers précédemment placés dans le répertoire /16_17/Admin/. Nous allons donc stocker dans différents tableaux globaux : les étudiants, les matières, les UE et les départements. Ce stockage dans des tableaux globaux permet la récupération facile d'informations à travers toutes les fonctions appelées.

Ensuite la fonction openDirectories va essayer d'ouvrir les répertoires afin de pouvoir y stocker les fichiers Excel par la suite. Dans cette fonction, si le répertoire est trouvé, l'appel d'une autre fonction va permettre d'écrire dans chaque fichier Excel de chaque matière, les informations des étudiants, et les informations de la matière. Pendant cette phase, un fichier Excel Bilan est créé et est rempli de fonction excel permettant d'aller chercher des données dans d'autres feuilles Excel de ce même fichier. Le problème étant que nous stockons chaque matière dans un fichier Excel séparé des autres. Donc la récupération est impossible à partir de ce moment (la phase d'automatisation est donc la suite logique de cette phase d'initialisation).



1	Numéro	Nom	Prénom	Groupe	Moyenne	Rang
2						
3	20150926	BAUDI	ieoffr	-B		
4	20150927	BIGAR	amue	-C		
44	20150967	MERLI	ntho	-A		
45	20150968	PETIT	hloe	-C		
46	20150969	POLA	ieoffr	-A		
47	20150970	POUIL	iaetar	-B		
48	20150971	ROLA	erem	-C		
49	20150972	ROTTI	heo	-B		
50	20150973	SACRE	ierre	-B		
51	20150974	SCHILI	ylan	-C		
52	20150975	SURET	évin	-A		
53	20150976	SZKAR	hibau	-B		
54	20150977	SZYML	lexar	-C		
55	20150978	TANG	rédi	-C		
56	20150979	TASSA	lexar	-B		
57	20150980	THERA	ucas	-A		
58	20150981	THOM	ouis	-B		
59	20150982	VAND	ymet	-B		
60	20150983	VANR	ulien	-A		
61	20150984	VELLA	manu	-C		
62	20150985	VERPC	ylvair	-A		
63						
64	Département	INFO				
65	Année	INFO2				
66	Date début	01/09/2016				
67	Date de fin	31/06/2017				
68	Semestre	S3				
69	Code	M3103				
70	Intitulé	Algorithmique avancée				
71	Coefficient	1,5				
72	Responsable					
73	Moyenne	#NOM?				
74	Max	0				
75	Min	0				
76						
77						
78						

Nous obtenons donc ceci pour chaque matière :

Ici nous avons donc chaque étudiant, avec leurs numéros d'étudiant, leur nom, prénom, groupe ainsi que beaucoup d'informations concernant la matière.

Par la suite chaque fichier Excel est distribué au professeur en charge de la matière correspondante au fichier, ils peuvent donc remplir : la moyenne de chaque étudiant en fin de semestre.

4) Automatisation

Cette partie s'exécute en fin de semestre, elle consiste à regrouper l'ensemble des fichiers Excel –remplis-, donnés par les professeurs en fin de semestre. Avant, cette partie était faite à la main, l'automatisation est donc inévitable car ce travail est rébarbatif.

Concrètement nous allons obtenir une liste de fichiers Excel comportant les moyennes de chaque élève dans chaque matière dans le répertoire /excel de



chaque semestre. Après exécution du programme, nous allons obtenir, un seul fichier Excel comportant la totalité des matières pour ce semestre.

Nous allons donc obtenir un fichier Excel comme ceci :

	A	B	C	D	E	F	G	H	I	J	K	L	X	Y	Z
1	Numéro	Nom	Prénom	Groupe	M S3	Rang	UE31 INFO	UE32 CCG	UE33 PT	M3101 SE3	M3102 RX	M3103 AP	A3303 PPP	sport :nc	Nom
2					30,00		12,00	12,00	6,00	2,50	1,50	1,50	1,00		
3	20150926	BAUDELET	Geoffrey	2-B	11,58	9	11,19	10,88	13,77	12,00	13,00	9,00	6,60		BAUDELET
60	20150983	VANROY	Julien	2-A	10,46	27	11,41	9,22	11,05	14,00	15,00	7,10	6,00		VANROY
61	20150984	VELLA	Emanuele	2-C	11,11	13	10,81	11,76	10,38	16,00	17,00	4,40	9,50		VELLA
62	20150985	VERPOEST	Sylvain	2-A	13,96	1	14,99	13,09	13,67	18,00	19,00	17,80	11,80		VERPOEST
63	20150986	WICART	Melvin	2-B	11,14	12	9,84	12,79	10,45	17,00	15,00	10,40	12,00		WICART
64	xxxxxxxxx		Moyenne		10,29		7,60	11,76	12,67	15,40	15,80	10,84	12,02		
65			Max		13,96		14,99	16,39	18,92	18,00	19,00	17,90	20,00		
66			Min		7,77		2,75	7,19	6,22	14,00	15,00	3,40	5,60		
67	Département	INFO													
68	Année	INFO2													
69	Date début	01/09/2016													
70	Date de fin	31/06/2017													
71	Semestre	S3													
72	Moyenne	10,28591592													
73	Max	13,96333333													
74	Min	7,77													
75															
76															
77															
78															
79															
80															
81															
82															
83															
84															
85															
86															
87															
88															
89															
90															
91															
92															
93															
94															
95															
96															
97															
98															
99															
100															

Ici, certaines colonnes et certaines lignes ont été masquées de manière à englober généralement la vue ainsi que les matières en dans les feuilles du tableau Excel.

5) Difficultés rencontrées

Tout d'abord nous avons développé en PHP, plus précisément avec PHPEXcel qui est une librairie permettant de coder en utiliser des fichiers Excel. Il a donc fallu s'habituer aux nouvelles méthodes de lecture de fichier



Excel, lecture de données, écriture de fichier Excel, écriture de données dans les fichiers, écriture dans les feuilles etc. Il nous a ainsi fallu quelques heures afin de s'habituer, et surtout, de tester les limites de cette librairie.

En ce qui concerne le code existant, et l'architecture existante du projet, cela n'a pas été facile de se l'approprier, les morceaux de codes n'étaient que peu commentés ou pas du tout, la séparation des étapes n'était pas marquée ne rendant ainsi pas le code générique dans un certain sens.

Un autre problème survenu assez rapidement, mais qui n'est toujours pas réglé, est un problème qui se situe sur les formules qui se trouvent dans les cellules Excel : ces dernières ne veulent pas aller chercher les données se trouvant dans une autre feuille Excel dans le même fichier créant ainsi des problèmes : #REF ou des erreurs 509.

En ce qui concerne les tests unitaires, ils ne sont pas faits car nous avons vu précisément comment les faire il y a peu.

6) Bilan de chacun

Florian : Pour ma part, ce projet m'a permis d'être plus à l'aise avec le PHP, je trouve que c'est un langage agréable à programmer. La nouveauté ici, est GIT : un système de versionning qui nous a permis de travailler ensemble sur un même projet en même temps sans pour autant déranger le code des autres.

Je suis assez content d'avoir exploré une librairie PHP qui n'est pas inutile, l'utilisation de fichiers Excel dans le monde du travail est vraiment énorme et le fait de pouvoir mélanger Excel et PHP c'est une agréable surprise.

Dans l'ensemble, je pense que quelques semaines de plus auraient permis de finaliser la refonte de l'existant.

Bastien : Pour ce qui est de ma part, je trouve que ce projet a été bénéfique. Cela m'a permis de découvrir certaines choses comme la classe PHPExcel pour écrire dans les fichiers Excel et de travailler sur un code existant qui n'est pas une tâche simple ma foi. Ce projet m'a permis aussi de collaborer avec mes collègues, plus particulièrement Florian, en utilisant un outil de version comme git qui était très adapté à notre projet. Je trouve que ce projet m'a fait accroître mes compétences en PHP.

Alexis : Pour la partie de conversion Excel en JSON, un code existait déjà fait par les anciens élèves. Après avoir analysé le code, j'ai préféré repartir de zéro. Dans le code déjà fourni la personne convertissait d'abord le fichier Excel en CSV puis de CSV en JSON. Pour ma part en utilisant la classe PHPExcel je pouvais convertir directement le fichier Excel en JSON. J'ai donc décidé de ne pas passer par le CSV.

J'ai attendu d'avoir le fichier Bilan fait pour commencer à coder, pendant que le fichier Bilan était en construction j'ai testé la classe PHPExcel pour me l'approprier. Une fois le fichier bilan.xlsx fait j'ai donc commencé vraiment ma partie. J'ai tout d'abord fait en sorte de ressortir en JSON les informations d'un seul étudiant avoir le schéma de construction de la liste. Quand j'avais dans mon fichier JSON un étudiant j'ai étendu mon code pour récupérer les informations de tous les étudiants.

Le seul bémol pour le moment est pour la partie de l'avatar que je n'ai toujours pas réussi à résoudre, je voudrais que l'avatar soit « nom-prénom.png » tout en minuscule pour faciliter la gestion des avatars.

Stephan :

VI] Technique

1) Introduction

Dans cette partie technique nous allons aborder les différents fichiers utiles au bon fonctionnement du projet. Tout d'abord nous allons regarder de plus près le fichier `initialiser.php` qui contient la première phase du projet suivis du fichier `automatisation.php` permettant d'entamer la seconde phase du projet.

Tout d'abord, la lecture de ce passage (ref : 2 Initialisation du projet) est primordiale.

2) Les vérifications

Comme vous pouvez le voir, des vérifications se font de manière simple en interrogeant le nombre d'arguments en paramètre, ainsi que la syntaxe de ceux-ci.

Nous allons donc commencer par les vérifications :

```

16 //Si il n'y a pas de paramètre ou plus de 1, on arrête le programme et on déclenche une erreur.
17 if ($argc != 2) {
18     echo "Erreur paramètre : passer le semestre choisi " . PHP_EOL;
19     exit(1);
20 }
21
22 //Vérifie si le paramètre est un semestre sinon il déclenche une erreur.
23 function checkParam($semestre)
24 {
25     if (strcmp($semestre, "S1") == 0 || strcmp($semestre, "S2") == 0 || strcmp($semestre, "S3") == 0 || strcmp($semestre, "S4") == 0) {
26         return;
27     }
28     echo "Erreur : Le paramètre n'est pas un semestre (S1,S2,S3,S4)" . PHP_EOL;
29     exit(1);
30 }
31 
```

Exemple de code du fichier `initialiser.php`



Ici, une simple verification permet de verifier si l'architecture est adequate afin d'automatiser la suite.

```

183 //Ouverture du dossier contenant Les fichiers Excel récupéré avec Les moyennes de chaque eleves
184 if ($excel = opendir("../S3/excel")) {

228 } else {
229     probleme("Fichier Excel se trouvant dans le dossier Excel");
230 }

```

Exemple de code du fichier initialiser.php

Les vérifications sont faciles à comprendre : nous vérifions si la saisie de l'utilisateur est correcte afin de partir sur de bonnes bases.

3) Le code

Le fichier initialiser.php n'appelle que 3 fonctions, ces dernières appellent elles-mêmes des fonctions traitant les différents cas.

Dans un premier temps, nous allons lire les fichiers CSV des étudiants et des matières. Les fichiers CSV sont agencés de cette manière :

Etudiants : Numéro;Nom;Prénom;Groupe;Date;Bac;Lycée/établissement

Matières : Référence;Nom Module;Abréviation;Coefficient;UE;Semestre

En ayant connaissance de cela, nous allons stocker ces données dans des tableaux multidimensionnels de manière ordonnée :

```

6 //tableau pour stocker Les données des fichiers CSV
7 $infoEtu = [];
8 $info_person = [];
9 $infoMat = [];
10 $info_matiere = [];
11 $ue = [];
12 $departement = [];
13 $semestre = $argv[1];
14

```



Ensuite nous allons chercher le répertoire à ouvrir qui va contenir les fichiers Excel de chaque matière, nous allons donc créer des fichiers Excel pour chaque abréviation qu'il y a dans le tableau info_matiere. Donc, concrètement, nous créons chaque fichier Excel de chaque matière de cette manière :

```
195 //Fonction qui va boucler en appelant la fonction qui créer les fichiers exce
196 function browseArrayMatiere($chemin)
197 {
198     global $info_matiere;
199     for ($compteur = 1; $compteur < count($info_matiere); $compteur++) {
200         creationFilesXLS($chemin . $info_matiere[$compteur]["Abréviation"]);
201     }
202     creationFilesXLS($chemin . "Bilan");
203     creationFilesXLS($chemin . "Absences");
204     creationFilesXLS($chemin . "Jury");
205 }
206
```

Nous créons à part – dans le sens où nous définissons les noms des feuilles en dur – les fichiers Bilan, Absences et Jury car ils vont être différents des fichiers Excel des matières.

Comme vous pouvez le voir nous allons appeler la fonction creationFilesXLS qui prend en paramètre le chemin du fichier où il va être stocké. Par exemple dans le répertoire : 16_17/S2/excel/

Comme dit précédemment, les fichiers de chaque matière sont similaires, donc une fonction permet de les gérer car grâce au tableau précédemment construit, les données n'ont qu'à être piochées dans les tableaux et ensuite placées dans la bonne cellule. Nous faisons ça en deux étapes : la première remplit les informations personnelles de chaque étudiant dans le fichier, et ensuite en fonction du nom de la feuille de calcul : matière ou absence/jury/bilan les fichiers subissent un traitement différent.



```
$classeur = new PHPExcel;
$classeur->getProperties()->setCreator("Bastien Cornu");
$classeur->setActiveSheetIndex(0);
$feuille = $classeur->getActiveSheet();

// ajout des données dans la feuille de calcul
$feuille->setTitle($abreviation[count($abreviation) - 1]);
writeInfoPersonFilesXLS($feuille);
```

Ce morceau de code est utilisé à chaque fois qu'un fichier Excel est créé : pour la création d'un fichier Excel nous avons besoin d'appeler le constructeur PHPExcel. Ensuite, il faut au minimum set un attribut qui est setCreator. Puis, il faut se positionner sur la première feuille du fichier Excel, définir son nom – ici le nom correspond à l'abréviation de la matière – et appliquer la fonction de remplissage des données des étudiants.

Cette fonction va remplir en début de fichier Excel (A1, A2, ...) les numéros, nom, prénom, groupe de chaque étudiant :

```
//Ecriture des Numero, Nom, Prenom et Groupe pour chaque étudiants.
for ($compteur = 1; $compteur <= count($info_person); $compteur++) {
    $feuille->setCellValueByColumnAndRow(0, $ligne, $info_person[$compteur]["Numéro"]);
    $feuille->setCellValueByColumnAndRow(1, $ligne, $info_person[$compteur]["Nom"]);
    $feuille->setCellValueByColumnAndRow(2, $ligne, $info_person[$compteur]["Prénom"]);
    $feuille->setCellValueByColumnAndRow(3, $ligne, $info_person[$compteur]["Groupe"]);
    $ligne++;
}
```

Lors de la création des fichiers Excel, nous créons 3 fichiers en dur : Bilan, Absence et Jury, ces trois fichiers sont pré-remplis grâce aux fonctions appelées spécifiquement pour ces fichiers :



```

if (strcmp($abreviation[count($abreviation) - 1], "Bilan") == 0) {
    fileBilan($feuille);
}
if (strcmp($abreviation[count($abreviation) - 1], "Absences") == 0) {
    fileAbsence($feuille);
}
if (strcmp($abreviation[count($abreviation) - 1], "Jury") == 0) {
    fileJury($feuille);
}

```

En fonction des besoins des fichiers Excel, le code est différent : dans le fichier Absence, nous générons des données aléatoires alors que dans Jury nous regardons si les semestres sont validés, ou non, en fonction des moyennes dans le fichier Bilan.

À la fin de cette génération, nous obtenons donc un ensemble de fichier matière similaire où le nom est différent et les informations de la matière dans le fichier le sont aussi.

Ensuite ces fichiers sont distribués aux professeurs et nous les récupérons en fin de semestre afin d'appliquer une « automatisation ».

Cette automatisation va permettre de rassembler les fichiers Excel en un seul fichier Excel. Ce fichier possèdera plusieurs feuilles : Bilan, Jury, Absence, et l'ensemble des matières :

Bilan	Absences	Jury	ProdApp	Droit	SE-3	Ang-3	Proba	GSI	ModMath	RX-2	BDA	Com	CPA	APA	Pweb-1	PT-3 ...
-------	----------	------	---------	-------	------	-------	-------	-----	---------	------	-----	-----	-----	-----	--------	----------

La fonction d'automatisation va récupérer tous les fichiers .xlsx dans le répertoire et regrouper la totalité dans un seul fichier.

