Rapport projet web: Calendrier Collaboratif

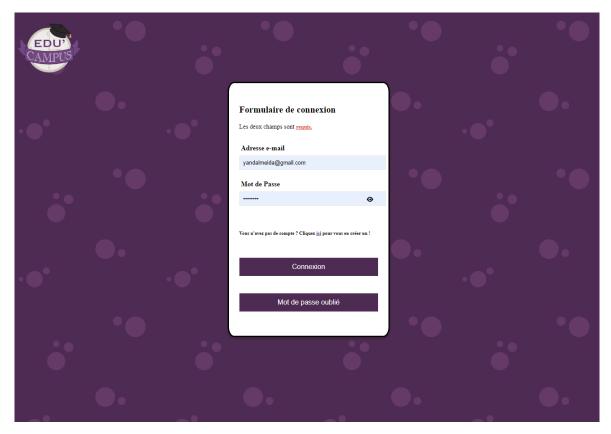
Introduction

Il nous a été demandé de réaliser le projet suivant : un calendrier Collaboratif dans lequel plusieurs personnes pourraient s'y connecter et accéder à la même page, pour le consulter et faire des modifications s'ils en avaient les droits. Ainsi, nous, Bastien Repaire et Yannick D'Almeida avons tentés de réaliser ce projet, et voici la présentation de celui-ci.

Analyse et conception

lci est la partie réservée à l'explication des fonctionnalités techniques, du code et des difficultés rencontrées.

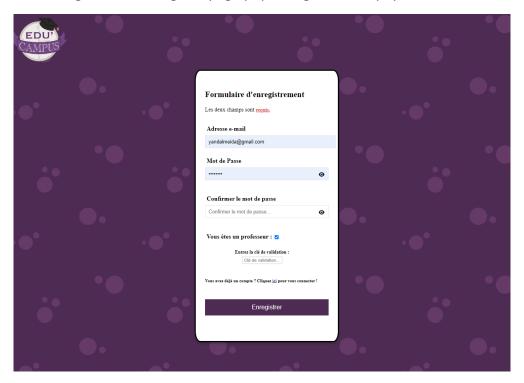
Page de connexion : login.class.php et login.page.php



Demande une adresse mail ainsi qu'un mot de passe, que doit donc renseigner l'utilisateur pour se connecter. Avec un lien vers le formulaire d'enregistrement si l'utilisateur ne possède pas de compte, un bouton mot de passe oublié qui, à l'aide d'une fonction générant un mot de passe de la page login.page.php, puis remplaçant le précédent, et de PHPMailer, service d'envoi de mail répandu qui dans notre cas est associé à gmail, envoie un mail avec ce

nouveau mot de passe à l'adresse email renseignée. Ainsi donc, la classe login.class ainsi que le php associé gèrent tous les deux cette étape. L'appui sur le bouton connexion connecte l'utilisateur si les informations entrées sont correctes.

Formulaire d'enregistrement : register.page.php et register.class.php



Adresse mail, mot de passe et confirmation du mot de passe sont demandés, ainsi qu'une case à cocher si l'utilisateur a le statut de professeur, statut octroyant la possibilité de faire des modifications sur le calendrier. Il doit par la suite entrer la clé (que nous avons initialisé à PROF2000 dans la classe register.class). A la fin de l'enregistrement, les infos nécessaires de l'utilisateur sont renseignées dans un json (dataAccount.json) qui sera lu aux tentatives de connexions.

Page du calendrier :

CAMPUS	••	(•	••	Newstam i yantalanda gymal com Hemania
i.		Emploi du	temps de l'année	2022-2023	
	A -	Marki	- Semaior du 91-05-2023	. And	Value
	\$600 Compay 1 \$131 \$132 \$132 \$1330 \$1311 \$1320 \$1311 \$1320 \$1311 \$1320 \$1311 \$1320 \$1311 \$	Groups 2 Groups 1 Gro		Oroge 2 Oroge 2 Oroge 2 Oroge 2 Program Augus Filmen Follows	Group 2
	935 1000 100	and form	IAA Lang termen 19 au 23 20 August 2000	h-y-tion h-y-tion true t	
-	1000 1013 1030 1040 1040 1041 1050 1050 1050 1050		Color to Color	Em Em	A Jointe un cours au calendrier : Non de la matier : Non de la matier : Tope de la matier : Type de la matier :
	1850 1865 1860 50 50 18415 1850	BICCI Anglis Walter G Anglis		Ab) Angi Filmin O Angin	Groupe : Name of prison. Name of the Sales.
	1960 1960 1961 1965 1965 1966	BEOD TP Wadar OI DOH OI E200		EAN TO THE PROPERTY OF THE PRO	Manning on to fault. Jour 1 Jon of to terrative. Direct discourse.
	1990 1900 1903 1903 1900 1910 1910 1910			Too.	Fin de cours : **Tourne of fin. **Sur photours somation: No.
	1904 5 1905 0 1904 5 1904 5			100	Author to coars Vender employees in charge as formulars forms

Voici donc la représentation concrète de notre projet : Un calendrier semainier de lundi à vendredi, avec une plage horaire de 8h à 19h (à l'aide d'un compteur, on initialise l'heure à 8, un timer à 0 et on incrémente de 15 pour chaque case, à 60 on remet le timer à 0 et on incrémente l'heure de 1, notre calendrier est donc divisé en cellules toutes les 15 minutes) ainsi que des groupes (groupe 1, groupe 2). Le calendrier est initialisé dans notre fichier calendrier.php, les infos liées sont stockées dans un json généré par la classe datec, classe vérifiant si un fichier lié à la semaine du calendrier existe (nous sommes sur un format Année-Semaine, exemple 2023-20) puis si ce n'est pas le cas, il crée un json avec les infos nécessaires. Dans le json du calendrier d'une semaine donnée, nous avons les heures concernées (ça va également de 15 min en 15 min comme pour les cases du calendrier), puis s'en suit les jours dans lequel on trouve les valeurs du groupe 1 et du groupe 2, dans lequel il y a une instance « val », initialisée à 0 s'il n'y a pas de cours ce jour-là, 1 s'il y a un cours et qu'il s'agit de l'heure du commencement de ce cours, et -1 pour les cellules suivant cette heure, ce qui permet donc de délimiter les cours entre eux, un cours initialisé le Lundi de 8h00 à 9h30 aura une valeur de 1 dans l'instance associée et celles qui suivront jusqu'à 9h30 auront une valeur de -1. Il y a également une instance Cours (le nom de la matière, le type du cours, le prof qui enseigne ainsi que le numéro de la salle), Class (qui dépend de la matière renseignée, si elle n'existe pas parmi celles établies dans notre scriptaddCours.php, elle est intitulée « unknowCours ». Sinon les class sont initialisé à caseNormal à la création du calendrier. Pour finir, une instance id qui vaut la concaténation du jour de l'heure du début, de l'heure de fin et du groupe concerné, id servant principalement à relier les cellules du calendrier au cours lors de l'utilisation de la fonction suppression (scriptRemoveCours.php, on sépare l'id dans la fonction associée en paramètres que l'on fournit au code lors de l'appel ajax pour supprimer le cours associé.)En haut du calendrier le logo de notre site, un clic fait revenir à la page de la semaine actuelle, à droite l'identifiant (donc le mail utilisé pour se connecter) avec le moyen de se déconnecter, renvoyant donc à la page login.

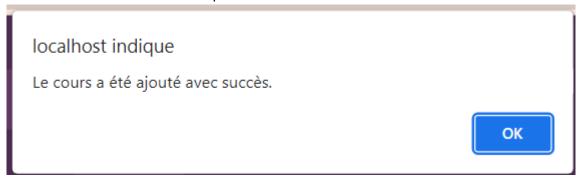
Affichée en gros, l'année actuelle et en dessous le moyen pour naviguer entre les emplois du temps, (en appuyant sur flèche de gauche semaine précédente, flèche de droite semaine suivante). Datec crée ici un calendrier s'il n'existe pas déjà à l'appui d'une flèche.

Revenons au calendrier en lui-même. Enfin, nous avons le formulaire pour ajouter un cours, accessible uniquement aux comptes ayant un statut de professeur. L'utilisateur entre le nom de la matière, le type de la matière accessible dans un menu déroulant, pareil pour ne numéro du groupe (les deux correspondent à un cours commun, dans le fichier scriptAddCours, les deux groupes sont gérés en même temps et dans le fichier calendrier.php, leur affichage commun l'est également), le numéro de la salle, le jour de la semaine (donc entre lundi et vendredi) l'horaire de début et l'horaire de fin (on a bien pris le soin d'empêcher l'utilisateur de choisir une horaire de fin définie avant l'horaire de début, ainsi qu'une option sur plusieurs semaines, définissant donc la récurrence d'un coup (nous avons opté pour un maximum de 24 semaines pour la récurrence d'un cours)

Ajoutez un cours au calendrier :	1	
Nom de la matière :		
Lang.Formel		
Type de la matière :		
Amphi	~	
Groupe:		
Groupe 1	~	
Nom du professeur :		
Gruau		
Numéro de la Salle :		
G.Amphi		
Jour:		
Mardi	~	
Début du cours :		
9h30	~	
Fin du cours :		
12h45	~	
Sur plusieurs semaines?		
Non	~	
Ajouter le cours		
Fermer		

L'appui sur le bouton ajouter le cours fait apparaître un texte explicitant à l'utilisateur qu'il faut bien renseigner tous les champs si un des champs reste vide, bouton fermer pour fermer le formulaire. Nous utilisons des alerts depuis notre script pour notifier l'utilisateur de la

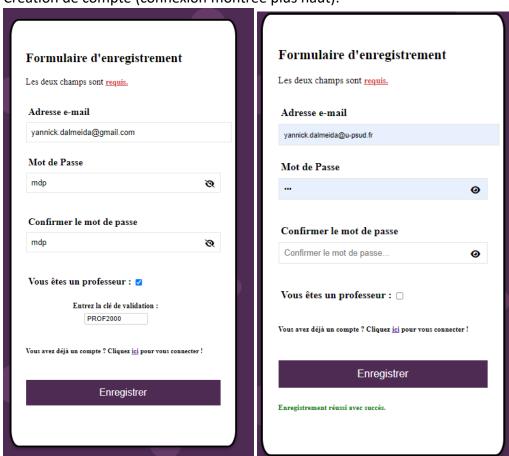
portée de leurs actions :



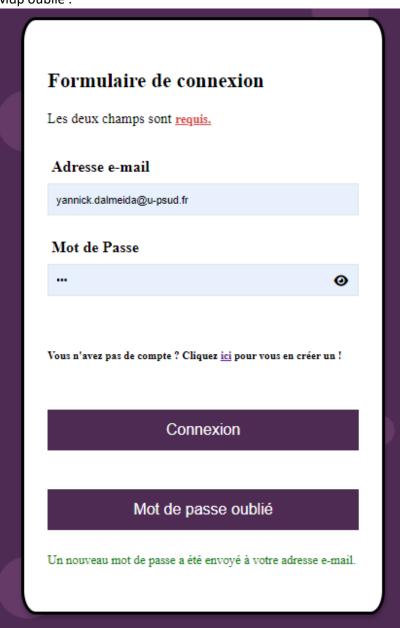
Globalement, les difficultés que nous avons rencontrées sont lié à notre méconnaissance des possibilités que nous avions à dispositions (comme le fait d'afficher sur les pages en ellemême les données et variables pour voir leur évolution, ou les différentes erreurs), ainsi que dans la conception de notre calendrier, où nous avons pu oublier de penser à des fonctionnalités qui seraient utiles.

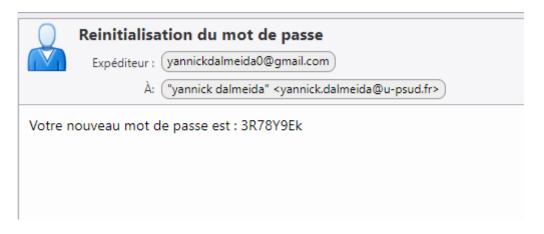
<u>Demo:</u>

Création de compte (connexion montrée plus haut):



Mdp oublié:



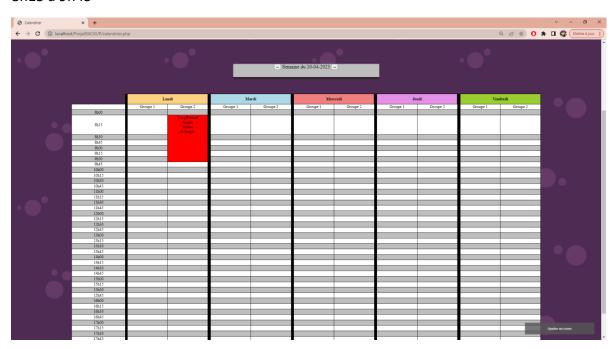




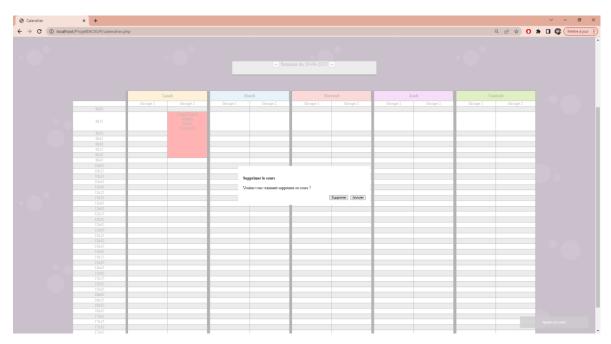
Calendrier vide:

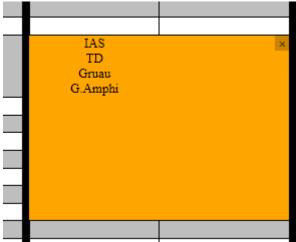


Calendrier après remplissage du formulaire d'ajout de cours pour le groupe 2 le lundi de 8h15 à 9h45

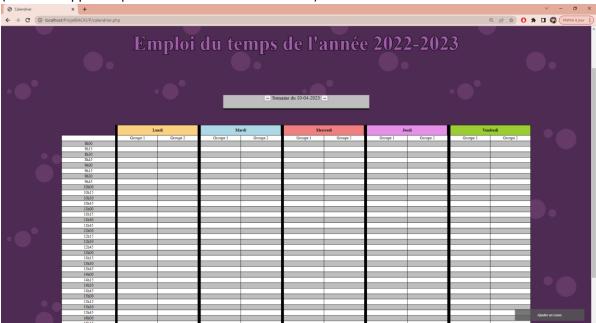


Calendrier après suppression du cours :





(une croix apparait quand on est sur la case du cours)



Conclusion et perspectives

Nous avons donc implémenté pas mal de fonctionnalités dans notre projet mais il reste bien évidemment nombre de points d'améliorations possibles :

Premièrement, de la manière dont nous avons défini notre calendrier, le texte constituant les infos d'un cours en question n'est pas centré comme on aurait voulu le faire préalablement, dans notre cas chaque cellule a une classe représentant le cours dans sa globalité mais le texte ne s'affiche que dans la 1 ère cellule. De plus, un système de paramètres de comptes n'aurait pas été de trop pour laisser à l'utilisateur la liberté de changer son mot de passe (inscrire le mot de passe précédent, puis le nouveau avec confirmation), voire de pouvoir changer son statut en celui de professeur (en entrant donc la clé nécessaire). Également la possibilité de remplacer un cours, donc en demandant à l'utilisateur s'il veut changer un cours par un autre, la possibilité de déplacer un cours (car ici cela se fait en supprimant puis en le récréant à un autre endroit), la possibilité de supprimer le même cours sur plusieurs semaines etc. Il y a donc de nombreux points d'améliorations possibles, choses qui auraient peut-être pu être possible avec du temps supplémentaire même si le problème d'affichage du cours nous demandait de revoir pas mal de bases de notre code. Finalement, ce fut un projet instructif, nous avons pu approfondir nos bases en php javascript.

Projet effectué par D'Almeida Yannick et Repaire Bastien.