

Rapport de stage



Année 2022 - 2023

Nicolas Vero

BUT Métiers du Multimédia et de l'Internet
IUT de Rouen
Université Rouen Normandie

Gribouillenet Design

Fabrice Arnaud

**Création de
thèmes WordPress et
d'applicatifs sur-mesures**

Développement web

Évaluation du stagiaire

IUT ROUEN dept MMI

Appréciation du maître de stage 2023

Évaluation du stagiaire.

Nom du maître de stage : FABRICE ARNAULD

Nom de l'entreprise : GRIBOUILLET DESIGN

Nom et prénom du stagiaire : VERO NICOLAS

Appréciation générale sur le stagiaire	
Niveau général	Nicolas a su s'intégrer parfaitement à l'équipe et résoudre les différentes problématiques posées.
Les points à améliorer	

NB : De insuffisant 1 pt à excellent 5 pts	insuffisant	moyen	bien	Très bien	Excellent
Assiduité :ponctualité,régularité,disponibilité, ...					*
Technicité : connaissance des outils, réalisation des travaux dans les délais,...					*
Organisation : collecte des informations, gestion du temps, planification des activités,...					*
Intégration : participation au sein de l'équipe, esprit d'entraide,...					*
Communication : aptitude à s'exprimer, aptitude à écrire, capacité d'écoute, ...					*
Initiative : capacité à formuler des remarques, à proposer des solutions,...					*
Implication : volonté d'aboutir à un travail de qualité, esprit de persévérance,...					*
Apprentissage : recherche personnelle de solutions, aptitude à formuler une demande d'aide,...					*
Note finale sur 40 :	40				
Commentaires :					
Très bon niveau professionnel.					

* : mettre une croix dans la colonne concernée pour chaque critère.

Date : 10/06/23

Signature :

(Merci de votre participation
à la formation de cet étudiant.)



Attestation de stage



ATTESTATION DE STAGE à remettre au stagiaire à l'issue du stage

ORGANISME D'ACCUEIL

Nom ou dénomination sociale : GRIBOUILLENET DESIGN
Adresse : 2 route de Bosnormand 27370 LE THUIT-DE-L-OISON FRANCE
Tél : 0687221415

Certifie que

LE STAGIAIRE

Nom : VERO Prénom : Nicolas Sexe : M Né(e) le : 03/03/2001
Adresse : 52 rue Lemoine 76300 SOTTEVILLE-LES-ROUEN FRANCE
Tél : 0651151680 Portable : +33651151680 Mél : nicolas.vero@univ-rouen.fr
ÉTUDIANT EN (intitulé de la formation ou du cursus de l'enseignement supérieur suivi par le ou la stagiaire) :
BUT Métiers du Multimédia et de l'Internet, Développement Web et Dispositifs Interactifs 2ème année
AU SEIN DE (nom de l'établissement d'enseignement supérieur ou de l'organisme de formation) :
Université de ROUEN Normandie

A effectué un stage prévu dans le cadre de ses études

DURÉE DU STAGE

Dates de début et de fin du stage : du 03/04/23..... au 10/06/23.....

Représentant une durée totale de ...10..... (~~Nombre de mois~~ / Nombre de semaines) (rayer la mention inutile)

La durée totale du stage est appréciée en tenant compte de la présence effective du stagiaire dans l'organisme, sous réserve des droits à congés et autorisations d'absence prévus à l'article L.124-13 du code de l'éducation (art. L.124-18 du code de l'éducation). Chaque période au moins égale à 7 heures de présence consécutives ou non est considérée comme équivalente à un jour de stage et chaque période au moins égale à 22 jours de présence consécutifs ou non est considérée comme équivalente à un mois.

MONTANT DE LA GRATIFICATION VERSÉE AU STAGIAIRE

Le stagiaire a perçu une gratification de stage pour un montant total de1275.75 €..... €

L'attestation de stage est indispensable pour pouvoir, sous réserve du versement d'une cotisation, faire prendre en compte le stage dans les droits à retraite. La législation sur les retraites (loi n°2014-40 du 20 Janvier 2014) ouvre aux étudiants dont le stage a été gratifié, la possibilité de faire valider celui-ci dans la limite de deux trimestres, sous réserve du versement d'une cotisation. La demande est à faire par l'étudiant dans les deux années suivant la fin du stage et sur présentation obligatoire de l'attestation de stage mentionnant la durée totale du stage et le montant total de la gratification perçue. Les informations précises sur la cotisation à verser et sur la procédure à suivre sont à demander auprès de la Sécurité sociale (code de la sécurité sociale art. L.351-17 - code de l'éducation art.D.124-9).

Fait à ...Elbeuf..... le ...10/06/23.....

Nom, fonction et signature du représentant de l'organisme d'accueil

Fabrice Arnould, CEO

Date d'impression : 30-05-2023 09:55:14

Remerciements

Je tiens à remercier les personnes ayant contribué au succès de ce stage et qui m'ont aidé lors de la rédaction de ce rapport.

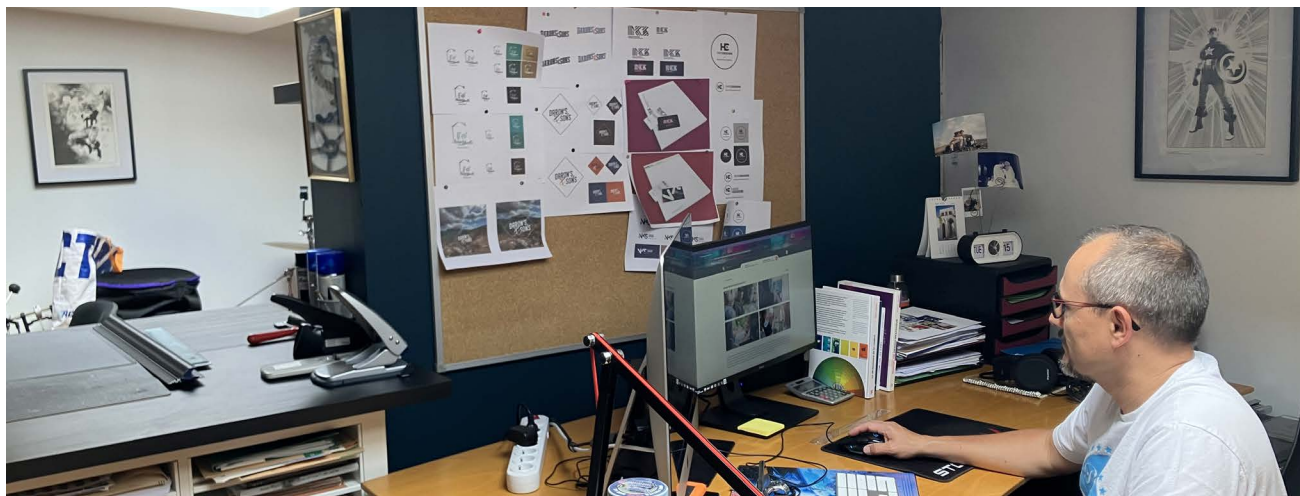
Je tiens tout d'abord à exprimer ma profonde gratitude envers l'entreprise Gribouillenet et en particulier envers mon maître de stage, Monsieur Fabrice Arnould, pour m'avoir accueilli et m'avoir offert l'opportunité d'effectuer mon stage au sein de son entreprise. Sa patience, sa bienveillance et ses connaissances ont été d'une grande importance pour mon développement personnel ainsi que pour mon intégration au sein de l'entreprise.

Je tiens également à remercier mon tuteur de stage, Monsieur Joetzjer, pour m'avoir aidé à la rédaction de ce rapport de stage.

Sommaire

L'entreprise	6
Le stage	7
Mes outils de travail	7
Présentation des missions effectuées	7
Travaux réalisés	8
EMIJ	
Révision des maquettes	9
Création de la base de travail	9
Création du style pour les pages	11
Fonctionnalités du site	13
Utilisation de WordPress	14
Volant Séminaire	
Travailler en équipe	16
Création du formulaire	16
Problèmes de \$_SESSION	18
Ecriture du mail automatique	19
Création de devis	21
Galerie aléatoire	22
Apports du stage	24
Connaissances acquises	24
Bilan humain	25
Poursuite d'études	25
Annexes	26
Glossaire	26
Illustrations	27

L'entreprise



Gribouillenet Design est une micro auto-entreprise située à Elbeuf, spécialisée dans la réalisation de sites et de thèmes WordPress et d'applicatifs sur-mesures. Bien que le secteur d'activité soit principalement le web, la société, et donc par extension mon tuteur (Fabrice Arnault), est spécialisé dans de nombreux autres domaines tels que le design graphique, la direction artistique, la vidéo et le design.

Les locaux dans lesquels l'entreprise est installée sont partagés avec deux autres entreprises freelances spécialisées dans les domaines du covering (le fait de recouvrir une carrosserie de voiture de film adhésif afin de changer sa couleur). Ces deux entreprises partagent des ateliers communs, et des bureaux.

Bien que les trois entreprises ne soient pas directement reliées entre elles, j'ai pu en apprendre beaucoup sur ces domaines grâce à des discussions où à l'observation des phases de conception sur Photoshop. Pour sa part, Gribouillenet Design est installé dans un bureau séparé des ateliers. Les locaux sont découpés en deux parties jointes : l'espace de travail (avec bureaux, ordinateurs), et l'espace pour manger.

J'étais installé sur un bureau avec mon ordinateur personnel, ainsi qu'une tablette faisant office de deuxième écran. J'ai également eu l'occasion de travailler sur l'ordinateur de mon tuteur lorsque celui-ci était en déplacement professionnel ou en rendez-vous client.

Le stage

Mes outils de travail

WordPress est un CMS (un CMS - *Content Management System* - est un logiciel grâce auquel il est possible de créer, de gérer et de modifier facilement un site web). Durant mon stage, mes missions ont toutes consisté à réaliser des thèmes WordPress sur mesure pour différents clients.

Un thème sur mesure est globalement ce qui permet de définir la présentation et l'apparence du site. Ainsi, c'est un type de développement qui a plusieurs points communs avec le développement d'un site web "classique" (utilisant HTML5 et CSS3), mais qui apporte également énormément d'avantages.

Tout d'abord, en développant un thème sur mesure, nous ne sommes pas amenés à concevoir les pages les unes après les autres, mais plutôt à concevoir des règles qui seront appliquées à un ensemble de données (toutes les pages du même type posséderont la même architecture).

Cette méthode a deux grands avantages principaux :

- il est extrêmement facile de rajouter du contenu via l'interface back-end de WordPress ;
- étant donné que l'ensemble des pages possèdent la même architecture, on peut changer le fichier sur lequel elles s'appuient pour ainsi modifier le style de toutes les pages concernées par ces règles.

En plus de ces deux avantages, le développement de thèmes sur mesure permet d'utiliser au mieux la méthode DRY (Don't Repeat Yourself). C'est une manière de programmer dans laquelle le code doit éviter autant que possible de se répéter pour éviter les redondances, ainsi que les erreurs potentielles.

Enfin, WordPress utilise une hiérarchie des différents fichiers très précise (voir Template Hierarchy en annexe), ce qui permet d'organiser son projet de manière très structurée.

Présentation des missions effectuées

Durant mon stage, je me suis vu confier plusieurs projets différents. J'ai décidé de vous présenter mon travail sur deux d'entre eux. Ce sont ceux qui m'ont demandé le plus de temps, le plus d'investissement, et parallèlement ceux qui m'ont permis d'acquérir le plus de compétences.

Le premier projet était la création du site d'une école de musique, de la maquette jusqu'au développement.

Le deuxième était le développement d'un site proposant des séminaires à des entreprises.

Les travaux réalisés

L'EMIJ

Création d'un site pour une école de musique d'Elbeuf

Volant Séminaire

Production collaborative d'un site de réservation de séminaires d'entreprises

Révision des maquettes

Pour commencer, j'ai dû retravailler la maquette du site web à l'aide du logiciel Adobe XD. Cette dernière avait en effet certains problèmes, comme par exemple des éléments visuels n'ayant pas toujours les mêmes propriétés.

J'ai également été chargé de réaliser les maquettes pour la version mobile du site. Pour ce faire, j'ai repris la maquette finalisée et j'ai redisposé les éléments pour les faire tenir sur un format mobile. Cette partie du travail n'a pas été très longue étant donné qu'il s'agissait uniquement d'une redistribution des différents éléments déjà conçus.

Pour finir, j'ai été chargé de réaliser une landing page (page mise en ligne pour avertir de l'arrivée prochaine du site web). J'ai décidé de réutiliser les éléments que je trouvais essentiels. Cette page contient donc le *slider* présent en haut de l'index, de cette manière l'utilisateur pourra avoir un rapide aperçu des activités qui sont proposées au sein de l'EMIJ. Elle contient également un petit texte explicatif permettant d'avertir les utilisateurs que le site de l'EMIJ est en développement. Enfin, la page contient le formulaire d'abonnement à la newsletter (pour pouvoir recevoir les différentes actualités directement dans leur boîte mail), ainsi que la localisation et l'extrait d'un article présentant l'école.

Voir la maquette

Création de la base de travail

Une fois la maquette terminée, j'ai pu commencer le développement du site web. Pour ce faire, j'ai utilisé le logiciel local by flywheel. C'est un logiciel qui permet de déployer un site WordPress très simplement en local sur son ordinateur.

Une fois WordPress installé, j'ai commencé par lister les différents composants du site web, et j'ai créé différents Custom Post Type (CPT) pour chacun d'entre eux. Un CPT est un type de contenu personnalisé. Par défaut, WordPress permet de ne créer que des articles, et des pages. Les CPT permettent de définir un nouveau type de contenu (qui contiendra ici l'ensemble des pages concernant les instruments).

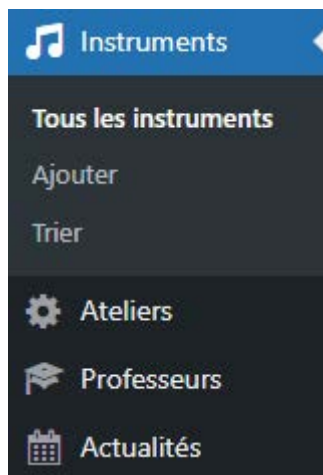
J'ai donc été amené à créer un CPT pour les éléments suivants :

- les instruments - *les différents types d'instruments enseignés ;*
- les ateliers - *les différents types d'ateliers dispensés ;*
- les professeurs - les différents enseignants ;
- les actualités - les actualités concernant l'EMIJ.

Voir création d'un Custom Post Type

Les Custom Post Type créés se retrouvent dans l'interface WordPress, et peuvent être utilisés comme une page ou un article classique.

L'un des gros avantages de cette façon de travailler est qu'elle permet d'une part de regrouper tous les éléments du site web ayant les mêmes caractéristiques, par exemple, nous pourrions traiter de la même manière toutes les articles de type instruments grâce au fait qu'il soit du même type. Une fois ces CPT créés, j'ai été chargé de générer les différents contenus textuels de chaque page.



Interface de WordPress

Les bois

Sous-titre

Lorem ipsum dolor sit amet.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Integer nec rhoncus neque, nec vehicula nisi. Suspendisse gravida turpis vitae nulla semper, at pretium ex efficitur. Mauris ac justo viverra, lacinia sapien id, malesuada justo. Suspendisse a nulla commodo, dignissim elit nec, fermentum elit. Praesent a augue felis. Donec ullamcorper leo sollicitudin est feugiat vestibulum. Morbi facilisis posuere consequat. Integer ullamcorper at magna id tristique.

Le saxophone

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Integer nec rhoncus neque, nec vehicula nisi. Suspendisse gravida turpis vitae nulla semper, at pretium ex efficitur. Mauris ac justo viverra, lacinia

La clarinette

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Integer nec rhoncus neque, nec vehicula nisi. Suspendisse gravida turpis vitae nulla semper, at pretium ex efficitur. Mauris ac justo viverra, lacinia

La flûte traversière

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Integer nec rhoncus neque, nec vehicula nisi. Suspendisse gravida turpis vitae nulla

Contenu d'un Custom Post Type de type Instruments

Une fois l'ensemble des contenus rédigés, je n'avais plus qu'à afficher les informations des différentes pages à l'aide d'une boucle WordPress.

La boucle WordPress est un système assez simple. Elle permet de définir une règle, et d'aller chercher l'ensemble des contenus qui répondent à cette dernière. Dans cet exemple, nous voulons obtenir les informations des pages de type instruments.

Nous faisons donc une requête à l'aide de la fonction `WP_Query()`, dans laquelle nous allons préciser un certain nombre de paramètres.

```
1  $query = new WP_Query( array(
2      'post-type'      => 'instruments',
3      'posts_per_page' => -1,
4      'post_parent'    => 0
5  ) );
6
7  while( $query->have_posts() ) :
8      $query->the_post();
9
10     the_post_thumbnail();
11     the_title();
12     the_excerpt();
13     the_permalink();
14
15 endwhile;
```

La boucle WordPress

Une fois la requête effectuée, on utilise une boucle *while*, qui va parcourir l'ensemble des posts de type instruments. Dans cet exemple, on affiche pour chacun (dans l'ordre) :

- l'image mise en avant ;
- le titre du post ;
- l'extrait du post ;
- l'url qui mène jusqu'à la page.

Une des tâches que j'ai réalisée a été de mettre l'ensemble de ces informations au sein de balises HTML, et de réaliser une structure que je pourrais éventuellement mettre en style par la suite.



Les bois

Saxophone, Clarinette, Flûte Traversière

[En savoir plus](#)

Résultat brut de la requête

Création du style pour les pages

Pour styliser les pages, j'ai été amené à utiliser le langage Less. C'est un langage intermédiaire, car celui-ci ne peut pas être lu par une page web (qui a elle besoin d'un fichier CSS). Le Less nécessite donc d'installer un compilateur permettant à partir d'un fichier .less de générer un fichier .css correspondant. Pour ce faire, j'ai utilisé une extension de l'éditeur de code Visual Studio Code nommée *Easy Less* permettant de générer de nouveau un fichier .css à chaque enregistrement du fichier .less.

Le langage Less a été relativement simple à apprendre étant donné qu'il ressemble très fortement à la syntaxe du CSS classique. Cela m'a permis de rapidement m'adapter à ce nouvel outil et de commencer à l'utiliser efficacement dans mes tâches de stylisation des pages web.

Ce langage possède de nombreuses fonctionnalités très utiles qui m'ont permis de gagner beaucoup de temps, et de produire un résultat plus homogène qu'avec du CSS classique.

L'une des fonctionnalités les plus importantes du Less est le principe d'encapsulation. En CSS classique, si l'on veut appliquer un style à un élément contenu dans un parent, il faut systématiquement réécrire le nom du parent devant celui de l'enfant. Avec Less, un élément parent peut encapsuler un enfant, c'est à dire de l'entourer avec ses propres accolades. Cela permet de gagner énormément de temps, et d'avoir un code moins difficile à lire.

Voir encapsulation Less

Le langage Less permet également de définir un ensemble de variables, qui peuvent ensuite être utilisées dans mes règles de style. Cela m'a permis de centraliser les valeurs couramment utilisées, telles que les couleurs, les tailles de police, et de les réutiliser facilement dans tout le projet. Cela permettait également de pouvoir très facilement changer ces valeurs, et que ce changement se répercute directement sur l'ensemble du site.

Voir variables Less

Parfois une simple variable ne suffit pas. Cela a par exemple été le cas lorsque j'ai voulu créer les déformations sur les éléments de la page, comme pour donner cet effet de diagonale à la fin et au début d'une section. Pour éviter de devoir constamment écrire les mêmes règles pour chaque élément, le langage Less propose une solution appelée des *mixins*. Ce sont des structures ressemblant aux fonctions dans un langage de programmation, et qui permettent d'appliquer à un élément des styles sur simple appel de cette *mixin*.

Voir mixins Less

Enfin, la langage Less permet de segmenter son code pour que celui-ci soit beaucoup mieux organisé. Nous pouvons par exemple créer un fichier pour le style du footer (partie inférieure d'une page web), et simplement l'appeler depuis un autre fichier Less. Cela permet d'avoir un fichier central, qui importe l'ensemble des autres fichiers. Grâce à cela, nous pouvons définir nos variables dans un fichier, nos mixins dans un autre, et appeler ces mêmes fichiers dans le fichier principal. Cette méthode permet de beaucoup mieux structurer son code, car il n'est plus composé d'une grosse feuille de style CSS, mais de plusieurs petits fichiers.

Voir imports Less

En plus d'être très simple d'utilisation, cet outil m'a permis de gagner énormément de temps sur l'écriture de mon style. De ce fait, je pense dorénavant l'utiliser au profit du CSS classique dans mes futurs projets.

Fonctionnalités du site

Le site de l'EMIJ doit également permettre aux futurs administrateurs de pouvoir facilement gérer les différentes informations directement depuis l'interface back-end de WordPress.

Le système marche en deux étapes distinctes :

- Dans un premier temps, l'administrateur doit se rendre sur la page d'un professeur et lui attribuer un ou plusieurs cours. Une fois ces changements effectués, le professeur apparaît sur les pages auxquelles il a été attribué ;
- Dans un second temps, l'administrateur doit entrer les horaires de chaque cours afin que ceux-ci soient mis en avant sur la page.

Cela permet au site de pouvoir être mis à jour facilement, en rajoutant, modifiant ou supprimant les informations qu'il contient.

Grâce à des fonctionnalités comme cela, le site n'est pas simplement une vitrine, mais également un site d'informations pouvant être mis à jour extrêmement facilement (en ajoutant des professeurs, en modifiant leurs horaires et les cours dans lesquels ils interviennent).



Les horaires d'un cours ainsi que les professeurs y participant

Utilisation de WordPress

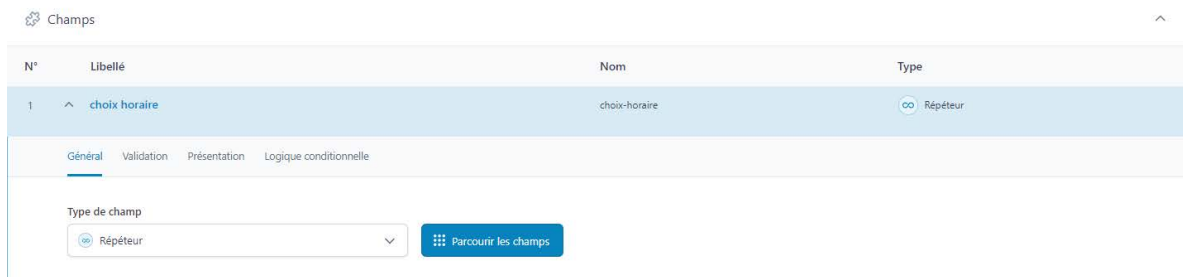
Pour faire le système de professeurs et de cours, j'ai appris à utiliser le *plugin* Advanced Custom Fields (ACF). Cette extension nous permet de créer des champs de formulaires pouvant être remplis sur les pages elles-mêmes.

Par défaut, WordPress propose également des champs pouvant être remplis par l'utilisateur, mais ceux-ci sont malheureusement très limités et ne permettent pas d'être aussi flexibles que les champs de l'extension ACF.

ACF marche avec un système de groupe de champs, pouvant être réutilisés sur différentes pages. Cet ensemble pourra apparaître sur les pages de notre choix, et contiendra l'ensemble des champs pouvant être rempli par l'utilisateur.

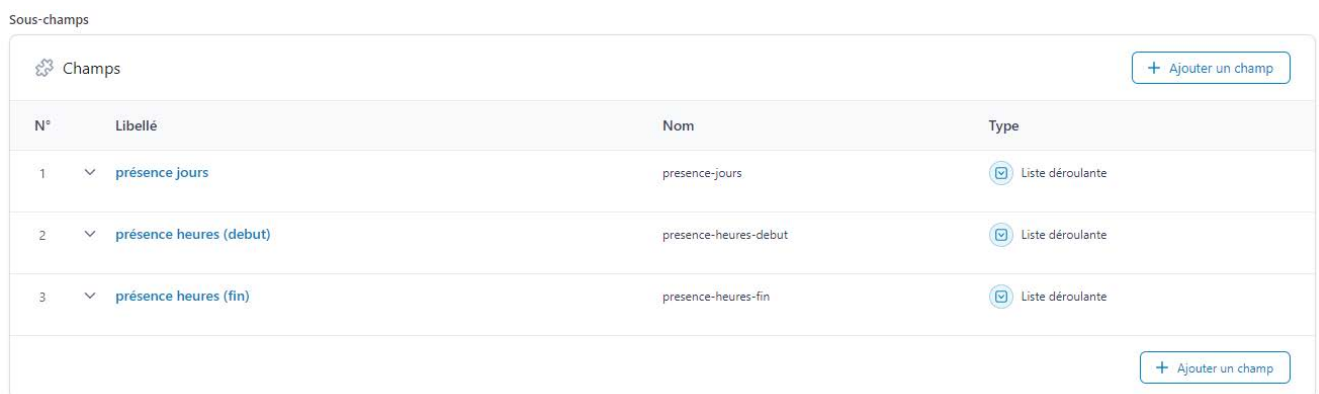
Pour créer le système des professeurs et de leurs horaires, j'ai utilisé un champ d'un type particulier : un répéteur. Un répéteur est un champ qui comme son nom l'indique se répète. Celui-ci contient en effet en son sein un sous groupe de champs pouvant se dupliquer à l'infini.

Prenons comme exemple une liste de course : cette liste peut contenir autant de produits que nécessaire, et à chaque produit correspond une quantité. C'est exactement le même système ici. L'utilisateur peut rajouter autant de jours de présence qu'il le désire, et à chaque jour de présence correspondent les horaires de début et de fin de cours.



The screenshot shows the 'Champs' (Fields) interface in the WordPress admin. It features a table with columns: N°, Libellé, Nom, and Type. The first row shows a field named 'choix horaire' with the type 'Répéteur'. Below the table, there are tabs for 'Général', 'Validation', 'Présentation', and 'Logique conditionnelle'. The 'Général' tab is active, showing a 'Type de champ' dropdown set to 'Répéteur' and a 'Parcourir les champs' button.

Création du champ de type répéteur dans ACF



The screenshot shows the 'Sous-champs' (Sub-fields) interface in the WordPress admin. It features a table with columns: N°, Libellé, Nom, and Type. The table lists three sub-fields for the 'présence jours' field: 'présence jours', 'présence heures (debut)', and 'présence heures (fin)'. Each sub-field has a 'Liste déroulante' (Dropdown list) type. There are '+ Ajouter un champ' (Add field) buttons at the top right and bottom right of the table.

Les sous-champs apparaissant pour chaque répétition

Enfin, on décide dans quelles conditions nos différents champs ACF doivent apparaître. Dans le cas présent, je devais attribuer des professeurs et des heures de cours pour deux éléments : les différents instruments et les ateliers proposés.

Règles ?

Afficher ce groupe de champs si

Type de publication

est égal à

Atelier

et

ou

Type de publication

est égal à

Instrument

et

ou

Ajouter un groupe de règles

Définition des règles d'apparition des champs

choix horaire

Attention à bien respecter l'ordre des jours de la semaine

	présence jours	présence heures (debut)	présence heures (fin)
1	Lundi	9	23
2	Mercredi	15	17
3	Samedi	13	14

[Ajouter un élément](#)

Les champs ACF sur les Custom Post Type Instruments et Ateliers

Avec cette manière de procéder, la partie back-end de WordPress (là où sont habituellement créées de toutes pièces les pages du site), devient uniquement un système pour stocker des informations. Ce système a l'avantage d'être très facilement modifiable par quelqu'un n'ayant aucune compétence en développement.

Grâce à PHP, je peux ensuite aller lire les données, et pouvoir donner le résultats que je vous ai présenté dans la partie précédente.

Voir traitement des données en PHP


Volant séminaire

Travailler en équipe

Contrairement au projet pour l'EMIJ, je suis arrivé sur ce projet alors qu'il était en cours de développement. Pour faciliter le travail en coopération, j'ai donc été amené à utiliser Git et le service d'hébergement Bitbucket (un équivalent de GitHub).

Git est un logiciel de contrôle de version. En d'autres termes, c'est un outil qui permet de créer des étapes clés lors d'un projet. Chaque étape est appelée commit. Chaque commit contient donc le code du projet tel qu'il était à ce moment-là.

Travailler avec Git a eu énormément d'avantages étant donné que nous étions à deux à travailler sur un même projet en même temps. Cela nous permettait de très facilement récupérer les modifications effectuées par l'un ou par l'autre, et de pouvoir modifier en même temps un seul et même fichier.



Extrait du graphe des commits du projet

Merge branch 'intégration' of https://bitbucket.org/Gribouillenet/volant-seminaire/src/int%C3%A9gration	16 May 2023 15:20	Nicolas Vero	5ea407de
finalisation implémentation des devis en backoffice révision filters	16 May 2023 15:17	gribouillenet	148f8667
finalisation implementation hotels	16 May 2023 15:08	Nicolas Vero	ebf4c658
finalisation de la creation des devis automatiques	16 May 2023 14:12	Nicolas Vero	dad397f9
Merge branch 'master' of https://github.com/NicolasVero/volant_seminaire	16 May 2023 13:23	Nicolas Vero	8007ac00
création fonction devis	16 May 2023 12:55	gribouillenet	ca7aa329
Correction bug page soumission	15 May 2023 16:43	gribouillenet	9ec8aaf8
debug probleme wordpress	15 May 2023 14:42	gribouillenet	8b240e83
mise en place du menu navigation css + jquery	12 May 2023 19:04	gribouillenet	f5a020fa
début création menu	12 May 2023 13:53	gribouillenet	61b947f7
modif name erreur	12 May 2023 10:51	gribouillenet	48c66df0
gestion affichage des erreurs php	12 May 2023 10:49	gribouillenet	d0ab3aca
finalisation suppression d'un ajout activité ajout de div pour affichage html erreur sur champs input...	12 May 2023 10:15	gribouillenet	54700c55
Merge branch 'master' of https://github.com/NicolasVero/volant_seminaire	12 May 2023 09:27	Nicolas Vero	d1da6871
effet rideau sur ajout activité	12 May 2023 09:26	gribouillenet	29e9f7bf

Extrait du graphe des commits du projet

Création du formulaire

Le site de Volant Séminaire marche de la manière suivante :

- Un utilisateur arrive sur le site où lui sont présentées différentes activités ;
- Il peut en sélectionner une et remplir le formulaire de demande de devis ;
- Au cours de la complétion du formulaire, il peut décider d'ajouter ou d'enlever dynamiquement des activités au formulaire.

Escape Game En Mini Austin Et 2CV
Plongez dans les années 70 et devenez un véritable espion à bord d'une Mini Austin ou d'une 2CV.

Nombre de personnes*

Lieu du séminaire*

Date de l'activité*

Horaires* de à

Partie du formulaire pouvant se répéter

Bien que le site soit réalisé grâce à WordPress, nous avons décidé de gérer l'entièreté du formulaire en PHP uniquement, sans avoir recours à un *plugin* comme Contact Form 7 pour le développer.

Ainsi, il était beaucoup plus facile pour nous de le moduler, en ajoutant et en enlevant des éléments grâce à JavaScript.

Voir ajout / suppression d'éléments du formulaire

Comme le montre cette image, les différentes entrées du formulaire possèdent toutes un nom différent qui est construit de la manière suivante : id_activite-id représente par exemple l'id correspondant à l'activité. De cette manière, on peut identifier chaque champ de formulaire.

Une fois le formulaire rempli par l'utilisateur, celui-ci est analysé pour s'assurer qu'il a correctement été rempli. Si ce n'est pas le cas, chaque champ comportant une erreur sera identifié, et chaque message d'erreur sera stocké dans un tableau.

Une fois le traitement des erreurs effectué, on retourne sur la page du formulaire de l'activité.

Voir détection des erreurs

Seulement, cela pose un problème. La page de base ne contient que l'activité sur laquelle avait cliqué l'utilisateur en premier, les autres activités ont disparu, étant donné qu'elles étaient uniquement ajoutées dynamiquement à la page à l'aide de JavaScript.

Pour contourner ce problème, j'ai décidé de stocker dans un tableau l'ensemble des IDs des activités contenues dans la page au moment où l'utilisateur a appuyé sur le bouton de soumission de formulaire (un ID est un identifiant unique à chaque élément).

En utilisant cette méthode, je n'avais qu'à faire une boucle pour écrire autant d'activités qu'il y en avait à la base, seulement en se basant sur leur id. De plus, chaque formulaire sera déjà pré-rempli avec les données qui avaient été fournies par l'utilisateur, pour lui éviter de devoir réécrire l'entièreté du formulaire à chaque erreur.

Ce processus va continuer jusqu'au moment où l'utilisateur aura donné un formulaire qui ne génère aucune erreur. Une fois cela fait, le script PHP va utiliser la fonction *header()* pour rediriger l'utilisateur sur des pages de traitement de données.

Seulement, le fait de faire une redirection comme celle-là nous coupe totalement de WordPress étant donné que nous ne sommes plus sur aucune page, seulement sur un fichier PHP écrit à la main.

Problèmes de \$_SESSION

L'un des problèmes que j'ai rencontré durant ce projet est justement lié au fait que le traitement des données ne se faisait plus au sein de l'architecture de WordPress. Etant donné que je n'avais pas accès aux fonctionnalités de WordPress, ainsi qu'aux données nécessaires à l'exécution du code, j'ai dû trouver un moyen de faire passer les données sans avoir besoin de dépendre de WordPress.

```
1 <input type="hidden" name="blog_info" value="<?= $blog_info ?>">
2 <input type="hidden" name="blog_url" value="<?= $blog_url ?>">
3 <input type="hidden" name="admin_email" value="<?= $blog_admin ?>">
4 <input type="hidden" name="id_principale" value="<?= $activiteID ?>">
5 <input type="hidden" name="activite_slug" value="<?= $activite_slug ?>">
```

Champs cachés du formulaire

Pour pallier ce problème, j'ai décidé de stocker toutes les informations dont j'aurai besoin dans des entrées de formulaire invisibles pour l'utilisateur (formulaire que l'utilisateur a rempli). De ce fait, je pouvais accéder à certaines données sans avoir à interroger la base de données de WordPress.

Mais le fait de ne pas pouvoir interagir avec WordPress a également été contraignant car je ne pouvais pas utiliser les sessions pour y stocker des données d'une page à une autre à cause de l'envoi prématuré des entêtes http.


J'ai donc décidé d'apprendre le fonctionnement des cookies, pour mettre en place un système permettant de récupérer les données dont j'avais besoin depuis la session. J'ai décidé d'utiliser le système des cookies, pour pouvoir stocker l'information du côté du client, et pouvoir la récupérer sur la page suivante (où la session n'était plus accessible). Cette méthode avait l'avantage d'être simple à mettre en place.

Voir fonction de création de cookies

Bien que les données stockées dans ces cookies ne soient pas sensibles (données issues d'informations que l'utilisateur connaît), j'ai volontairement mis un temps de vie faible afin que ces données ne soient plus accessibles après 10 secondes.

La durée de vie du cookie est définie grâce au timestamp Unix (nombre de secondes écoulées depuis le 1er janvier 1970 à 00h00). On ajoute donc 10 secondes à ce timestamp, et cela nous donne la date de péremption du cookie.

Une fois le processus terminé, j'ai fait en sorte de rendre les cookies inaccessibles en appelant une fonction qui donne à chaque cookie un timestamp datant d'une heure dans le passé, les rendant inutilisables.



```

1  function destroy_cookies() {
2      foreach ($_COOKIE as $cookieName => $cookieValue) {
3          setcookie($cookieName, '', time() - 3600, '/');
4      }
5  }

```

Fonction de destruction des cookies

Le fait de donner une durée de vie extrêmement faible aux cookies et de les supprimer lorsque le processus est terminé permet d'avoir une double sécurité. En effet, même si pour une raison quelconque le site web ne peut pas aller jusqu'à l'appel de la fonction `destroy_cookies()`, les cookies seront tout de même inaccessibles rapidement.

Ecriture du mail automatique

Pour notifier à l'utilisateur et à l'administrateur qu'une demande de devis vient d'être réalisée, j'ai travaillé sur un système d'envoi de mails.

Le système de mails est assez simple, il récupère les informations saisies par l'utilisateur dans le formulaire, et va les mettre en forme pour créer deux mails : un pour l'utilisateur et l'autre pour l'administrateur.

Pour construire la structure du mail, j'ai dû utiliser une structure HTML bien précise. En effet, les clients de messageries ne prennent généralement pas en charge les règles CSS avancées. J'ai donc dû utiliser les tableaux HTML, pour être sûr que la pagination marcherait sur l'ensemble des clients de messagerie.

Voir structure simplifiée des mails

Le mail contient également un numéro de devis qui servira de référence à cette demande. La référence est construite de la manière suivante :

230525NV_XX

23 : *année de demande du devis*

05 : *mois de demande du devis*

25 : *jour de demande du devis*

NV : *initiales de la personne ayant fait la demande de devis*

XX : *série de deux lettres aléatoires*

Voir fonction de création de référence



Bonjour Nicolas Vero

Référence de votre devis : 230525NV_XX

Vous avez fait une demande de devis, et nous vous en remercions.

Récapitulatif de votre demande :

Escape Game en Mini Austin et 2CV

Pour 12 personnes,
Le 2023-08-17, à Paris, de 11h30 à 19h00

© 2023 Volant Séminaire

Mail destiné à l'utilisateur



Un utilisateur a fait une demande de devis

Référence du devis : 230525NV_XX

Nicolas Vero
01 23 45 67 89
nicolas.vero@univ-rouen.fr
Gribouillenet

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Quisque eu posuere enim. Pellentesque quis felis id libero consectetur luctus ac nec urna. Aenean quis turpis sit amet nisl tincidunt faucibus et non est. Maecenas iaculis tortor sit amet urna sagittis, ac eleifend enim tempor. Nullam luctus nec est ut vulputate. Etiam ultrices consequat imperdiet. Vestibulum laoreet nulla nec nunc sollicitudin maximus. Nam nec aliquet erat, nec elementum eros.

Escape Game en Mini Austin et 2CV

Pour 12 personnes,
Le 2023-08-17, à Paris, de 11h30 à 19h00

© 2023 Volant Séminaire

Mail destiné à l'administrateur

Création de devis

J'ai également décidé de développer une fonctionnalité qui laisse une trace du devis au sein de WordPress lui-même (un Custom Post Type de type Devis).

Cette fonctionnalité est particulièrement intéressante car elle permet de garder un écrit du devis au cas où les mails ne seraient pas arrivés ou s'ils avaient été supprimés par erreur.

Pour réaliser cela, je me suis servi de la fonction WordPress `wp_insert_post()`, pour générer automatiquement des Custom Post Type de type devis.

Voir insertion de post dans WordPress

<input type="checkbox"/>	Titre	Auteur/autrice	Date
<input type="checkbox"/>	230526NN_FL	Admin-VOLanTSemiNE	Publié 26/05/2023 à 9h23
<input type="checkbox"/>	230525NV_XX	Admin-VOLanTSemiNE	Publié 24/05/2023 à 22h43
<input type="checkbox"/>	Titre	Auteur/autrice	Date

Visualisation des Custom Post Type Devis depuis WordPress

230525NV_XX

Référence du devis [230525NV_XX]

Demandé par Nicolas Vero

Fiche contact :

Email : <nicolas.vero@univ-rouen.fr>

Téléphone : 01 23 45 67 89

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Quisque eu posuere enim. Pellentesque quis felis id libero consectetur luctus ac nec urna. Aenean quis turpis sit amet nisl tincidunt faucibus et non est. Maecenas iaculis tortor sit amet urna sagittis, ac eleifend enim tempor. Nullam luctus nec est ut vulputate. Etiam ultrices consequat imperdiet. Vestibulum laoreet nulla nec nunc sollicitudin maximus. Nam nec aliquet erat, nec elementum eros. Mauris ac tellus ante. Donec efficitur sed sapien et bibendum. Proin ex lorem, aliquam sit amet quam eu, venenatis molestie ex. Mauris pretium metus vitae vestibulum gravida. Nullam urna velit, ullamcorper venenatis sagittis vel, varius ut odio. Mauris lorem nulla, accumsan ac pharetra a, efficitur ut urna. Nunc in fringilla eros.

Escape Game en Mini Austin et 2CV

Nombre de personnes : 12 personne(s)

Date, heures, et lieu souhaités : 17.08.2023, de 11:30 à 19:00, à Paris

Contenu du Custom Post Type Devis

Galerie aléatoire

Lors du développement de ce site web, j'ai eu l'occasion de voir beaucoup d'échanges entre mon tuteur et le commanditaire du site, et ainsi de comprendre qu'il est parfois difficile de cerner précisément les besoins de son client dès le lancement du projet. Ce problème s'illustre parfaitement avec une des fonctionnalités que le client voulait sur son site : une galerie d'images pour chaque activité proposée.

Nous avons créé une galerie fixe, dans laquelle les images avaient des places définies à l'avance.

Malheureusement, cette galerie d'images ne convenait pas au client, qui voulait que cette dernière ne soit pas toujours identique, mais que les images se positionnent avec des tailles et des placements aléatoires.

Après plusieurs pistes de réflexion, je me suis lancé sur le développement d'une galerie fonctionnant de la manière suivante :

Les grilles pouvant être générées sont écrites en Less (à l'aide d'une grille CSS).


A chaque fois qu'un utilisateur arrive sur la page ou la rafraîchit, une classe CSS sera sélectionnée au hasard en fonction de deux paramètres : le nombre minimum et le nombre maximum d'images pouvant être affichées.

Ces paramètres sont directement renseignés dans la partie back-office de chaque activité, ainsi que l'entièreté des images pouvant apparaître dans la galerie.

Galerie images

Galerie

Ajout de photos à la galerie de l'activité.



Ajouter à la galerie

Actions de groupe

Min de photo

Nombre minimum de photos à afficher

4

Max de photo

Nombre maximum de photos à afficher

6

Affichage en mode aléatoire de la galerie

Mode d'affichage des photos aléatoire

Oui

Champs ACF nécessaires au fonctionnement de la galerie

Les différentes classes contenant les dispositions de grilles sont construites de la manière suivante : `.grid-catégorie-slug`

La catégorie correspond au nombre d'images que comportera la galerie (`.grid-6-x` est une classe pour une grille de 6 images). Ce nombre sera toujours compris entre les deux valeurs renseignées par l'utilisateur (nombre minimum et maximum de photos).

Quant au slug, il correspond à la version de la grille. Pour chaque catégorie, il existe un certain nombre de placements définis. J'ai repris le terme *slug* de WordPress, où il définit un identifiant unique à chaque publication.

Bien que l'utilisateur choisisse les deux paramètres librement, il est tout de même contraint sur certains points :

- il ne peut choisir qu'entre 1 et 8 images ;
- si le nombre d'images fourni est insuffisant, le nombre maximum sera égal au nombre de photos données.

L'utilisateur dispose également d'une option pour faire apparaître ces images de manière aléatoire dans la grille.

Enfin, un clic sur l'une des images fera apparaître la galerie entière, contenant l'ensemble des photos et des vidéos.

Voir fonction de création de galerie aléatoire

Voir fonction de création de la galerie complète

Escape Game en Mini Austin et 2CV

+ Ajouter



Escape Game en Mini Austin et 2CV

+ Ajouter



Deux disposition différentes d'une même page

Les apports du stage

Connaissances acquises

Grâce à ces projets, j'ai pu apprendre plus en profondeur le fonctionnement de WordPress, comment les différents éléments marchent les uns avec les autres, et comment ce CMS est organisé.

J'ai également pu consolider ma connaissance des langages de programmation PHP et JavaScript / jQuery. Ces projets m'ont également permis de découvrir le langage LESS.

Parallèlement, le projet sur Volant-Séminaire m'a obligé à utiliser Git de manière collaborative, et à mieux comprendre le fonctionnement de cet outil.

J'ai aussi pu mettre en pratique la méthode DRY (don't repeat yourself). Cette méthode est une manière de programmer qui consiste à éviter au maximum les redondances de code au sein d'un système. Ainsi, j'essayais toujours de fonctionner par "blocs" que je n'avais qu'à réutiliser

Enfin, j'ai plusieurs fois assisté à la mise en ligne d'un site web. Cela m'a permis de mieux comprendre l'ensemble du processus de déploiement d'un site web, et de comment cela fonctionnait.

Bilan humain

Grâce à ces 10 semaines de stage, j'ai eu l'occasion de découvrir comment fonctionnait le monde de l'entreprise, et de me rendre compte des connaissances qui étaient exigées dans le milieu professionnel.

Il m'a également permis de me rendre compte des difficultés qui peuvent survenir dans le déroulement d'un projet, comme par exemple le changement d'avis d'un client sur une fonctionnalité précise.

Enfin, ce stage m'a permis également de mieux comprendre l'utilité des différents métiers des personnes ayant collaboré avec nous durant ces projets (rédacteurs web, designer), et comment ces différentes personnes travaillent ensemble.

Poursuite d'études

Ce stage m'a permis de voir ce qu'était le monde du développement dans le milieu professionnel. Il m'a également conforté dans l'idée de poursuivre une carrière dans ce domaine, que ce soit dans le développement web ou dans un autre domaine.

Ce stage a également renforcé ma volonté de vouloir travailler dans une structure déjà existante en tant qu'employé. En effet, bien qu'être *freelance* possède énormément d'avantages, ce n'est pas une situation qui me conviendrait.

Les points bloquants comprennent notamment la nécessité de gérer plusieurs aspects en étant à son compte, tels que les tâches administratives et le démarchage de clients. De plus, travailler en *freelance* peut rendre plus complexe la gestion du salaire et sa variation, car les revenus peuvent varier considérablement d'un mois à l'autre, ce qui fait qu'un salaire fixe mensuel n'est jamais garanti.

Les annexes

Glossaire

CMS : système de gestion de contenu, un programme informatique permettant de créer un site internet.

WordPress : système de gestion de contenu gratuit.

Front-end : partie invisible d'une application pour l'utilisateur final.

Back-end : partie visible d'une application pour l'utilisateur final.

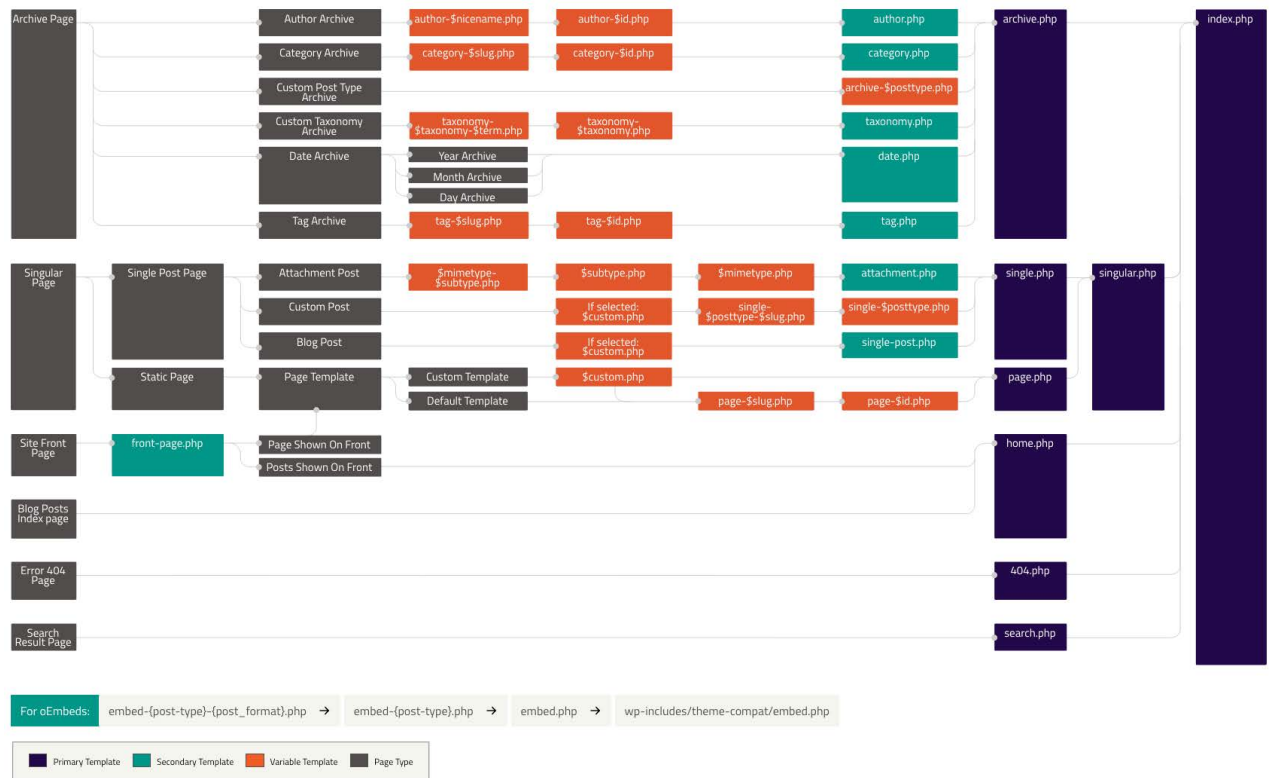
Custom Post Type : type de contenu personnalisé dans WordPress.

Git : logiciel de contrôle de version d'un logiciel.

Bitbucket / GitHub : service web d'hébergement et de gestion de développement de logiciels.

Plugin : logiciel conçu pour se greffer à un autre et lui apporter des fonctionnalités supplémentaires.

Illustrations



Le Template Hierarchy



LE XX/XX/2019

TITRE ARTICLE

Primi igitur omnium statuuntur Epigonus et Eusebius ob nomen gentilitatem oppressi. praediximus enim Montium sub ipso vivendi



LE XX/XX/2019

TITRE ARTICLE

Primi igitur omnium statuuntur Epigonus et Eusebius ob nomen gentilitatem oppressi. praediximus enim Montium sub ipso vivendi termino his vocabulis appellatos fabricarum culpasse tribunos ut adminicula futurae

LE XX/XX/2019

TITRE ARTICLE

Primi igitur omnium statuuntur Epigonus et Eusebius ob nomen gentilitatem oppressi. praediximus enim Montium sub ipso vivendi termino his vocabulis appellatos fabricarum culpasse tribunos ut adminicula futurae



LE XX/XX/2019

TITRE ARTICLE

Primi igitur omnium statuuntur Epigonus et Eusebius ob nomen gentilitatem oppressi. praediximus enim Montium sub ipso vivendi termino his vocabulis appellatos fabricarum culpasse tribunos ut adminicula futurae molitioni pollicitos.

LES DATES À NE PAS MANQUER

→ VOIR PLUS DE DATES

L'ACTUALITÉ EN DIRECT



LES CONCERTS

BO WEAVIL

POSTÉ LE XX/XX/2019

Primi igitur omnium statuuntur Epigonus et Eusebius ob nomen gentilitatem oppressi. praediximus enim Montium sub ipso vivendi termino his vocabulis appellatos fabricarum culpasse tribunos ut adminicula futurae molitioni pollicitos. Quam ob rem cave Catoni anteponas ne istum quidem ipsum, quem Apollo, ut ais, sapientissimum iudicavit: huius enim facta, illius dicta laudantur. De me autem, ut iam cum utroque vestrum loquar, sic habetote.

LIRE LA SUITE

→ VOIR TOUTES LES ACTUALITÉS

RESTEZ INFORMÉ

Colligendos quicquid super destinarentur vilitate et cognita.

Prénom et Nom

Email

ENVOYER

☒ En cochant cette case j'accepte les conditions d'utilisation des données par l'EMIJ, consultables ici.
Je peux à tout moment avoir accès à mes données. J'ai un droit de modification et de retrait de mes données
par simple demande écrite à emij@emij.fr.

ÉCOLE DE JAZZ ET MUSIQUES ACTUELLES

Colligendos quicquid super destinarentur vilitate et cognita.

Primi igitur omnium statuuntur Epigonus et Eusebius ob nomen gentilitatem oppressi. praediximus enim Montium sub ipso vivendi termino his vocabulis appellatos fabricarum culpasce tribunos ut adminicula futurae molitioni pollicitos. Quam ob rem cave Catoni anteponas ne istum quidem ipsum, quem Apollo, ut ais, sapientissimum iudicavit: huius enim facta, illius dicta laudantur. De me autem, ut iam cum utroque vestrum loquar, sic habetote.

LIRE LA SUITE



NOUS TROUVER



ENSEIGNEMENTS

L'ÉCOLE

L'ACTUALITÉ

LES ACTIONS CULTURELLES

CONTACTEZ-NOUS

NOUS SUIVRE SUR



Conditions générales d'utilisation du site et utilisation des données
Réalisé par Gribouilletnet©

EN PARTENARIAT AVEC



```

1 function wp_custom_post_instruments() {
2
3     $labels = array(
4         'name' => __('Instruments', 'Post Type General Name'),
5         'singular_name' => __('Instrument', 'Post Type Singular Name'),
6         'menu_name' => __('Instruments'),
7         'all_items' => __('Tous les instruments'),
8         'view_item' => __('Voir les catégories d\'instrument'),
9         'add_new_item' => __('Ajouter une nouvelle catégorie d\'instrument'),
10        'add_new' => __('Ajouter'),
11        'edit_item' => __('Editer la catégorie d\'instrument'),
12        'update_item' => __('Modifier la catégorie d\'instrument'),
13        'search_items' => __('Rechercher une catégorie d\'instrument'),
14        'not_found' => __('Non trouvée'),
15        'not_found_in_trash' => __('Non trouvée dans la corbeille'),
16    );
17
18    $args = array(
19        'label' => __('Catégorie d\'instrument'),
20        'description' => __('Toutes les catégories d\'instrument'),
21        'labels' => $labels,
22        'menu_icon' => 'dashicons-format-audio',
23        'supports' => array( 'title', 'editor', 'excerpt', 'author', 'thumbnail', 'comments', 'revisions', 'custom-fields', ),
24        'show_in_rest' => true,
25        'hierarchical' => false,
26        'public' => true,
27        'has_archive' => true,
28        'rewrite' => array( 'slug' => 'les-instruments'),
29    );
30
31    register_post_type('instruments', $args);
32
33 }

```

Création du Custom Post Type Instruments

```

1 // CSS
2 .parent {
3     background-color: white;
4 }
5 .parent .enfant {
6     color: black
7 }
8 .parent .enfant:hover {
9     color: red;
10 }

```

Un exemple en CSS

```

1 // LESS
2 .parent {
3     background-color: white;
4     .enfant {
5         color: black;
6         &:hover {
7             color: red;
8         }
9     }
10 }

```

Un exemple en LESS



```
1  @light-purple: #D9BFFF;  
2  @purple:      #B28FAC;  
3  @dark_purple: #651F59;  
4  @pink:        #FF6971;  
5  
6  @title-weight: 600;  
7  @title-letter-spacing: 1px;
```

Variables LESS



```
1  .clip-path() {  
2      clip-path: polygon( 0 11%, 100% 0, 100% 100%, 0 100% )  
3  }  
4  
5  .figure-background(@alpha) {  
6      background-color: fade(@light-purple, @alpha)  
7  }
```

Mixins LESS



```
1  @import url("import/variables.less");  
2  @import url("import/mixins.less");  
3  @import url("import/animate.less");  
4  
5  @import url("import/header.less");  
6  @import url("import/carousel.less");  
7  @import url("import/footer.less");
```

Import de fichiers LESS


```

1 $post_id = get_the_ID();
2 $query = new WP_Query( array( 'post_type' => array( 'instruments', 'ateliers' ), 'p' => $post_id ) );
3
4 while ($query->have_posts()) {
5     $query->the_post();
6
7     ?>
8     <h2>Jour(s) et horaires du cours</h2>
9     <div class="jours-cours">
10    <?php
11
12    if( have_rows('choix-horaire') ) {
13        while( have_rows('choix-horaire') ) {
14
15            the_row();
16
17            $presence_jours[] = get_sub_field('presence-jours', $post_id );
18            $presence_heure_d[] = get_sub_field('presence-heures-debut', $post_id);
19            $presence_heure_f[] = get_sub_field('presence-heures-fin', $post_id);
20        }
21
22        $semaine = ["Lundi", "Mardi", "Mercredi", "Jeudi", "Vendredi", "Samedi"];
23        $cpt = 0;
24
25        for($i = 0; $i < count($semaine); $i++) {
26            if(in_array($semaine[$i], $presence_jours)) { ?>
27                <div class='present'>
28                    <?=$semaine[$i]; ?>
29                    <span class="horaires"><?=$presence_heure_d[$cpt] . "h à " . $presence_heure_f[$cpt] . "h"; $cpt++; ?></span>
30                </div>
31                <?php } else { ?>
32                    <div><?=$semaine[$i]; ?></div>
33                <?php }
34            }
35        } else {
36            echo "<p>Aucun cours de programmé actuellement<p>";
37        }
38    }
39    wp_reset_query(); ?>

```

Traitement des données ACF pour les horaires

```

1 <div class="professeurs">
2     <?php
3
4     $query = new WP_Query( array( 'post_type' => 'professeurs' ) );
5
6     global $post;
7     $titre_article = $post->post_name;
8
9     while ($query->have_posts()) {
10         $query->the_post();
11
12         if(have_rows('cours')) {
13             while( have_rows('cours') ) {
14                 $nom_categorie = the_row();
15                 if($nom_categorie == $titre_article) {
16                     ?>
17                     <div class="professeur">
18                         <?php the_post_thumbnail(); ?>
19                         <p><?php the_title(); ?></p>
20                         <p><?php the_field('instruments'); ?></p>
21                     </div>
22                     <?php
23                 }
24             }
25         }
26     }
27
28     echo "</div>";
29     wp_reset_query();

```

Traitement des données ACF pour les professeurs

```

1 // AJOUT
2 $(this).click( function(){
3     $('#form-devis').prepend('<div id="titre_activite-' + data_activiteID + '-' + cpt_activites +
4     ' " class="devis-item"><div class="row">' + html_item +
5     '<button class="d-flex justify-content-center align-items-center delete-activite button-circle button-delete"><i class="icon-delete"></i></button></div>' +
6     form( data_activiteID, cpt_activites ) + '</div>');
7
8     $('#titre_activite-' + data_activiteID + '-' + cpt_activites + ' .icon-plus').remove();
9     $('#titre_activite-' + data_activiteID + '-' + cpt_activites + ' .devis-item-image').addClass('col-3 col-md-2');
10    $('#titre_activite-' + data_activiteID + '-' + cpt_activites + ' .devis-item-content').addClass('col-8 col-md-9');
11
12    var DivnewTransform = 0 + 'px';
13    $('#container-list-items-shutter').css( 'transform', 'translateY(' + DivnewTransform + ')').removeClass('open');
14
15    cpt_activites++;
16
17    $('.main-page-devis').css('height', calcul_height() );
18 });
19
20
21 // SUPPRESSION
22 $(document).on('click', '.delete-activite', function() {
23     $(this).closest('.devis-item').remove();
24
25     $('.main-page-devis').css('height', calcul_height());
26
27 });

```

Ajout / Supression d'éléments du formulaire

```

1  if(isset($_POST['email'])) {
2
3      // Sécurisation des données
4      foreach($_POST as $post) {
5          $post = secure_input($post);
6      }
7
8      $ids = array();
9      $errors_log = array();
10     $inputs_errors_name = array();
11     $skip_verif = false;
12
13     // Récupérer ID des différentes activités
14     foreach($_POST as $index => $post) {
15         if(preg_match('/nombre_personnes-/', $index) && !isset($_POST['submit_delete-' . substr($index, 17 - strlen($index))])) {
16             $ids[] = substr($index, 17 - strlen($index));
17         }
18     }
19
20     // Si tous Les posts supprimés, revient à celui de base
21     if(count($ids) == 0 && isset($_POST['id_principale'])) {
22         $ids[] = $_POST['id_principale'];
23         $skip_verif = true;
24     }
25
26
27     if(strlen($_POST['phone']) < 10 ) {
28         if(strlen($_POST['phone']) < 1 ) {
29             $errors_log[] = "Veuillez renseigner un numéro de téléphone";
30             $inputs_errors_name[] = 'phone';
31         } else {
32             $errors_log[] = "Le numéro de téléphone renseigné est invalide";
33             $inputs_errors_name[] = 'phone';
34         }
35     }

```

Extraits du script de gestion d'erreurs - 1 / 2

```

1  $inputs = ['nombre_personnes', 'date_activite', 'lieu_seminaire', 'horaires_debut', 'horaires_fin'];
2  $errors = [
3      'Veuillez renseigner un nombre de personne',
4      'Veuillez renseigner une date pour votre activité',
5      'Veuillez renseigner un lieu pour votre séminaire',
6      'Veuillez renseigner un horaire de début correct',
7      'Veuillez renseigner un horaire de fin correct'
8  ];
9
10 if(!$skip_verif) {
11     foreach($ids as $id) {
12         for($i = 0; $i < count($inputs); $i++) {
13             if(strlen($_POST[$inputs[$i] . '-' . $id]) < 1) {
14                 $errors_log[] = $errors[$i];
15                 $inputs_errors_name[] = $inputs[$i] . '-' . $id;
16             }
17         }
18         if(strlen($_POST[$inputs[3] . '-' . $id]) > 0 && strlen($_POST[$inputs[4] . '-' . $id]) > 0) {
19             if(!compare_hours($_POST[$inputs[3] . '-' . $id], $_POST[$inputs[4] . '-' . $id])) {
20                 $errors_log[] = "Les heures de début et de fin de séminaire ne sont pas correctes";
21                 $inputs_errors_name[] = $inputs[3] . '-' . $id;
22                 $inputs_errors_name[] = $inputs[4] . '-' . $id;
23             }
24         }
25     }
26 }
27
28 if(in_array('Veuillez renseigner un horaire de début correct', $errors_log) && in_array('Veuillez renseigner un horaire de fin correct', $errors_log)){
29     array_pop($errors_log);
30     array_pop($errors_log);
31     array_pop($inputs_errors_name);
32     $errors_log[] = "Veuillez renseigner des horaires corrects";
33 }
34
35 $_SESSION = $_POST;
36 $_SESSION['erreurs'] = $errors_log;
37 $_SESSION['input_erreurs'] = $inputs_errors_name;
38 $_SESSION['ids'] = $ids;
39
40 if(count($errors_log) == 0 && have_a_slug($_SESSION)) {
41     //header("Location: " . $urlTemplate . "/inc/functions/function-submit-devis.php");
42     header("Location: https://volant-seminaire.gribdev.net/wp-content/themes/volant-seminaire/inc/functions/function-submit-devis.php");
43     exit();
44 }
45
46 }
47
48 function have_a_slug($session, $index = 0) {
49     if($index < count($session['ids']))
50         return str_contains($session['ids'][$index], '-');
51
52     return str_contains($session['ids'][$index], '-') && have_a_slug($session, $index + 1);
53 }

```

Extraits du script de gestion d'erreurs - 2 / 2

```

1 function set_cookies($reference) {
2
3     $cookie_live_time = 10;
4
5     $personnal_datas_index = ['firstname', 'lastname', 'social_reason', 'phone', 'email', 'message'];
6     $activite_datas_index = ['titre_activite', 'nombre_personnes', 'lieu_seminaire', 'date_activite', 'horaires_debut', 'horaires_fin'];
7
8     foreach($personnal_datas_index as $personnal_data_index) {
9         setcookie($personnal_data_index, $_SESSION[$personnal_data_index], time() + $cookie_live_time, '/');
10    }
11
12    foreach($_SESSION['ids'] as $id) {
13        foreach($activite_datas_index as $activite_data_index) {
14            setcookie($activite_data_index . '-' . $id, $_SESSION[$activite_data_index . '-' . $id], time() + $cookie_live_time, '/');
15        }
16    }
17    $ids = '';
18    foreach($_SESSION['ids'] as $id) {
19        $ids .= $id . '/';
20    }
21
22    setcookie('ids', substr($ids, 0, -1), time() + $cookie_live_time, '/');
23    setcookie('reference', $reference, time() + $cookie_live_time, '/');
24
25    if( isset($_SESSION['lieu_seminaire_hotel']) )
26        setcookie('lieu_seminaire_hotel', $_SESSION['lieu_seminaire_hotel'], time() + $cookie_live_time, '/');
27 }

```

Création des cookies

```

1 <table style="width: 100%;">
2     <tbody>
3         <tr>
4             <td align="center">
5                 <table style="background-color: #333333; height: 75px; min-width: 800px; max-width: 1000px;">
6                     <tbody>
7                         <tr>
8                             <td style="text-align: center">
9                                 <img src="" . $image_url . '/email/volant-seminaire-logo.png' alt="volant-seminaire-logo">
10                            </td>
11                        </tr>
12                    </tbody>
13                </table>
14            </td>
15        </tr>
16    </tbody>
17 </table>

```

Structure simplifiée du mail utilisant les tableaux HTML

```

1 function create_reference($user_datas) {
2     $reference = date('ymd');
3     $reference .= strtoupper( substr($user_datas['firstname'], 0, 1) );
4     $reference .= strtoupper( substr($user_datas['lastname'], 0, 1) );
5     $reference .= '_';
6
7     $alphabet="abcdefghijklmnopqrstuvwxyz";
8     $reference .= strtoupper($alphabet[rand(0,25)]);
9     $reference .= strtoupper($alphabet[rand(0,25)]);
10
11     return $reference;
12 }

```

Fonction de création d'une référence unique pour le devis

```

1 if(isset($_COOKIE['reference'])) {
2     if(!is_devis_unique($_COOKIE['reference'])) {
3
4         $args = array(
5             'post_title' => $_COOKIE['reference'],
6             'post_content' => generate_content(),
7             'post_status' => 'publish',
8             'post_author' => 1,
9             'post_type' => 'devis',
10        );
11
12        wp_insert_post($args);
13        unset($_COOKIE['reference']);
14    }
15 }
16
17 function generate_content() {
18     $message = '
19     <h2>Référence du devis ['. $_COOKIE['reference'] . ']</h2>
20     <p>Demande par <strong> . ucfirst(strtolower($_COOKIE['firstname'])) . ' ' . ucfirst(strtolower($_COOKIE['lastname'])) . '</strong></p>
21     <p>Fiche contact : <br/>Email : <strong>&lsquo; . $_COOKIE['email'] . '&rsquo;</strong><br/>Téléphone : <strong> . $_COOKIE['phone'] . '</strong><p>';
22
23     $message .= '<p><i> . $_COOKIE['message'] . '</i></p>';
24
25     $activite_datas_index = ['titre_activite', 'nombre_personnes', 'lieu_seminaire', 'date_activite', 'horaires_debut', 'horaires_fin'];
26
27     foreach(explode('/', $_COOKIE['ids']) as $id) {
28
29         $message .= '<h3> . $_COOKIE[$activite_datas_index[0]] . ' . $id] . '</h3>';
30         $message .= '<p>Nombre de personnes : <strong> . $_COOKIE[$activite_datas_index[1]] . ' . $id] . ' personne(s)</strong>
31         Date, heures, et lieu souhaités :<strong> . $_COOKIE[$activite_datas_index[3]] . ' . $id] .
32         ', de ' . $_COOKIE[$activite_datas_index[4]] . ' . $id] .
33         ' à ' . $_COOKIE[$activite_datas_index[5]] . ' . $id] .
34         ', à ' . ucfirst(strtolower($_COOKIE[$activite_datas_index[2]] . ' . $id)) . ' . $id)) . '</strong></p>';
35         $message .= '<hr>';
36
37         if( isset( $_COOKIE['lieu_seminaire_hotel'] ) )
38             $message .= "<p>Préstation hôtelière à : <strong> . ucfirst(strtolower($_COOKIE['lieu_seminaire_hotel'])) . '</strong></p>";
39     }
40
41     return $message;
42 }
43 }

```

Insertion d'un Custom Post Type dans WordPress

```

1  function get_galerie() {
2
3      $images      = get_field('galerie'      );
4      $n_min       = get_field('min_de_photo' );
5      $n_max       = get_field('max_de_photo' );
6      $is_random   = get_field('mode_aleatoire');
7
8
9      if(!$images) return;
10
11     if($n_max > count($images)) $n_max = count($images);
12     if($n_min > count($images))
13         $n_min = ( count($images) > 1 ) ? $n_max - 1 : $n_max;
14
15     if($n_min < 1 || $n_min > 8) $n_min = 4;
16     if($n_max < 1 || $n_max > 8) $n_max = 6;
17
18     if($n_min > $n_max) [$n_min, $n_max] = [$n_max, $n_min];
19
20     $intervalle = $n_max - $n_min + 1;
21     $categorie  = rand($n_min, $n_max);
22
23     if(in_array($categorie, [1, 2, 8])) $max_slug = 1;
24     if(in_array($categorie, [3, 7] )) $max_slug = 2;
25     if(in_array($categorie, [4, 5, 6])) $max_slug = 5;
26
27     $slug = rand(1, $max_slug);
28     $class = 'grid-' . $categorie . '-' . $slug;
29
30     if($is_random)
31         shuffle($images);
32
33
34     $galerie = "<ul id='galerie-medium' class='grid-galerie $class'>";
35
36     for( $i = 1; $i <= $categorie; $i++ ) {
37         $size = 'full';
38         $image_id = $images[$i - 1]['id'];
39
40         $galerie .= "<li class='item-grid item-$i'>";
41         $galerie .= wp_get_attachment_image( $image_id, $size );
42         $galerie .= "</li>";
43     }
44
45     return $galerie .= "</ul>";
46 }
47
48 function the_galerie() {
49     echo get_galerie();
50 }
51
52 function get_the_galerie() {
53     return get_galerie();
54 }

```

Fonction de création de la galerie aléatoire


```

1 function get_full_galerie() {
2     $images = get_field('galerie');
3     if(count($images) > 18)
4         $images = array_slice($images, 0, 18);
5
6     $videos = array();
7
8     if( have_rows('galerie_videos') ) {
9         while( have_rows('galerie_videos') ) {
10             the_row();
11             $videos[] = get_sub_field('video');
12         }
13     }
14
15     $total_elements = count($images) + count($videos);
16
17     if($total_elements > 20) $total_elements = 20;
18
19     $galerie = "<ul class='grid-full-galerie grid-full-galerie-$total_elements'>";
20
21     $videos_positions = [];
22
23     if(count($videos) > 0) {
24         if(count($videos) == 1) $videos_positions = [1];
25         if(count($videos) == 2) $videos_positions = [1, 7];
26
27         for($i = 0; $i < count($videos); $i++) {
28             $galerie .= "<li class='item-grid item-" . $videos_positions[$i] . "'>" . $videos[$i] . "</li>";
29         }
30     }
31
32     $gap = 0;
33     for($i = 1; $i <= $total_elements; $i++) {
34
35         if(in_array($i, $videos_positions)) {
36             $gap++;
37             continue;
38         }
39
40         $galerie .= "<li class='item-grid item-$i'>";
41
42         $size = 'full';
43         $image_id = $images[$i - 1 - $gap]['id'];
44         $galerie .= wp_get_attachment_image( $image_id, $size );
45
46         $galerie .= "</li>";
47     }
48
49     $galerie .= "</ul>";
50     return $galerie;
51 }
52
53 function the_full_galerie() {
54     echo get_full_galerie();
55 }
56
57 function get_the_full_galerie() {
58     return get_full_galerie();
59 }

```

Fonction de création de la galerie complète contenant des vidéos