



L'association PAUL **Green**

2022

Dossier PROJET





Table des matières



Table des matières	2
Avant-Propos	4
Remerciements	5
Liste des compétences couvertes par le référentiel.....	5
Résumé de projet.....	6
Environnement de travail	8
Les outils utilisés.....	8
Cahier des charges.....	11
Analyse de l'existant	11
Définition de projet et Cahier des charges.....	11
Objectifs du site que je me suis fixés pour l'examen.....	11
Périmètre du projet.....	12
Graphisme et ergonomie	12
Veille informationnelle.....	13
Charte graphique	14
Wireframe et Maquettage	16
Maquetter une application	16
Arborescence du site	17
Arborescence détaillée	18
Réalisation du site	19
Planification du projet	19
Définition des rôles	20
MCD / MLD / UML	21
MCD	21
UML.....	22
MLD.....	23
Use Case Diagramme	24
Diagrammes d'activités.....	25
Développement Back-End.....	27

Création de la base du squelette et de la base du projet	27
Construction de la base de données	28
Création des entités	29
Sécurisation et tests.....	39
<i>Front-End.....</i>	46
Mise en place du Framework CSS Bootstrap.....	46
Mise en place des icons	47
Mise en place des polices d'écriture.....	47
Mise en place du CSS.....	47
Réaliser une interface web statique et dynamique.....	49
Exemples de codes.....	51
Gestions de photos	51
Affichage du Bureau de l'association :	59
<i>Mise en Production.....</i>	62
Hébergement.....	62
Optimisation des performances et Référencement	63
Cookies/RGPD	66
<i>Conclusion</i>	70
<i>Annexe.....</i>	72



Avant-Propos



Ce dossier professionnel a été réalisé dans le cadre d'un projet de remise à niveau professionnel suite à un licenciement économique. Informaticien de métier, lors de mes premiers entretiens avec ma conseillère Pôle emploi j'ai évoqué l'idée de suivre une formation en développement informatique vers un secteur porteur. C'est ainsi qu'elle a pu me proposer de suivre une formation de Développeur Web et Web mobile en ayant l'opportunité d'obtenir un financement auprès de Pôle emploi. J'ai donc pris contact avec l'organisme de formation ARINFO de Saint Nazaire.

J'ai tout de suite été séduit par le parcours proposé par l'organisme, le concept « Learning by doing » me permettant de toujours découvrir et apprendre, et assouvir la curiosité qui me caractérise.

J'ai également été attiré **par la variété des emplois et des thèmes** que regroupe le métier de développeur Web, et par la **possibilité d'évolution** qu'il propose dans une carrière.

En effet, je suis informaticien de métier d'abord dans un premier temps en tant que développeur informatique exercé pendant plus de 20 ans puis en tant que technicien de maintenance en informatique pendant 9 ans.

J'ai choisi cette formation car j'ai voulu revenir à mon métier de base et actualiser mes compétences en matière de développement en me familiarisant avec les outils du web pour concevoir des sites et découvrir de nouveaux langages de programmation web afin d'être plus en phase avec la réalité du marché du travail.

Cette formation a duré 6 mois, du 2 novembre 2021 au 30 avril 2022.

Remerciements

Je tiens à remercier Nicolas GICQUEL notre formateur, pour la qualité de sa formation, sa disponibilité, sa pédagogie. Elodie BOINET directrice du centre ARINFO de Saint-Nazaire pour son soutien et ses conseils pour réaliser cette formation, ainsi que Magali VINCE qui m'a donné des conseils quant à la présentation et la rédaction de mon CV.

Je remercie également ma conseillère Pôle Emploi pour m'avoir soumis cette proposition et suivi mon dossier pour l'obtention de son financement

Je remercie également mes camarades de promo, Yann, Arlinda, Sophie et Léa pour l'entraide bienveillante dont ils ont tous fait preuve et ma femme Mireille, qui a su me soutenir et m'encourager dans les moments de doutes et de remises en question.

Liste des compétences couvertes par le référentiel

1. Maquetter une application
2. Réaliser une interface utilisateur web statique et adaptable
3. Développer une interface utilisateur web dynamique
4. Réaliser une interface utilisateur avec une solution de gestion de contenu ou e-commerce
5. Créer une base de données
6. Développer les composants d'accès aux données
7. Développer la partie back-end d'une application web ou web mobile
8. Élaborer et mettre en œuvre des composants dans une application de gestion de contenu ou e-commerce



Résumé de projet



L'idée de ce projet est née après une discussion avec ma fille étudiante en master 1 en environnement à l'université Paul Valéry de Montpellier 3 qui m'évoque une association d'étudiants au sein de sa faculté

Cette association a pour but de présenter des sujets environnementaux, écologiques auprès de la communauté universitaire mais aussi de présenter le Master, sur les 2 années, Gestion de L'Environnement (GE), ainsi que les droits des étudiants,

Elle expose notamment l'idée du jardin partagé mis en place par l'association mais qui a du mal à se développer de façon pérenne, en évoquant peut être le fait que l'association n'ait pas encore acquis suffisamment de visibilité et de notoriété. D'où l'idée sous-jacente du bureau de l'association de pouvoir développer la communication à travers un site d'information sur le master Gestion de l'environnement et sur les différents projets écologiques menés par l'association et les étudiants de Master dans le cadre de leur cursus.

Sensible aux problématiques environnementales, et soucieux de me rendre utile à un groupe de jeunes étudiants, l'idée m'a vite séduit et j'ai donc proposé la création d'un site au bureau administratif pour mettre en valeur leur association et leurs projets.

Cette démarche s'inscrit aussi dans l'idée de moderniser l'image de l'association, d'attirer de nouveaux adhérents, d'augmenter la visibilité du Master et de pérenniser les supports dans le temps.

Afin de pouvoir échanger sur les différents aspects du projet et sur sa faisabilité, nous avons pratiqué des échanges via Zoom.

Suite à l'analyse de leurs besoins définis en équipe de leur côté, j'ai pu leur proposer un cahier des charges, en exposant les contraintes techniques et les limites du projet, puis par la réalisation de la maquette graphique (des wireframes à la maquette HTML/CSS) et enfin tout le développement du site avec le déploiement des composants d'accès à la base de données et l'ajout au fur et à mesure des différentes fonctionnalités qui permettent aujourd'hui aux clients étudiants d'être quasiment 100% autonomes dans la gestion de leur site : ajout, modification et suppression d'actualités, d'offres d'emploi, d'outils numériques et la gestion des utilisateurs par les administrateurs étudiants désignés.

Au niveau de l'environnement et des technologies utilisés, j'ai développé mon projet avec l'éditeur de code « Visual studio code ». Le langage de programmation est le PHP et JavaScript sous le framework Symfony 5.4, avec le support du framework Bootstrap pour le front-end. Je me sers également de l'ORM Doctrine, inclus dans Symfony pour la création et la gestion de la base de données. La plateforme de développement est WampServeur avec phpMyAdmin pour l'administration Web des bases MySQL.

Ce projet fut enrichissant, car il m'a permis de m'adapter en fonction des besoins évolutifs de mes interlocuteurs.

Le projet et son déroulement furent très appréciés par le bureau de l'association des étudiants sachant que ce site pour répondre complètement aux besoins de l'association qui ne cessent d'évoluer, devra être développé davantage, ce que je pourrai prétendre une fois ma formation finie afin de leur proposer un produit plus complet.



Les outils utilisés

Au cours de son développement, le projet a nécessité l'utilisation de nombreux outils.

Maquettage et diagrammes

 draw.io	L'application google draw.io , pour construire les diagrammes d'activité, UML, ainsi que les wireframes du site
	Looping , un logiciel gratuit et libre d'utilisation, qui a été conçu par l'Université Toulouse III, et qui permet notamment de construire des MCD (Modèles Conceptuels de Données) et des MLD (Modèles Logiques de Données), conçus à partir de la méthode Merise

Back-end

 Symfony	Symfony est un framework PHP open source avec une architecture MVC. C'est l'un des framework les plus populaires parmi la communauté des développeurs open source. Il est utilisé pour construire des applications web complexes et performantes. Un framework a pour but de simplifier et d'accélérer le processus de développement. Ce framework PHP présente l'avantage d'être puissant et robuste.
	Twig est un moteur de templates pour le langage de programmation PHP, utilisé par défaut par le framework Symfony.
	PHP est le langage de programmation associé à Symfony

	<p>Composer me permet de déclarer et d'installer les bibliothèques dont le projet a besoin.</p>
	<p>La plate-forme de développement web Wampserver, permet de simuler localement le fonctionnement du site internet.</p>
	<p>En ce qui concerne la base de données j'ai utilisé MySQL (My Structured Query Language) qui est un gestionnaire Open-Source couplé à PHPMyAdmin une interface simple et facile d'accès.</p>
	<p>Pour la mise en ligne du site, j'ai utilisé WinSCP, un client SFTP (SSH File Transfer Protocol).</p>
	<p>cPanel, un panneau de configuration permettant de gérer l'hébergement d'un site web.</p>

Front-End

	<p>J'ai utilisé le framework HTML Bootstrap lors de la conception de la maquette du site en HTML/CSS.</p> <p>Bootstrap est une collection d'outils utiles à la création du design de sites et d'applications web. C'est un ensemble qui contient des codes HTML et CSS, des formulaires, boutons, outils de navigation et autres éléments interactifs et également une bibliothèque d'icône, ainsi que des extensions JavaScript en option. Son système de Grid est particulièrement efficace.</p>
	<p>Le préprocesseur CSS (Cascading Style Sheet) Sass, m'a permis d'organiser et de construire mon CSS plus aisément, de faciliter les mises à jour en cas de modification du code</p>

	<p>Pour les effets dynamiques du site j'ai utilisé le langage JavaScript et sa bibliothèque « jQuery ».</p>
	<p>Ajax pour Asynchronous JavaScript and XML, correspond à un groupe de méthodes et de moyens visant à permettre d'établir une communication asynchrone entre le navigateur et le serveur. Ajax permet d'effectuer des modifications parcellaires sur une page web, sans recharger l'ensemble de la page internet.</p>

Logiciels

	<p>Concernant mon environnement j'ai développé sur l'IDE (Integrated Development Environment ou Environnement de Développement) Visual Studio Code, qui permet de personnaliser facilement son espace de travail, et les nombreuses extensions que l'on peut lui adjoindre de rendre plus facile le travail de codage.</p> <p>Coté terminal j'ai utilisé powershell qui est intégré à Visual Studio Code, couplé à la gestion de dépendance composer il m'a permis la construction de mon projet, le téléchargement des différents bundles ainsi que d'effectuer les mises à jour nécessaire</p>
	<p>J'ai utilisé la plateforme d'hébergement Github, afin de faire des sauvegardes en cas de mauvaise manipulation ou de problèmes techniques et de son interface Github Desktop</p>
	<p>Afin de créer certains designs, de redimensionner certains contenus multimédias, ou de produire les différentes icônes, j'ai utilisé le logiciel libre et gratuit de retouche d'images Gimp</p>



Cahier des charges



Analyse de l'existant

Il s'agit d'un nouveau projet de création de site pour l'association Paul Green. Un site existait auparavant mais était devenu obsolète, sans aucune maintenance, ni suivi et les codes administrateurs étaient inconnus depuis plusieurs années par les étudiants.

Définition de projet et Cahier des charges

Après plusieurs échanges en visioconférence et par mail, les clients ont mis en évidence les éléments importants que le projet final devait respecter (annexe 1). Étant donné l'ampleur des besoins exprimés, j'ai pu donc préparer un cahier des charges non exhaustif où j'ai repris les principaux points dans l'objectif de l'examen, sachant que le site fera l'objet d'une continuité après la formation pour répondre au mieux aux besoins des étudiants.

Objectifs du site que je me suis fixés pour l'examen

Rendre les informations de l'association et du master GE accessibles et permettre la compréhension des actions menées et les projets collectifs du Master 1 et 2 par le plus grand nombre, que ce soit les étudiants du Master ou tout autre étudiant

Permettre aux adhérents d'accéder à un espace pour y trouver des informations précises tout en limitant l'accès selon leurs statuts (membres du bureau, adhérents simples membres du Master GE et pour les autres étudiants du campus non membres du Master GE.)

C'est un site à vocation informative (pallier au manque de visibilité de l'association) mais aussi contributive (apporter des idées, prendre contact, consulter des documents)

Je souhaite mettre en avant l'association, la présentation du master GE et les actions menées par l'association. Puis y ajouter une rubrique contact pour joindre l'association et un espace sur les partenaires de l'association.

Enfin, un espace de promotion des actions locales pour l'environnement sur le bassin montPELLIÉRAN.

Les visiteurs visés sont principalement des étudiants de l'université Paul Valéry Montpellier 3 en master GE (actions, offres de stage, anciens travaux) mais aussi à tout autre étudiant sensible aux problématiques de l'environnement et de l'écologie mais aussi de façon plus large aux futurs étudiants intéressés par le Master GE.

Périmètre du projet

Mon site ne doit pas être multilingue.

Le site internet doit être évolutif, et pouvoir intégrer les différents réseaux sociaux et médias au fur et à mesure que l'association développera sa communication.

Le contenu doit pouvoir être modifié et ajouté par les membres du bureau de l'association, et la partie back-end se devait donc d'être accessible et facilitée pour des néophytes en développement et gestion de sites web.

D'un point de vue général, le site devra être moderne, pour atteindre une cible de jeunes étudiants mais aussi pour montrer la dynamique du secteur de l'environnement.

Graphisme et ergonomie

Aucune charte graphique, tout était à définir !

Le logo existant allait servir de guide pour concevoir le design, qui se devait d'être accueillant et épuré, et rappeler les activités du site.



Veille informationnelle



Afin d'établir une charte graphique du site web et de correspondre au mieux à mes attentes et à celles de mes clients, j'ai effectué différentes recherches afin de trouver des renseignements sur divers sites que ce soit associatif, environnemental, ou universitaire. Mes recherches m'ont beaucoup aidé à trouver des solutions design et efficaces pour répondre aux besoins de l'association, mais aussi à approfondir le contenu pour être au plus près des requêtes des membres du bureau de l'association.

Voici les différents sites qui m'ont beaucoup aidé pour le design et les fonctionnalités que les sites ont tendances à utiliser :

Des associations d'étudiants sur l'environnement :

<https://www.helloasso.com/associations/association-des-etudiants-en-droit-de-l-environnement-aede/evenements/tote-bags-100-recycles-aede>

<https://artsetmetiers.fr/index.php/fr/actualites/environnement-quand-les-etudiants-sengagent>

<https://www.univ-nantes.fr/sepanouir-sur-les-campus/vie-associative-et-engagement/les-associations-etudiantes-environnement>

<https://le-reses.org/>

Des sites de présentation du master GE

<https://master-act-alumni.jimdofree.com/>

<https://esigat.wordpress.com/stages/>

<https://www.etudiants-mediation-scientifique.com/association-s-e-n-s-symbiose-entre-nature-et-societe/>

<https://anestaps.org/nos-partenaires/>

<https://www.master-gedd-unice.fr/Association-GAIA.htm>

Des sites d'associations environnementales

<http://www.natur-action.org/partenaires/>

<https://fne.asso.fr/>

Autres sites :

<https://www.cite-sciences.fr/fr/au-programme/lieux-ressources/bibliotheque/quoi-faire/evenements/environnement-les-initiatives-etudiantes/>



Charte graphique



Pour le choix de la typographie :

L'étude du logotype m'a servi de référence pour choisir la typographie du site pour le texte et pour les titres en me rapprochant des polices utilisées par les étudiants à l'origine de la refonte du logo.

abcdefghijklmnopqrstuvwxyz
ABCDEFGHIJKLMNOPQRSTUVWXYZ
0123456789 (!#\$%&/.|*`@',?:;)

Penultimate
The spirit is willing but the flesh is weak
SCHADENFREUDE
3964 Elm Street and 1370 Rt. 21
The left hand does not know what the right hand is doing.
mail@example.com

Pour le texte :

Trébuchet



Pour les titres :

Gelasio

Parisienne

abcdefghijklmnopqrstuvwxyz
ABCDEFGHIJKLMNOPQRSTUVWXYZ
0123456789 (!#\$%&/.|*`@',?:;)

Penultimate
The spirit is willing but the flesh is weak
SCHADENFREUDE
3964 Elm Street and 1370 Rt. 21
The left hand does not know what the right hand is doing.
mail@example.com

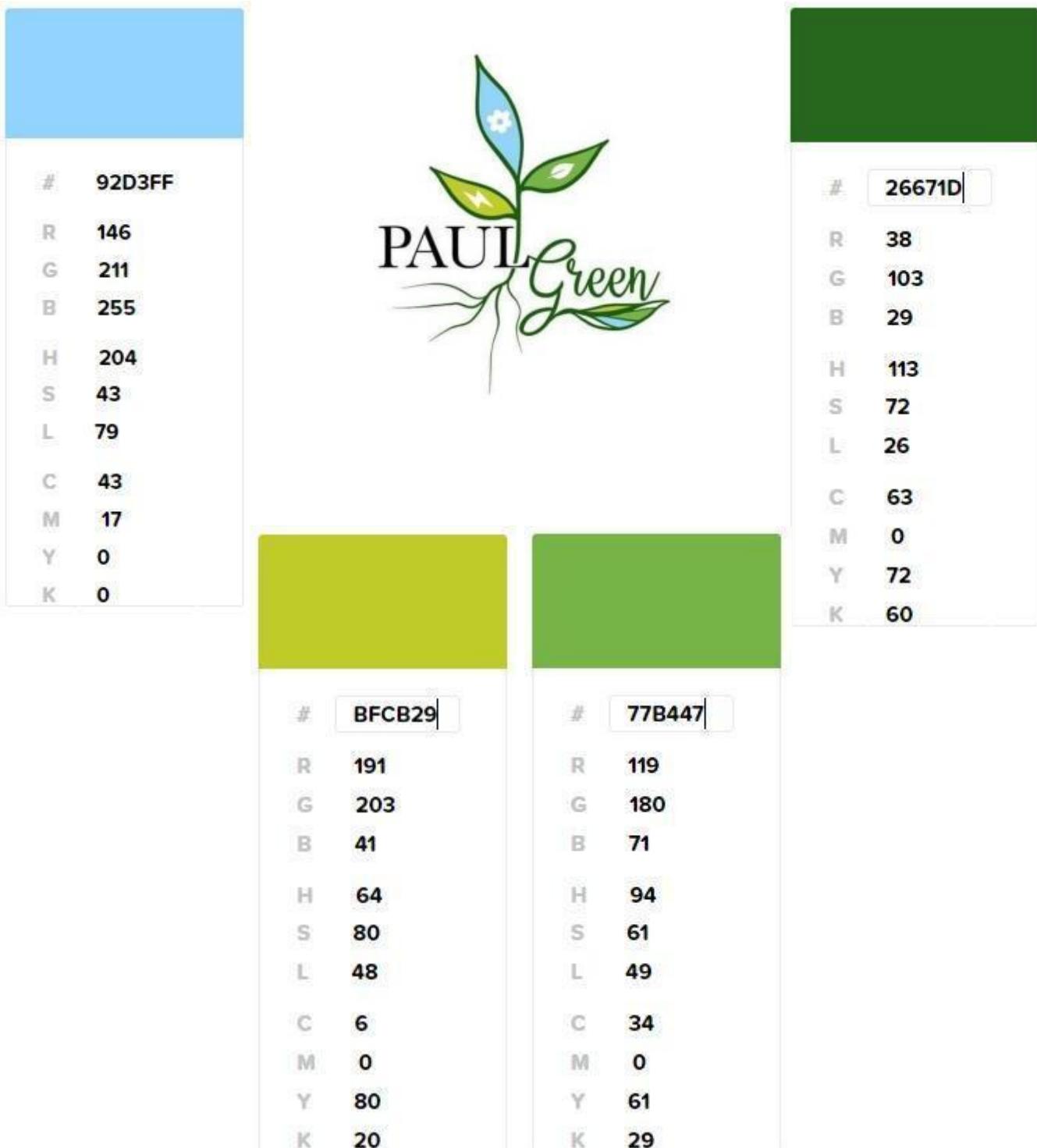
abcdefghijklmnopqrstuvwxyz
ABCDEFGHIJKLMNOPQRSTUVWXYZ
0123456789 (!#\$%&/.|*`@',?:;)

Penultimate
The spirit is willing but the flesh is weak
SCHADENFREUDE
3964 Elm Street and 1370 Rt. 21
The left hand does not know what the right hand is doing.
mail@example.com

Pour le choix des couleurs :

Le chemin directeur pour le choix des couleurs est également parti du découpage du logo fourni par l'association.

Un dégradé de couleur de couleur peut être effectué en partant de ces couleurs principales rappelant les thématiques de l'environnement (nature, eau)





Wireframe et Maquettage

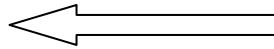


Maquetter une application

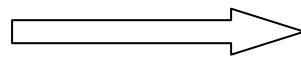
J'ai utilisé l'application « diagrams.net ». Elle est une solution en ligne, gratuite, qui permet la création de schémas, de diagrammes, d'organigrammes. Ici, elle me sert à la génération des premiers wireframes, ceux de la page d'accueil, aussi bien en mode PC qu'en mode mobile ou responsive.

The wireframe shows a header with three logo variants (Logo 1, Logo 2, Logo 3) and a menu bar. Below is a large image placeholder labeled "Présentation de l'association". A sidebar on the left contains sections for "Les dernières du Journal de l'environnement", "Les dernières Actions de L'association", "Les dernières Actions du Master", "Les dernières Sorties Pédagogiques", and "Nos Partenaires". At the bottom, there's a contact form for "Nous Contactez..." and a footer with "Association...", "Localisation", and "Web" sections.

À gauche nous avons le wireframe de la page d'accueil.



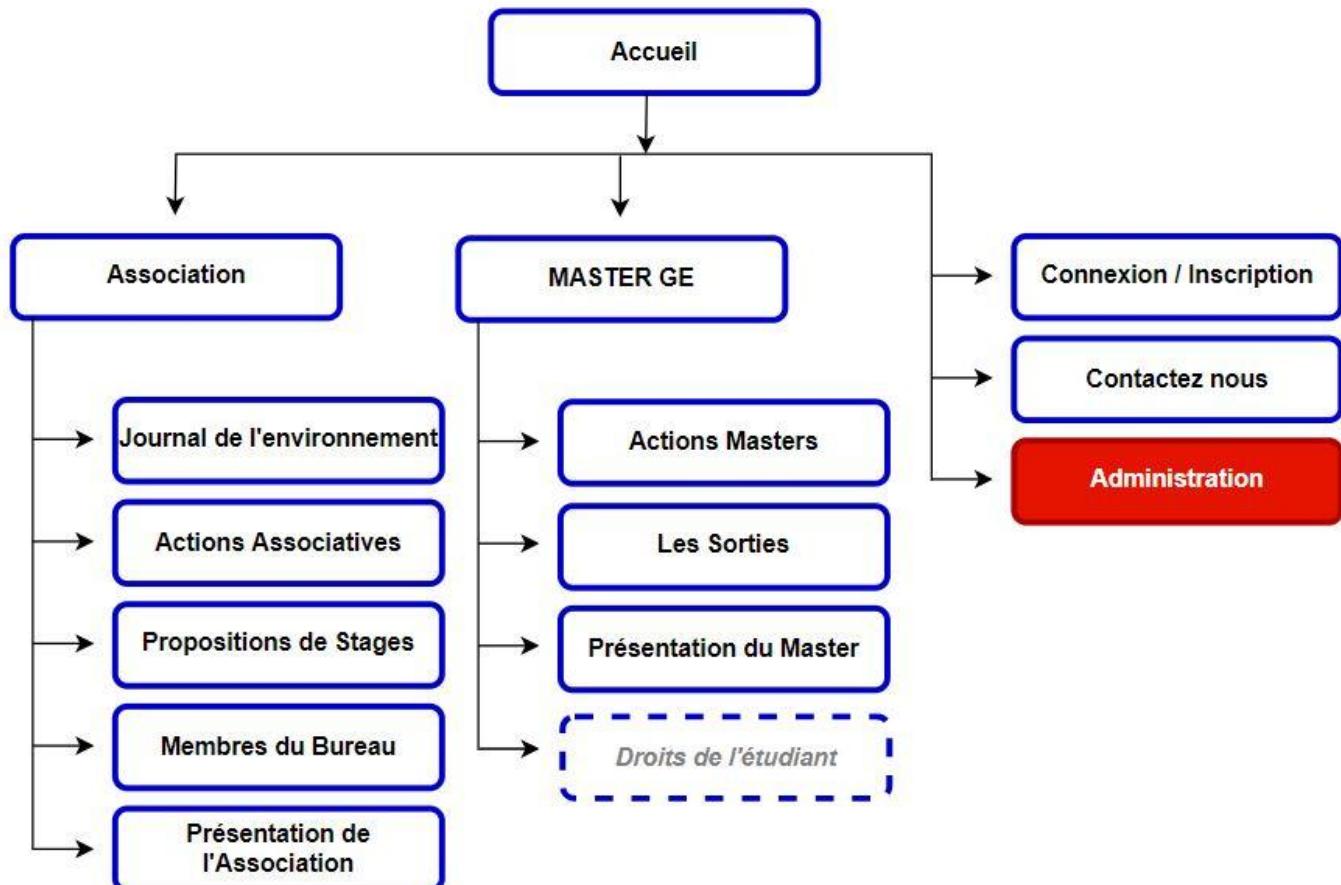
À droite nous avons le wireframe de la page d'accueil responsive.



The responsive wireframe shows a header with logos and a menu icon. It includes a main content area with "Présentation de l'association" and "Les dernières du Journal de l'environnement" sections. Below is a "Nos Partenaires" section with three cards. On the right, there's a "Nous Contactez..." form, "Association..." details, "Localisation" (with a map placeholder), and "Web" social media links. The footer includes copyright and privacy information.



Voici l'arborescence du site:



Arborescence détaillée

Chaque page du site contient plusieurs éléments dont voici les principaux :

La page Accueil :

- Présentation de l'association
- Par un système de carrousel,
 - présentation des 5 dernières nouvelles du journal de l'environnement
 - présentation des 5 dernières actions menées par l'association
 - présentation des 5 dernières actions menées par le Master
 - présentation des 5 dernières sorties pédagogiques organisées par le master
 - présentation des partenaires de l'association
- Avoir la possibilité d'avoir un système de contact

L'ensemble des sections se présente sensiblement de la même manière pour les utilisateurs

- Un titre
- Une photo principale
- Différentes informations, dates,...
- D'autres photos non mises en avant,
- Des liens sur des sites ou des documents...

L'affichage du bureau avec photos et informations consultables.

La partie connexion des utilisateurs et la **déconnexion**.

La partie administration :

Cette partie est réservée aux seules personnes connectées et définies en tant qu' administrateur

Elle permet la modification de pratiquement l'ensemble des données affichées sur le site, ainsi que la gestion complète des différents items.

- La gestion complète des utilisateurs et de leurs rôles pouvant avoir lieu au sein de l'association.
- La présentation du Master, ou de l'association se fait par l'intermédiaire d'une zone de texte entièrement libre réalisée avec le bundle CKEditor.
- Les actions du Master et de l'Association, les sorties, les offres de stages ou de travail utilisent pour les textes longs ce même bundle. D'autres informations prédéfinies sont intégrées. Des photos, des liens sur des documents ou sites viennent compléter les différentes sections. Une gestion complète de ces items est proposée.



Réalisation du site



Planification du projet

J'ai réalisé une liste de tâches détaillées. Je tiens tout de même à notifier qu'il m'aurait été possible de réaliser un diagramme de Gantt, mais n'étant pas encore familiarisé avec le Framework Symfony j'étais dans l'incapacité de quantifier correctement mon temps utilisé pour les différentes parties du site.

Back-End :

- Définir les rôles utilisateurs.
- Création de diagrammes.
- Création du squelette Symfony.
- Création de la base de données.
- Création des entités avec leurs relations.
- Mise en place du service global d'authentification utilisateur.
- Sécurité du site et des utilisateurs.
- Mise en place de la gestion de cookies, reCAPTCHA.
- Test de sécurité

Front-End :

- Mise en place du framework CSS Bootstrap.
- Travail sur la partie Header.
- Travail sur la partie Footer.
- Intégration des différents blocs en fonction des pages.
- Travail sur le responsive design.

Finalisation :

- Passage en production.
- Phase de test.
- Correction des bugs.
- Test du site en version mobile.
- Livraison du site.

Définition des rôles

Pour ce qui concerne les rôles des utilisateurs ayant un compte, quatre niveaux sont prévus :

L'administrateur (ADMIN), les comptes Association Paul Green, les comptes Master GE et les utilisateurs Autres.

L'Administrateur

Les administrateurs sont les seuls qui peuvent modifier l'ensemble des données, modifier les présentations de l'association, du master, de gérer complètement les utilisateurs, les différentes fiches actions associatives, du master, des sorties, des stages...

L'ensemble des comptes suivants ne peuvent que consulter les informations saisies par les administrateurs.

Les comptes de l'Association Paul Green

Ces comptes ont accès à l'ensemble des informations.

Les comptes du Master GE

Ces comptes n'ont accès qu'à la visualisation des informations concernant directement le master, c'est-à-dire, les actions du Masters, les sorties pédagogiques, la présentation du master et de l'association.

Le compte Autres

Ces comptes n'accèdent qu'aux informations concernant l'association, c'est-à-dire la présentation de l'association, les actions de l'association, les propositions de stages

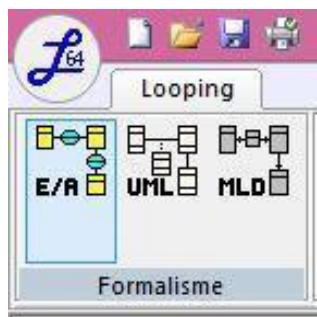
Les visiteurs

Les visiteurs, sans compte, n'accèdent qu'à la page accueil. On ne peut pas considérer qu'ils ont un rôle.



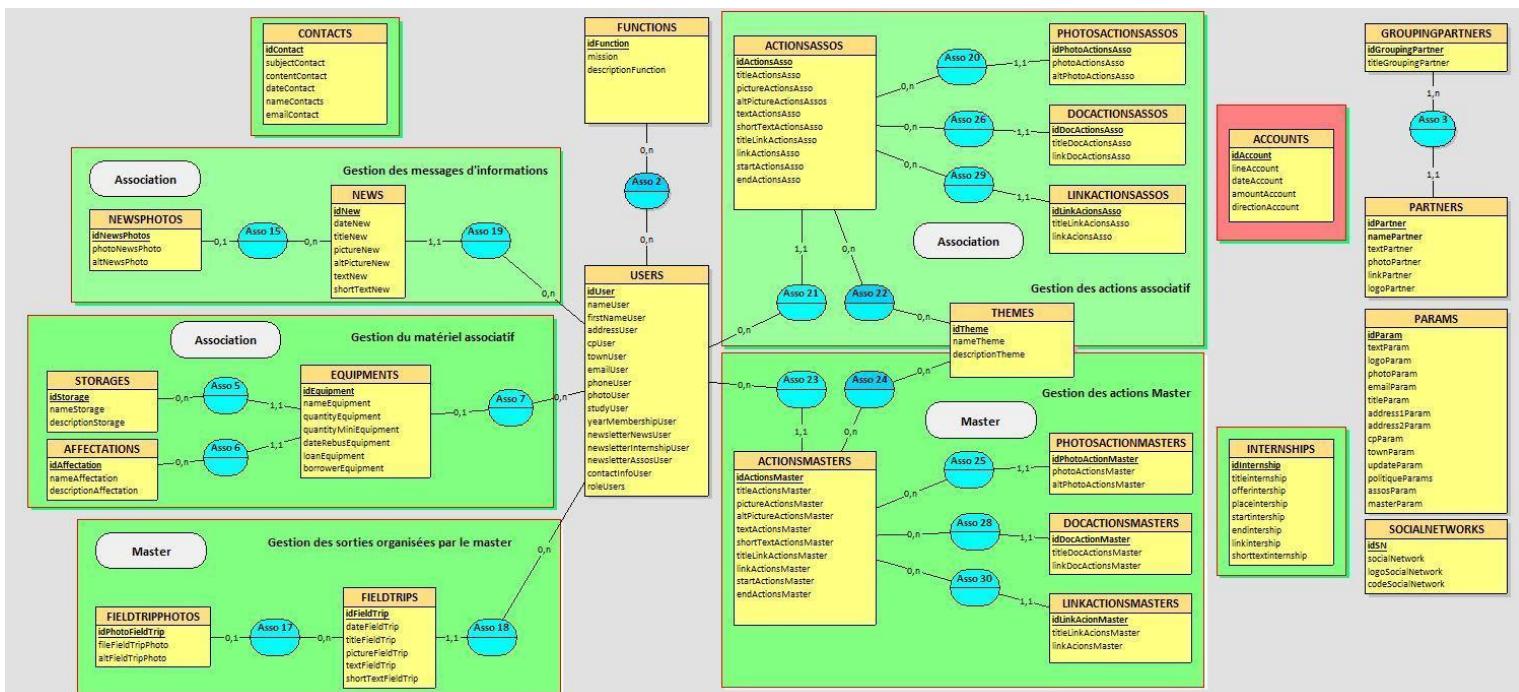
Afin de construire les **MCD** (Modèles Conceptuels de Données) et les **MLD** (Modèles Logiques de Données), j'ai utilisé le logiciel **Looping**. Ce logiciel est gratuit et libre d'utilisation. Il permet la modélisation des entités avec leurs propriétés et types de données, les associations et cardinalités des liens entre les entités. Il est une aide importante pour la réalisation de la base de données.

MCD (Modèle Conceptuel des Données)



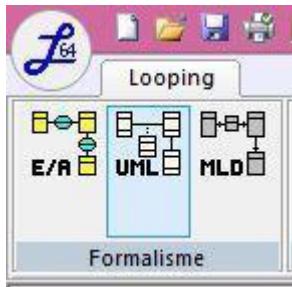
L'étude des demandes pour la réalisation du site, et des différentes informations à stocker, a permis de réaliser le MCD (Modèle Conceptuel de Données). Le logiciel Looping permet de préparer la conception des entités, des propriétés, des caractéristiques des propriétés, et des liens entre entités. La structure globale de la base de données d'une manière schématique est ainsi prête pour un transfert vers la base de données réelle.

Les entités sont en Majuscule et les propriétés sont en camel case, 1^{er} mot tout en minuscule et les suivants attachés, en minuscule avec l'initiale en majuscule.

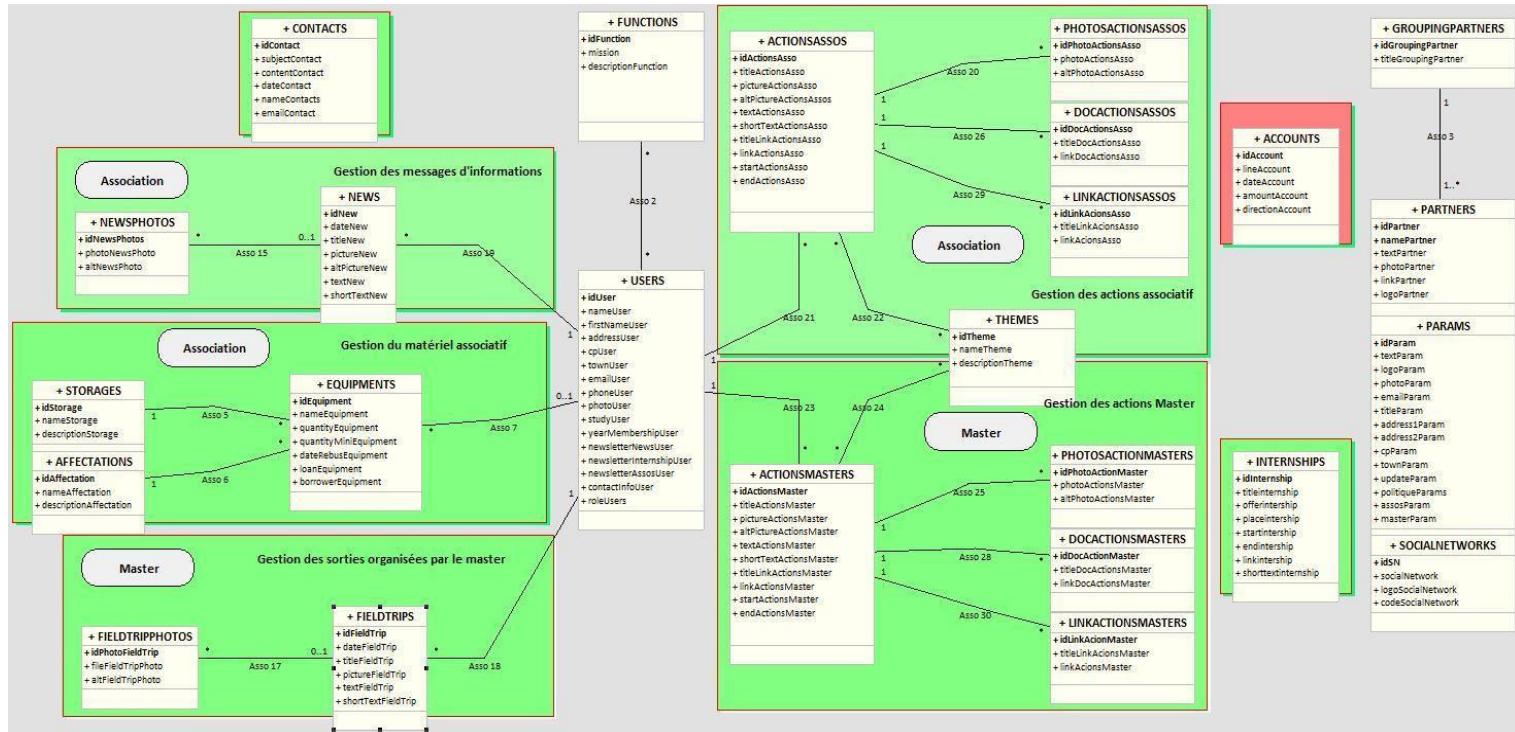


(Voir en Annexe 2 l'affichage du MCD en paysage)

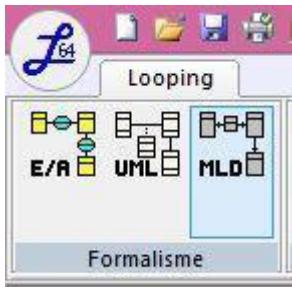
UML (Unified Modeling Language)



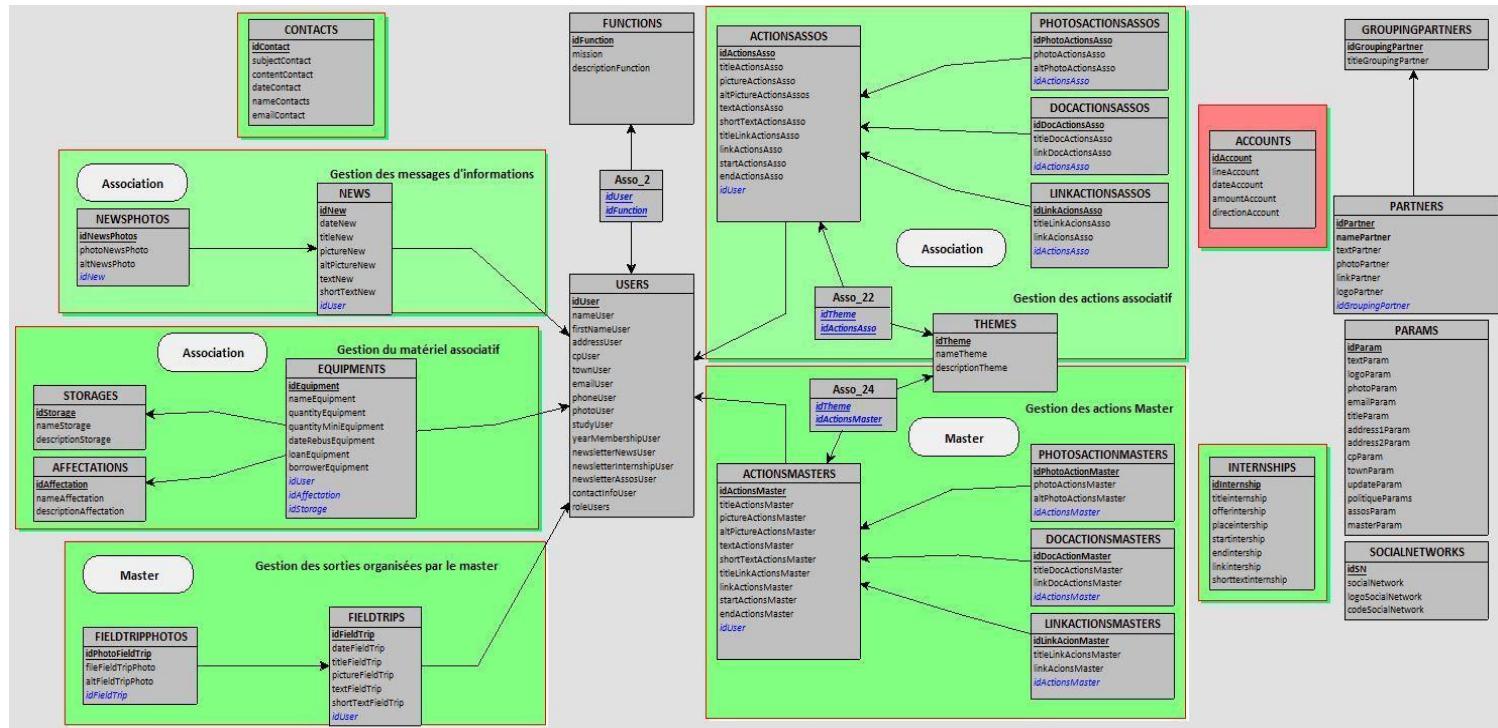
L'UML (Langage de Modélisation Unifié) est généré automatiquement à partir du MCD réalisé.



MLD (Modèle Logique de Données)



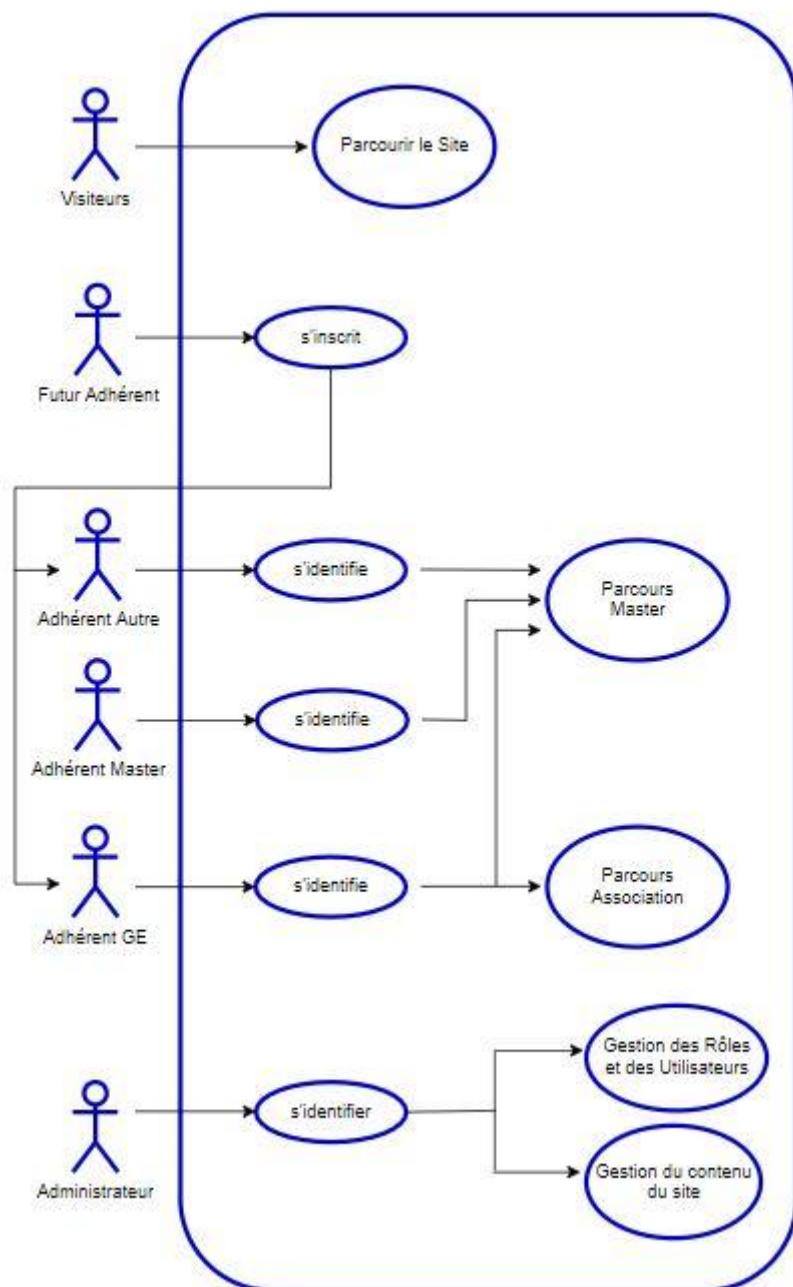
De même, le MLD (Modèle Logique de Données) est réalisé automatiquement. Ce dernier permet de voir la création des tables intermédiaires (tables pivots), entre deux entités ayant des cardinalités 0-1,n / 0-1,n.



(Voir en Annexe 2 l'affichage du MLD en paysage)

Use Case Diagramme

Afin d'avoir une vision globale du comportement fonctionnel du site en fonction de chaque rôle utilisateur, je réalise un diagramme de cas d'utilisation.



Diagrammes d'activités

Afin de pouvoir visualiser de manière beaucoup plus précise les interactions entre usager et l'interface du site j'ai conçu des diagrammes de séquences. Voici deux de mes diagrammes d'activités :

Diagramme d'activité Connexion Utilisateur

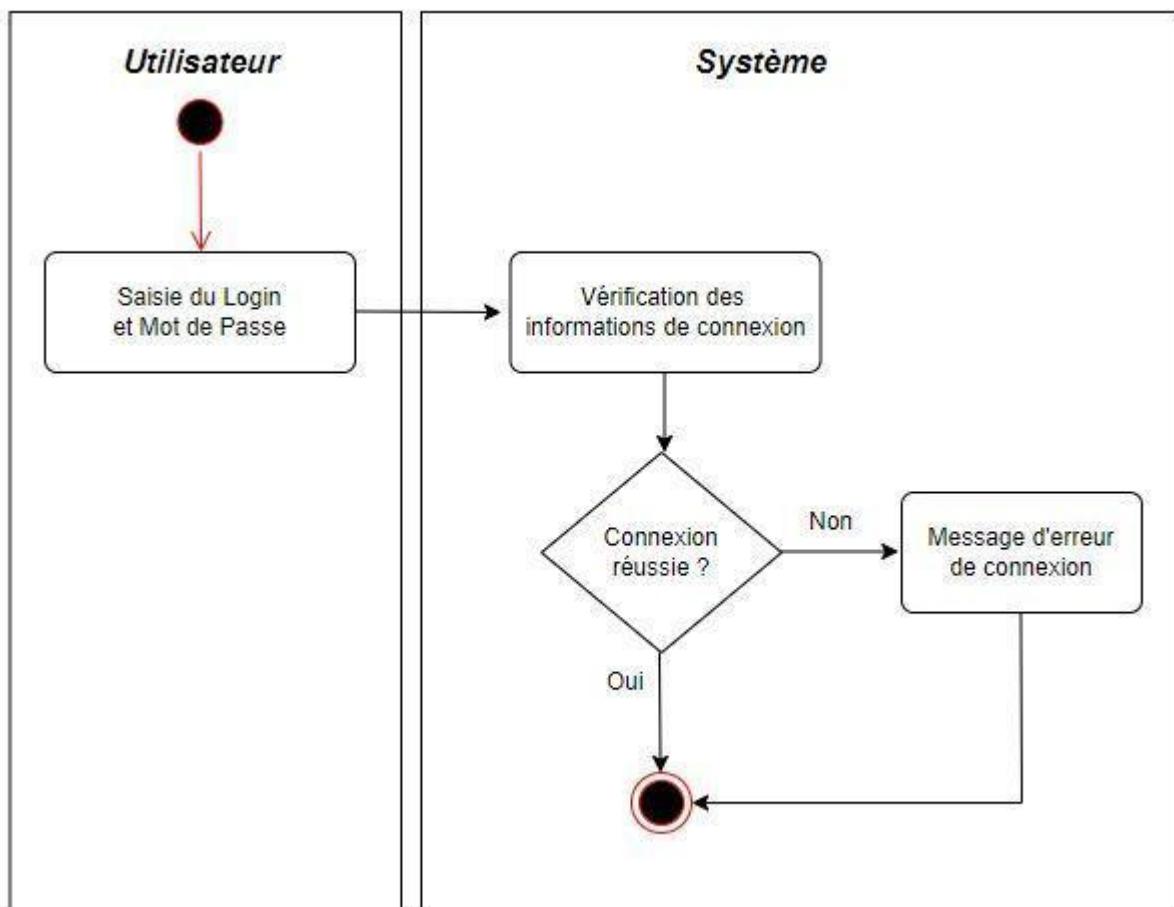
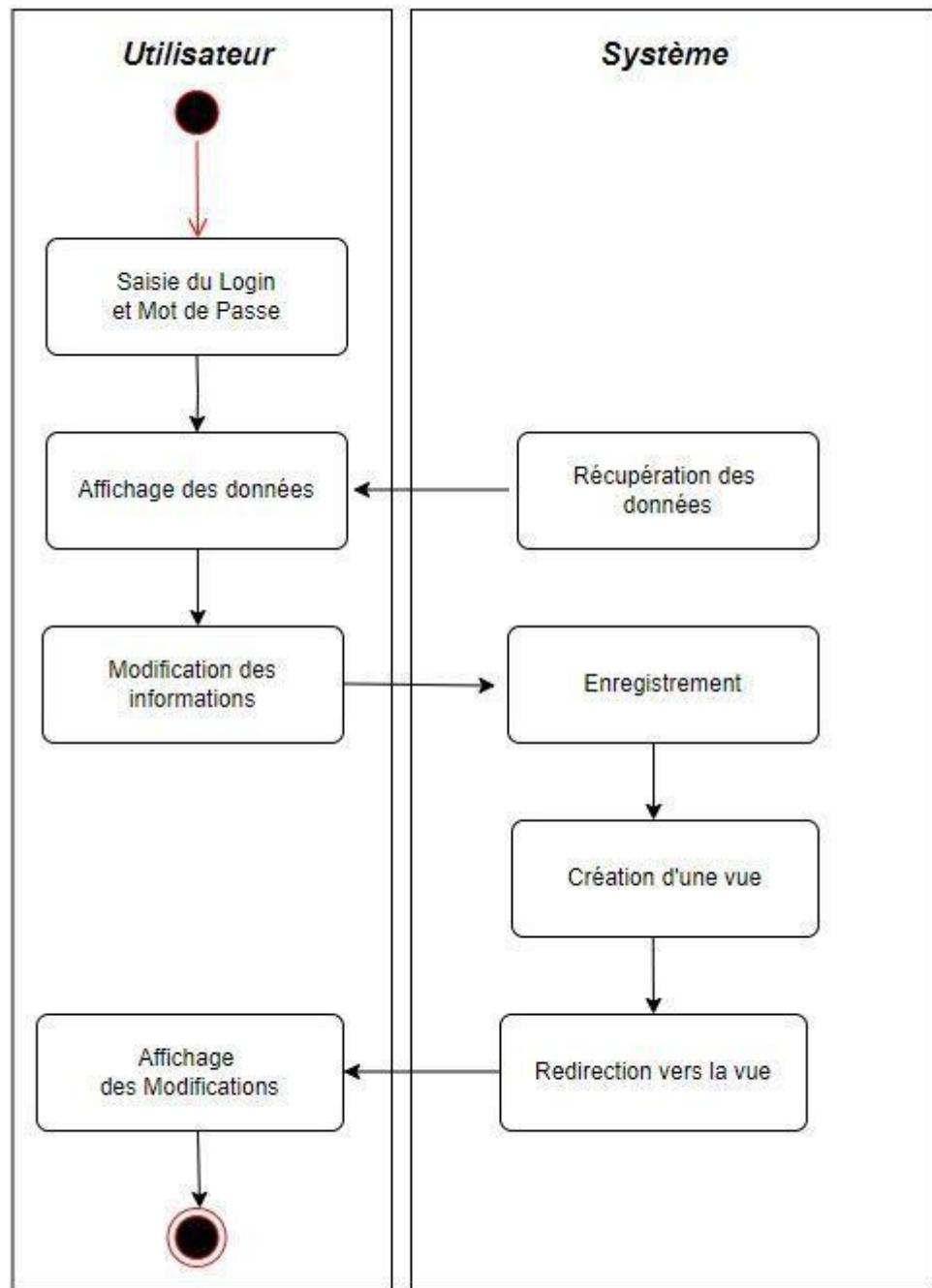


Diagramme d'activité Modification De Contenu





Développement Back-End



Création de la base du squelette et de la base du projet

Pour commencer correctement le projet avec le framework symfony, il faut vérifier en premier lieu que les logiciel PHP et Composer soit correctement installés et utilisable sur Visual Studio Code, l'IDE utilisé.

Le logiciel Composer est un gestionnaire de dépendances et permet à ses utilisateurs de déclarer et d'installer les bibliothèques

PS C:\Jacques\Projet> composer -v

et d'installer les bibliothèques requises pour notre projet.

Composer version 2.2.6 2022-02-04 17:00:38

Le PHP (PHP : Hypertext Preprocesseur) est notre langage de programmation principal, associé au Twig de Symfony et d'autres...

```
PS C:\Jacques\Projet> php -v
PHP 7.4.26 (cli) (built: Nov 16 2021 18:15:31) ( ZTS Visual C++ 2017 x64 )
Copyright (c) The PHP Group
Zend Engine v3.4.0, Copyright (c) Zend Technologies
```

La ligne de commande suivante, grâce au logiciel Composer, installe Symfony, notre socle pour notre projet. Il crée le répertoire « AssociationPaulGreen », avec l'ensemble des modules pour Symfony.

PS C:\Jacques\Projet> composer create-project symfony/website-skeleton AssociationPaulGreen

Afin de démarrer le serveur web de développement, du serveur interne de php, il nous suffit de lancer la commande ci-dessous :

```
PS C:\Jacques\Projet\AssociationPaulGreen> php -S 127.0.0.1:8000 -t public
```

Pour le fonctionnement, il a bien sûr fallu installer au préalable un serveur web (type Apache). Pour notre part nous avons installé la plateforme de développement web de type Wamp, « WampServer ». C'est un environnement comprenant trois serveurs (Apache, MySQL et [MariaDB](#)), un interpréteur de script (PHP), ainsi que [phpMyAdmin](#) pour l'administration Web des bases MySQL, comprenant trois serveurs (Apache, MySQL et MariaDB), un interpréteur de script (PHP), ainsi que phpMyAdmin pour l'administration Web des bases MySQL.



Symfony étant en POO (Programmation Orientée Objet), il faut créer un ‘controller’ qui va permettre de retourner une réponse à une route ou une requête donnée. On rentre alors dans le terminal la commande suivante pour créer notre controller “MainController” :

```
PS C:\Jacques\Projet\AssociationPaulGreen> php bin/console make:MainController
```

Construction de la base de données

Symfony utilise par défaut l'ORM Doctrine.

Afin que le projet soit opérationnel, il faut créer une base de données de type MySQL.

Il nous faut modifier le fichier `.env`, à la racine du projet.

La variable d'environnement « **DATABASE_URL** » est modifiée afin d'obtenir une relation entre mon projet et la base de données sous MySQL.

```
## #> doctrine/doctrine-bundle ## #
DATABASE_URL=mysql://root:@127.0.0.1:3306/AssociationPaulGreen?serverVersion=5.7&charset=utf8mb4
## #< doctrine/doctrine-bundle ## #
```

Le terme dans la commande Database, « `AssociationPaulGreen` » est alors le nom de base de données dans MySQL.

Pour la créer, il faut lancer la commande suivante

```
PS C:\Jacques\Projet\AssociationPaulGreen> php bin/console doctrine:database:create
```

Il est important de vérifier si elle a bien été créée dans PhpMyAdmin.

The screenshot shows the PhpMyAdmin interface with the following details:

- Serveur courant : MySQL
- Récentes / Préférées
- Nouvelle base de données
- associationpaulgreen (highlighted with a red box)
- information_schema
- mysql
- performance_schema
- sys

Création des entités

Création de l'entité Users

Sur Symfony, la construction des entités se fait par le biais de ligne de commande dans le terminal à la racine du projet et bénéficie d'une trame chronologique nous demandant sa structure avec les paramètres des champs et les relations entre tables.

La première entité à créer est l'entité pour les utilisateurs du site : Users. Elle permet la gestion complète des personnes connectées, et de définir les rôles et droits de chacun. Elle est spécifique à la sécurité. Elle encode les mots de passe, et gère l'authentification pour la connexion au site, et l'enregistrement des utilisateurs.

Pour générer l'entité Users, et toute la gestion de la sécurité, la ligne de commande suivante est lancée :

```
PS C:\Jacques\Projet\AssociationPaulGreen> php bin/console make:user
```

Pour nous ici, le nom de l'entité est Users.

Cette commande nous crée l'ensemble des modules dans notre répertoire de notre projet pour la gestion de la sécurité.

Pour la table dans la base de données, il suffit de migrer l'entité vers la base de données.

Les lignes de commande dans le terminal suivantes lancent la mise à jour de la base de données.

Le fichier de migration de la classe, créé par la commande, liste une suite de requêtes SQL afin de modifier la base de données en fonction de l'entité préalablement préparée dans le projet, et suivant l'existant.

```
PS C:\Jacques\Projet\AssociationPaulGreen> php bin/console make:migration
```

Puis, cette commande exécute ces requêtes SQL du fichier de migration.

```
PS C:\Jacques\Projet\AssociationPaulGreen> php bin/console doctrine:migrations:migrate
```

En vérifiant la base de données, nous constatons que la table users a bien été générée dans la base de données.



Construction de l'authentification

Grâce aux lignes suivantes saisies dans le terminal, Symfony créer un système authentification par email et mot de passe.

```
PS C:\Jacques\Projet\AssociationPaulGreen> php bin/console make:auth
```

Une fois rentré les bons paramètres, il suffit de mettre à jour la base de données de la même manière que précédemment :

```
PS C:\Jacques\Projet\AssociationPaulGreen> php bin/console make:migration
```

Puis :

```
PS C:\Jacques\Projet\AssociationPaulGreen> php bin/console doctrine:migrations:migrate
```

J'ai par la suite rajouté une double vérification de mot de passe dans le formulaire d'enregistrement sur le site.

J'ai eu recours à deux bundles afin d'apporter un peu plus d'ergonomie au site :

Pour les installer j'ai utilisé les lignes de commande suivantes :

```
PS C:\Jacques\Projet\AssociationPaulGreen> composer require symfonycasts/reset-password-bundle
```

Et

```
PS C:\Jacques\Projet\AssociationPaulGreen> composer require symfonycasts/verify-email-bundle
```

Le premier Bundle « reset password » créer un système de reset de mot de passe en cas de perte ou d'oubli.

Le second permet de rajouter une vérification par email de ce changement de mot de passe.

Construction du formulaire d'enregistrement

La création de ce formulaire de saisie se fait d'une manière automatique par l'intermédiaire de la commande du terminal suivante :

```
PS C:\Jacques\Projet\AssociationPaulGreen> php bin/console make:registration-form
```

Un formulaire d'inscription est ainsi initié.

Il est bien entendu qu'il est maintenant important d'adapter l'ensemble des entités, classes, de la base de données, des tables aux besoins du projet. Sur la classe Users, des champs supplémentaires sont à prévoir.

De même les redirections de page sont à modifier suivant les besoins.

Des droits d'utilisation du site sont élaborés suivant ce qui a été défini par les étudiants de l'association. Ces droits sont ajoutés à l'entité Users.

```
use Symfony\Component\Security\Core\User\UserInterface;
use Symfony\Component\Security\Core\User\PasswordAuthenticatedUserInterface;

/**
 * @ORM\Entity(repositoryClass=UsersRepository::class)
 * @UniqueEntity(fields={"email"}, message="Il existe déjà un compte avec cet
email !")
 */
class Users implements UserInterface, PasswordAuthenticatedUserInterface
{

    public const ROLE_USER = 'ROLE_USER';
    public const ROLE_OTHER = 'ROLE_OTHER';
    public const ROLE_GE = 'ROLE_GE';
    public const ROLE_ADMIN = 'ROLE_ADMIN';

    public function __construct()
    {
        $this->roles = [$self::ROLE_USER];
    }
}
```

Ces différents niveaux de droits sont contrôlés par l'annotation `@isGranted` ou la fonction `is_Granted()`.

Par exemple, la partie des équipements est réservée exclusivement aux Administrateurs :

```
<?php

namespace App\Controller;

use App\Entity\Equipments;
use App\Form\EquipmentsType;
use App\Repository\EquipmentsRepository;
use Sensio\Bundle\FrameworkExtraBundle\Configuration\IsGranted;
use Symfony\Bundle\FrameworkBundle\Controller\AbstractController;
use Symfony\Component\HttpFoundation\Request;
use Symfony\Component\HttpFoundation\Response;
use Symfony\Component\Routing\Annotation\Route;

/**
 * @Route("/equipments")
 * @IsGranted("ROLE_ADMIN")
 */
class EquipmentsController extends AbstractController
{
    /**
     * @Route("/", name="app_equipments_index", methods={"GET"})
     */
    public function index(EquipmentsRepository $equipmentsRepository): Response
    {
```

Dans le cas où une personne saisissait l'adresse « /equipments » directement sans avoir les droits admin, elle serait renvoyée vers la page de connexion.

Ou par exemple dans le fichier Twig du menu :

```
{% if is_granted('ROLE_ADMIN') %}
    <li class="nav-item dropdown">
        <a class="nav-link dropdown-toggle"
            href="#" id="navbarDropdown" role="button"
            data-bs-toggle="dropdown" aria-expanded="false">
```

Création des Autres Entités

L'ensemble des entités est créé sur le même principe.

La ligne de commande lance la création des entités dans notre projet.

Pour créer une entité avec Symfony, il suffit d'écrire dans le terminal la commande :

```
PS C:\Jacques\Projet\AssociationPaulGreen> php bin/console make:entity
```

Un questionnaire nous permet alors de définir ses propriétés, et les caractéristiques de celles-ci.

Exemple détaillé pour l'entité SocialNetworks :

```
Class name of the entity to create or update (e.g. BraveElephant):  
> SocialNetWorks  
created: src/Entity/SocialNetWorks  
created: src/Repository/SocialNetWorksRepository.php  
Entity generated! Now let's add some fields!  
You can always add more fields later manually or by re-running this command.  
New property name (press <return> to stop adding fields):  
> socialNetWork  
  
Field type (enter ? to see all types) [string]:  
>  
  
Field length [255]:  
> 50  
  
Can this field be null in the database (nullable) (yes/no) [no]:  
>  
  
updated: src/Entity/SocialNetWorks.php  
  
Add another property? Enter the property name (or press <return> to stop adding fields):  
> logoSocialNetWork  
  
Field type (enter ? to see all types) [string]:  
>  
  
Field length [255]:  
>  
  
Can this field be null in the database (nullable) (yes/no) [no]:  
>  
  
updated: src/Entity/SocialNetWorks.php  
  
> codeSocialNetWork
```

```
Field type (enter ? to see all types) [string]:
```

```
>
```

```
Field length [255]:
```

```
>
```

```
Can this field be null in the database (nullable) (yes/no) [no]:
```

```
>
```

```
updated: src/Entity/SocialNetWorks.php
```

```
> linkSocialNetWork
```

```
Field type (enter ? to see all types) [string]:
```

```
>
```

```
Field length [255]:
```

```
>
```

```
Can this field be null in the database (nullable) (yes/no) [no]:
```

```
>
```

```
updated: src/Entity/SocialNetWorks.php
```

```
Add another property? Enter the property name (or press <return> to stop adding fields):
```

```
>
```

```
Success!
```

```
Next: When you're ready, create a migration with make:migration
```

Après avoir rentré les bons paramètres, il suffit de mettre à jour la base de données de la même manière que précédemment :

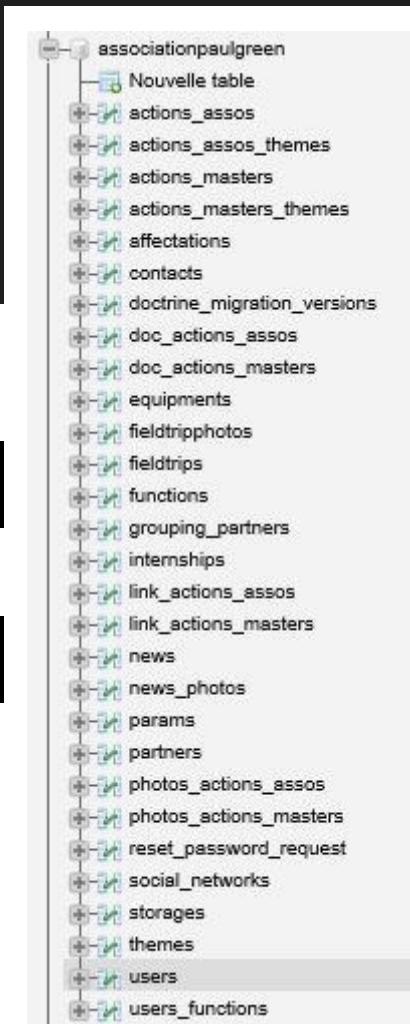
```
PS C:\Jacques\Projet\AssociationPaulGreen> php bin/console  
make:migration
```

Puis :

```
PS C:\Jacques\Projet\AssociationPaulGreen> php bin/console  
doctrine:migrations:migrate
```

La base de données est ainsi mise à jour.

Pour chaque entité, nous définissons ses propriétés, puis nous synchronisons avec la base de données.



Pour être clair, les différentes ‘entités’, sont autant de ‘classes’ dans Symfony, et les propriétés qui les caractérisent autant de ‘fonctions’ dans ces ‘classes’.

Manipulation des données - La fonction CRUD

Symfony nous permet de générer une suite de fichiers pour la gestion complète des entités. La fonction CRUD (Created, Read, Update, Delete) nous construit ces fichiers et nous pouvons l’appliquer sur l’ensemble de nos entités modifiable par les utilisateurs. Elle crée l’ensemble des fonctions de gestion dans les « controllers » ainsi que les formulaires liés à nos entités, et les « templates » associées .

La commande suivante est lancée pour chaque entité que nous voulons gérer :

```
PS C:\Jacques\Projet\AssociationPaulGreen> php bin/console make:crud
```

Certaines entités ne sont pas gérées par ce biais.

Une fois ces CRUD réalisés, nous pouvons adapter plus précisément les gestions, améliorer les qualités de présentation des informations.

Les administrateurs sont les seuls users à pouvoir modifier les données. Une fois connecté, ils ont accès à un nouveau menu Administration, leur permettant la gestion complète.

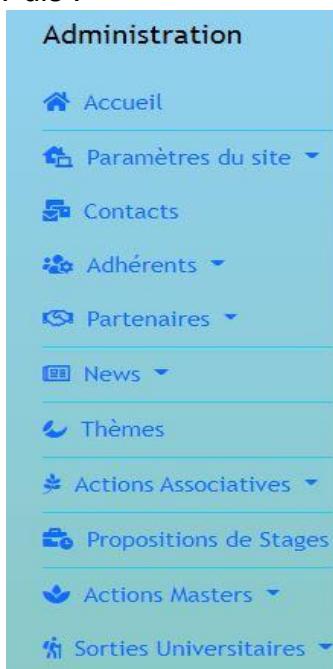


PAUL 

Association Paul Green

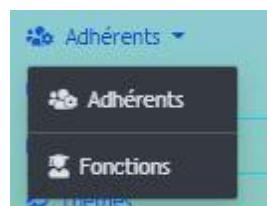


Puis :



En cliquant sur Administration(en rouge) sur la barre de menu principal,

Les utilisateurs ont accès à la gestion complète des données. Les gestions des entités liées sont souvent regroupées.



Plus loin nous verrons une liaison spécifique pour l'affichage du bureau de l'association.

Dans beaucoup de gestions de données, des textes libres, importants et de présentations sont mis à la disposition.

Par exemple pour les paramètres de l'application et en particulier la présentation de la page d'accueil.

Modification des Paramètres de l'application

Titre :
Association Paul Green

Sous-Titre :
Pour vous mettre au vert !

Mail :
asso.paul.green@univ-montp3.fr

Texte de Présentation :

L'ASSOCIATION Paul Green

L'association Gaïa a été fondée il y a près de 20 ans par des étudiants du Master Gestion de l'Environnement et du Développement Durable (GEDD). Portée aujourd'hui par des étudiants actuels et diplômés du Master GEDD, l'Association a pour mission de développer et animer le réseau d'alumni du Master et de contribuer ensemble à des actions en faveur de l'environnement et du développement durable. Gaïa a pour ambition de faire rayonner ses actions et ses valeurs auprès d'un public plus large que le réseau des alumni GEDD afin de sensibiliser le plus grand nombre aux enjeux environnementaux.

Nos missions et projets phares :

Gestion du réseau alumni

Par ses 30 années d'existence, le Master GEDD a permis de former près de 700 professionnels sur les thématiques variées du développement durable : de la gestion de l'eau à la gestion des déchets en passant par la gestion de ressources naturelles et l'amélioration continue des pratiques de développement durable.

Pour cela, plusieurs projets :

- Mise en place et animation d'une plateforme d'échanges pour les étudiants actuels et diplômés du Master GEDD.
- Organisation d'événements types cafés-débats

Logo : Aucun fichier choisi

Photo : Aucun fichier choisi





Adresse :
Rte de Mende

Complément d'adresse :

Code Postal :

Ville :

Pour ce texte de présentation nous utilisons le bundle ckEditor, un éditeur de texte open-source. Il ressemble à n'importe quel éditeur de texte, et donc facilement appréhender par les utilisateurs. Son installation se fait via la « composer ».

```
PS C:\Jacques\Projet\AssociationPaulGreen> composer require friendsofsymfony/ckeditor-bundle
```

L'installation se fait dans public/bundle/fosckeditor, et s'inscrit dans la fichier config/bundles.php

```
FOS\CKEditorBundle\FOSCKEditorBundle::class => ['all' => true],  
FM\ElfinderBundle\FMElfinderBundle::class => ['all' => true],
```

La deuxième étape est d'installer ckeditor en entrant la commande Symfony suivante :

```
PS C:\Jacques\Projet\AssociationPaulGreen> php bin/console ckeditor:install
```

Puis on vérifie que les assets soient bien installés avec la ligne de commande suivante :

```
PS C:\Jacques\Projet\AssociationPaulGreen> php bin/console assets:install public
Installing assets as hard copies.

-----
  Bundle          Method / Error
-----
OK  FOSCKEditorBundle    copy
OK  FMElfinderBundle   copy
-----

! [NOTE] Some assets were installed via copy. If you make changes to these assets
you have to run this command again.

[OK] All assets were successfully installed.
```

Puis il faut paramétriser les options auxquels les utilisateurs auront accès dans le fichier config/packages/fos_ckeditor.yaml.

```
# Read the documentation: https://symfony.com/doc/current/bundles/FOSCKEditorBundle/index.html

twig:
  form_themes:
    - '@FOSCKEditor/Form/ckeditor_widget.html.twig'
fos_ck_editor:
  configs:
    my_config:
      toolbar:
        - { name: "styles", items: ['Bold', 'Italic', 'Underline', 'Strike', 'Blockquote', '-', 'Link', '-', 'RemoveFormat', '-', 'NumberedList', 'BulletedList', '-', 'Outdent', 'Indent', '-', '-', 'JustifyLeft', 'JustifyCenter', 'JustifyRight', 'JustifyBlock', '-', 'Image', 'Table', '-', 'Styles', 'Format', 'Font', 'FontSize', '-', 'TextColor', 'BGColor', 'Source'] }
        filebrowserBrowseRoute: elfinder
        filebrowserBrowseRouteParameters: []
```

Les utilisateurs ont la possibilité aussi d'ajouter des photos directement dans la zone de texte.

Pour cela j'ai ajouté l'option « parcourir le serveur », avec le bundle ElFinder.

Nous utilisons le composer pour l'installer.

```
PS C:\Jacques\Projet\AssociationPaulGreen> composer require helios-ag/fm-elfinder-bundle
```

Puis la partie JavaScript du bundle

```
PS C:\Jacques\Projet\AssociationPaulGreen> php bin/console elfinder:install
```

Nous configurons ensuite le fichier config/packages/fm_elfinder.yaml

```
fm_elfinder:
    assets_path: /assets # chemin des fichiers JS
    instances:
        default:
            locale: fr # Langue
            editor: ckeditor # Editeur utilisé
            fullscreen: true # Taille d'affichage
            theme: smoothness # Thème à utiliser
        #   include_assets: true # Charge automatiquement les fichiers nécessaires
        connector:
            debug: false # Désactive le débug
            roots:
                uploads:
                    show_hidden: false # Masque les fichiers cachés
                    driver: LocalFileSystem # Pilote des fichiers
                    path: uploads/images # Chemin d'upload
                    upload_allow: ['image/png', 'image/jpg', 'image/jpeg'] # Fichiers autorisés
                    upload_deny: ['all'] # Fichiers interdits
                    upload_max_size: 2M # Taille maximum
```

Voir plus haut la configuration complémentaire dans le fichier config/packages/fos_ckeditor.yaml (les 2 dernières lignes).

Il faut ensuite intégrer ckEditor dans les formulaires. Dans notre cas, le fichier src/Form/ParamsType.php est modifié afin d'ajouter la classe CKEditorType à la propriété « textParam »

```
use FOS\CKEditorBundle\Form\Type\CKEditorType;

class ParamsType extends AbstractType
{
    public function buildForm(FormBuilderInterface $builder, array $options): void
    {
        $builder
            ->add('textParam', CKEditorType::class, [
                'label' => 'Texte de Présentation : ',
                'required'=> false,
```

```
        'attr' => [
            "class" => 'ckeEditor',
        ],
    ])
)
```

Et pour terminer il faut afficher un contenu généré par un ckeditor d'une manière correcte, avec la propriété twig « raw », qui interprète le HTML.

```
{# ===== Présentation ===== #}
<section class="PresentationAssos">
    <div class="container">
        <div class="row">
            <div class="col-md-12 d-flex flex-row justify-content-center mb-3">
                
            </div>
            <div class="col-md-12">
                {{ params[0].textParam|raw }}
            </div>
        </div>
    </div>
</section>
```

Sécurisation et tests

Veille sur la sécurité

Pour effectuer ma veille de sécurité, j'ai consulté le site de l'[OWASP](#) (Open Web Application Security Project), une communauté en ligne travaillant sur la sécurité des applications Web. Il publie régulièrement des recommandations de sécurité et explique aux internautes comment sécuriser leurs projets. Il présente les dix principales attaques sur les sites.



Dans la documentation Symfony, une partie est réservée à la sécurité.



J'ai consulté aussi certains blogs dont celui de Wanadev, et en autre pour article « 5 sources d'amélioration pour la sécurité de votre application Symfony ».



<https://www.wanadev.fr/78-5-sources-d-amelioration-pour-la-securite-de-votre-application-symfony/>

Ou encore le blog de Vaadata, une société spécialisée dans le ‘pentest’, les tests d’intrusion.



<https://www.vaadata.com/blog/fr/>

Tests unitaires de sécurité

Pour effectuer des tests unitaires j'ai installé le Framework Open Source PHPUnit.

Ce dernier est dédié au langage de programmation PHP et nous permet de tester certaines fonctionnalités du site internet.

Pour cela on utilise Composer et on rentre les lignes de commande suivantes :

```
PS C:\Jacques\Projet\AssociationPaulGreen> composer require --dev symfony/phpunit-bridge
```

Puis :

```
PS C:\Jacques\Projet\AssociationPaulGreen> composer require --dev symfony/browser-kit  
symfony/css-selector
```

Pour tester les “controllers” et vérifier que toutes les routes que l'on a définies mènent bien à une page, je crée un dossier ‘Controllers’ dans le dossier ‘test’ (présent dans l’architecture de base d’un projet Symfony), dans lequel j’ajoute un nouveau fichier pour chacune de mes entités.

Exemple pour l’entité ‘Themes’ :

```
<?php  
//tests/Controller/ThemesControllerTest.php  
namespace App\Tests\Controller;  
  
use Symfony\Bundle\FrameworkBundle\Test\WebTestCase;
```

```

class ThemesControllerTest extends WebTestCase
{
    public function testNewThemes()
    {
        $client=static::createClient();

        $client->request('GET', '/themes_new');

        $this->assertEquals(200, $client->getResponse()->getStatusCode());
    }
}

```

Ici nous avons un test pour vérifier si la page d'ajout d'un nouveau thème existe et fonctionne correctement.

Pour mes Formulaires :

Je cherche également à tester mes formulaires. Pour cela, dans la même logique que pour les “controllers”, je crée dans ‘tests’ un dossier ‘Form’, et j’ajoute un nouveau fichier pour chaque formulaire que je souhaite tester. Par exemple pour l’entité Internships’ (mon formulaire des stages) :

```

<?php

namespace App\Test\Form;

use Datetime;
use App\Entity\Internships;
use Symfony\Bundle\FrameworkBundle\Test\KernelTestCase;

class InternshipsFormTest extends KernelTestCase{

    public function testInternships(){

$date=new Datetime();

$internships=(new Internships())

->setOfferinternship('77')
->setPlaceinternship('77')
->setStartinternship($date)

```

```

->setEndinternship($date)
->setLinkinternship('77');

self::bootKernel();
$error = self::$container->get('validator')->validate($internships);
$this->assertCount(0,$error);
}

}

```

Pour lancer le test, il suffit d'exécuter la commande suivante :

```
PS C:\Jacques\Projet\AssociationPaulGreen> php bin/phpunit
```

Dans notre cas les résultats sont Ok:

```

PS C:\Jacques\Projet\AssociationPaulGreen> php bin/phpunit
PHPUnit 9.5.19 #StandWithUkraine

Testing
.
1 / 1 (100%)

Time: 00:00.374, Memory: 24.00 MB

OK (1 test, 1 assertion)

Remaining indirect deprecation notices (1)

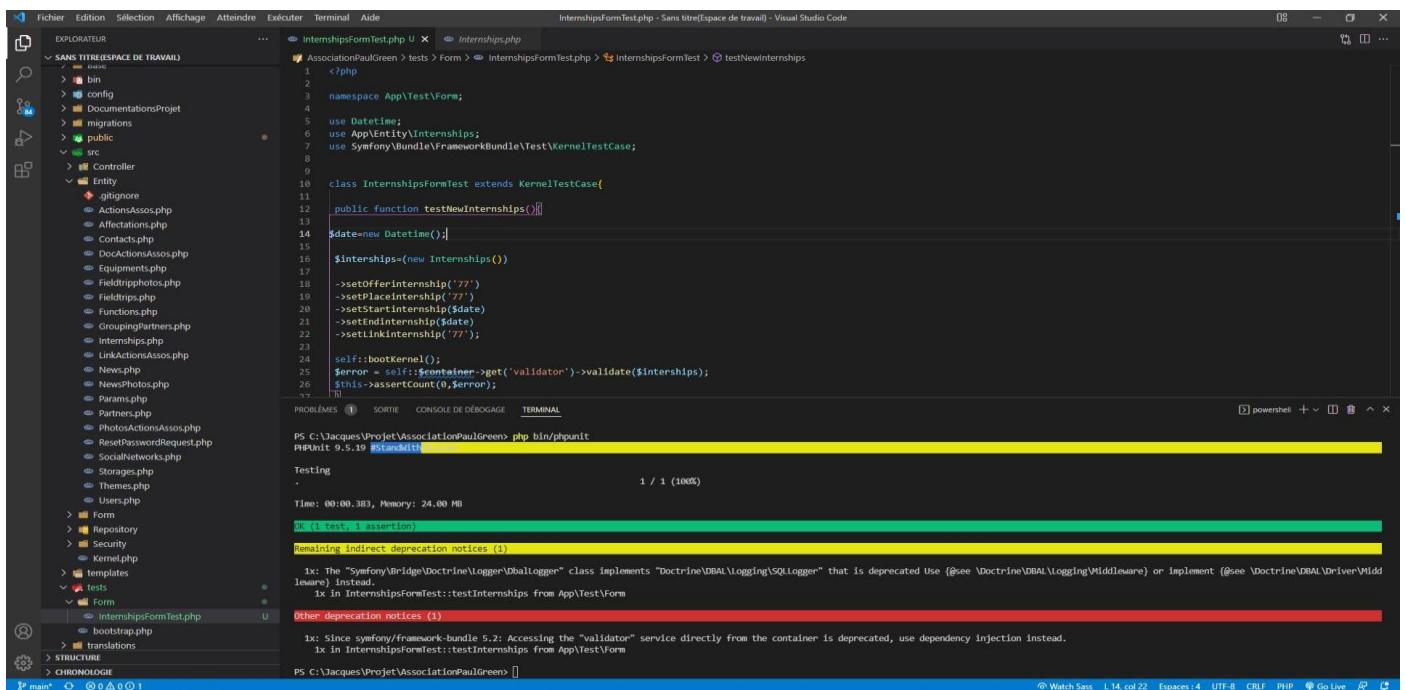
1x: The "Symfony\Bridge\Doctrine\Logger\DbalLogger" class implements "Doctrine\DBAL\Logging\SQLLogger" that is deprecated. Use {@see \Doctrine\DBAL\Logging\Middleware} instead.
  1x in InternshipsFormTest::testNewInternships from App\Test\Form

Other deprecation notices (1)

1x: Since symfony/framework-bundle 5.2: Accessing the "validator" service directly from the container is deprecated, use dependency injection instead.
  1x in InternshipsFormTest::testNewInternships from App\Test\Form

PS C:\Jacques\Projet\AssociationPaulGreen> []

```



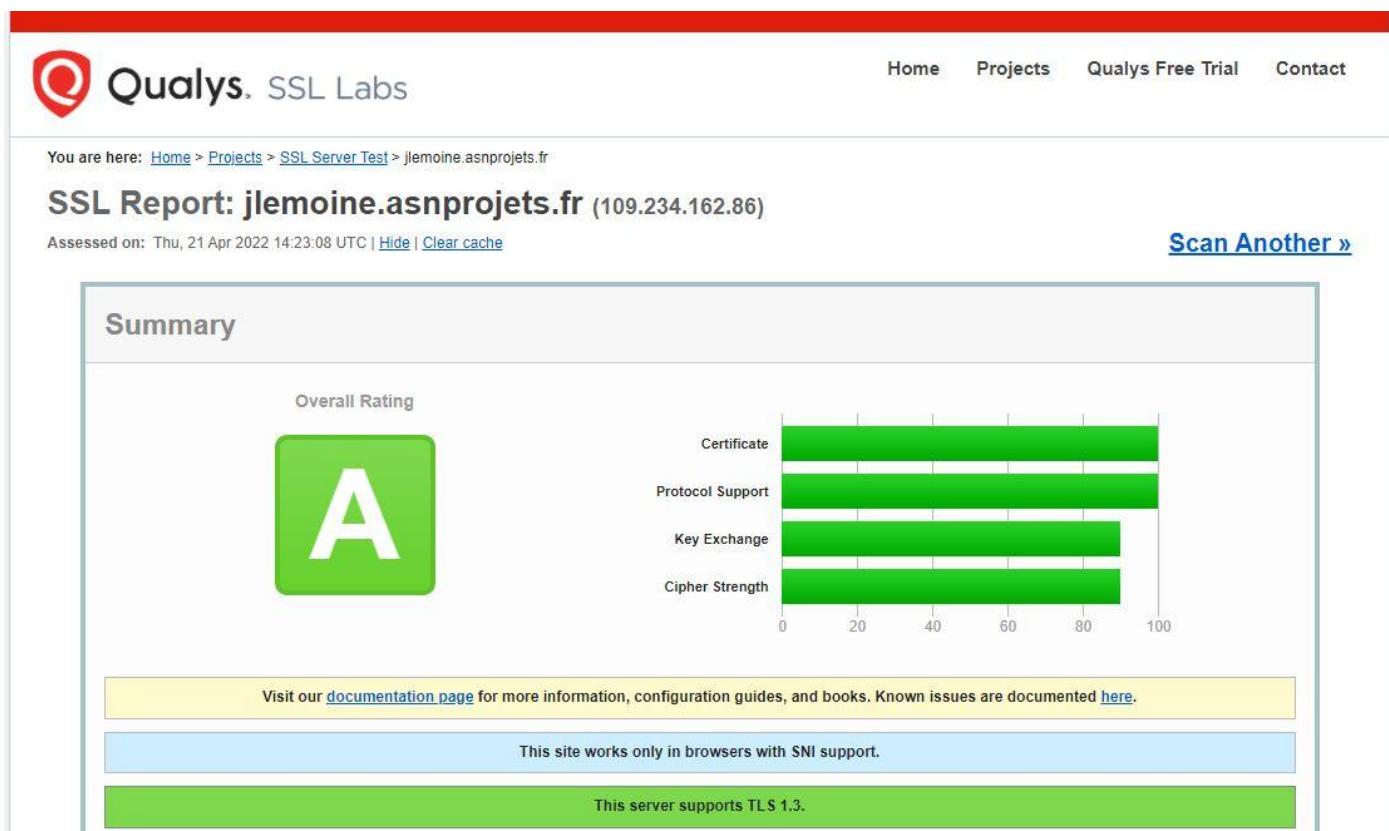
Mise en place du « https » et sécurité en production

Le https (= protocole de transfert hypertextuel sécurisé) est variante du protocole http (HyperText Transfer Protocol) chiffrée grâce au protocole TLS (Transport Layer Security). Il garantit la confidentialité et l'intégrité des données envoyées par et reçues du serveur.

Pour des raisons de sécurité, mais également pour optimiser le référencement d'un site (*Exemple : Dans ses résultats de recherche, Google favorise les sites en https*), il devient indispensable de mettre en place le https.

Pour cela, une fois le site mis en production à partir du panneau de contrôle cPanel, notre formateur utilise Let's Encrypt qui permet de délivrer une certification de validation de domaine.

J'ai ensuite effectué des tests de sécurité grâce au site SSL Labs , qui a donné à mon site un score tout à fait satisfaisant.



Installation du reCAPTCHA

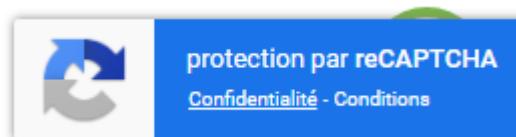
Le **reCAPTCHA** protège votre site Web contre la fraude et les abus. Ce service de sécurité utilise un moteur d'analyse des risques avancés et des défis adaptatifs pour empêcher les logiciels malveillants de se livrer à des activités abusives sur les sites Web. Les utilisateurs légitimes pourront ainsi se connecter, effectuer des achats, afficher des pages ou créer des comptes tandis que les faux utilisateurs sont bloqués.

Afin de l'installer, je me suis inscrit sur le site <https://www.google.com/recaptcha/about/> et j'ai ensuite intégré le reCAPTCHA sur mon site via le code :

```
{# ===== reCAPTCHA ===== #}

<script
src="https://www.google.com/recaptcha/api.js?render=6LdcH44fAAAAACrehdt6C6m6TerjtWj04krFK05w"><
/script>
<script>
    function onClick(e) {
        e.preventDefault();
        grecaptcha.ready(function() {
            grecaptcha.execute('6LdcH44fAAAAACrehdt6C6m6TerjtWj04krFK05w', {action:
'submit'}).then(function(token) {
                console.log('captcha');
            });
        });
    }
</script>
```

Un symbole bleu apparaît alors en bas à droite de la page Web, signalant que le reCAPTCHA est actif.



J'ai ensuite fait un autre test qui vérifie la totalité des fichiers présents sur le site et détecte les malwares ainsi que les failles de sécurité éventuelles, il s'agit du site **SiteGuarding.com**. Il effectue aussi une vérification des liens présents sur le site en comparant ces derniers avec une liste de lien « blacklisté ». Aucun problème n'a été constaté lors de ce traitement dont voici un extrait.

Detailed Blacklist Analyze

Global Blacklists	SPAM Blacklists
OK CMC Threat Intelligence	OK ivmURI
OK Snort IP sample list	OK Nordspam DBL
OK VX Vault	OK SEM FRESH
OK Armis	OK SEM URI
OK ViriBack	OK SEM URRED
OK Comodo Valkyrie Verdict	OK SORBS RHSBL BADCONF
OK PhishLabs	OK SORBS RHSBL NOMAIL
OK K7AntiVirus	OK Spamhaus DBL
OK CINS Army	OK SURBL multi
OK Cyren	OK OSPAM
OK Outtera	OK Abuse.ro
OK BlockList	OK Abusix Mail Intelligence Blacklist
OK OpenPhish	OK Abusix Mail Intelligence Domain Blacklist
OK 0xSi_f33d	OK Abusix Mail Intelligence Exploit list
OK Feodo Tracker	OK Anommails DNSBL
OK Web Security Guard	OK BACKSCATTERER
OK Scantian	OK BARRACUDA
OK AlienVault	OK BLOCKLIST.DE
OK Sophos	OK CALVENT
OK PhishTank	OK CYMRU BOGONS
OK EonScope	OK DAN TOR
OK Cyan	OK DAN TOREXIT
OK Spam404	OK DNS SERVICIOS
OK SecureBrain	OK DRMX



Front-End



Mise en place du Framework CSS Bootstrap

Afin de développer plus rapidement la partie Front-End du site, ainsi que de profiter des fonctionnalités qu'il propose, j'ai choisi d'utiliser le framework Bootstrap.

Symfony est conçu plus précisément pour utiliser le framework **Bootstrap**.

Mise en place

Pour que les formulaires prennent le thème de Symfony avec les class= « form-control » et autres, il faut modifier le fichier de configuration de twig qui se trouve dans config/package/twig.yaml comme ci-dessous :

```
twig:  
    default_path: '%kernel.project_dir%/templates'  
    form_themes: ['bootstrap_5_layout.html.twig']  
  
when@test:  
    twig:  
        strict_variables: true
```

Pour mettre en place ce framework, j'intègre les lignes suivantes dans le <head> du template base.html.twig comme ci-dessous :

```
<!-- bootstrap-5.1.3-dist -->  
<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.min.css"  
      rel="stylesheet"  
      integrity="sha384-1BmE4kWBq78iYhFlvKuhfTAU6auU8tT94WrHftjDbrCEXSU1oBoqyl2QvZ6jIW3"  
      crossorigin="anonymous">
```

puis ajouter le script JavaScript suivant avant la fermeture de la balise </body> :

```
<!-- Bootstrap ===== -->  
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/js/bootstrap.bundle.min.js"  
       integrity="sha384-ka7Sk0Gln4gmtz2MlQnikT1wXgYsOg+OMhuP+IlRH9sENB00LRn5q+8nbTov4+1p"  
       crossorigin="anonymous">  
</script>
```

Mise en place des icons

J'ai aussi utilisé les icônes de Font Awesome et pour cela j'ai également ajouté le code JavaScript dans mon <head>

```
<!-- Icons-->
<script src="https://kit.fontawesome.com/4d2daebcf2.js"
        crossorigin="anonymous"></script>
```

Mise en place des polices d'écriture

J'ai utilisé les fonts de Google Fonts, qui est un service d'hébergement gratuit de polices d'écriture pour le web.

Dans le <head> j'ajoute les lignes suivantes :

```
<!-- Fonts-->
<link rel="preconnect" href="https://fonts.googleapis.com">
<link rel="preconnect" href="https://fonts.gstatic.com" crossorigin>
<link
    href="https://fonts.googleapis.com/css2?family=Gelasio:ital,wght@0,400;0,500;0,
    600;0,700;1,400;1,500;1,600;1,700&family=Parisienne&display=swap"
    rel="stylesheet">
```

Mise en place du CSS

J'ai décidé pour le CSS t'utiliser SASS, en partie pour la gestion plus simple du code et de ses imbrications, son système @extend, et son système @import.

Pour la mise en place su SASS, il faut créer un fichier en .scss (styles.scss) et d'utiliser l'extension « Live Sass Compiler » dans Visual Studio Code. La compilation de ce fichier génère le fichier « styles.css ».

Je rappelle que le fichier CSS est lié avec la ligne suivante dans le <head> du fichier base.html.php :

```
<!-- Styles -->
<link rel="stylesheet" href="{{asset('css/styles.css')}}">
```

A titre d'information la fonction « asset » renvoi public, c'est-à-dire pour notre exemple

```
<!-- Styles -->
<link rel="stylesheet" href="public/css/styles.css">
```

Le principe du SASS, est de pouvoir imbriquer les différentes commandes CSS, plus clair pour la programmation.

```
// == FOOTER =====
footer {
  .clIconeEnLigne {
    display: flex;
    flex-direction: row;
    justify-content: center;
  }
  .clFooter {
    padding-top: 2rem;
    padding-bottom: 2rem;
    background-color: rgb(44, 62, 80);
    width: 100%;
    color: white;
  }
  .clCopyright {
    background-color: rgb(26, 37, 47);
    font-size: 70%;
    p {
      margin-top: 10px;
    }
  }
}
```

Réaliser une interface web statique et dynamique

Afin de maximiser la gestion du site, j'ai mis en place la possibilité à l'association de mettre à jour le contenu de pratiquement toutes les pages. L'association peut importer ses photos, mettre du texte, le modifier.

Suite à la connexion d'un utilisateur déclaré en Administrateur, ce dernier a accès à un menu vertical lui donnant la possibilité de modifier l'ensemble des données.

Par exemple, la gestion des actions de l'association, qui apparaît sous forme de tableau où il est possible de modifier les fiches. La possibilité de supprimer une action se fait en passant avant, par l'affichage de cette dernière.

Les Actions de l'Association						
		Nouveau				
Titre	Image Principal	Texte de présentation	Lien	Dates	actions	
Boîte à Insectes		Généralement, on utilise un texte en faux latin (le texte ne veut rien dire, il a été modifié), le Lorem ipsum ou Lipsum, qui permet donc de faire office de texte d'attente. L'avantage de le mettre en latin est que l'opérateur sait au premier coup d'œil que la page contenant ces lignes n'est pas valide, et surtout l'attention du client n'est pas dérangée par le contenu, il demeure concentré seulement sur l'aspect graphique. Ce texte a pour autre avantage d'utiliser des mots de longueur varia500ble, essayant de simuler une occupation normale. La méthode simpliste consistant à copier-coller un court texte plusieurs fois (« ceci est un faux-texte »)750 a l'inconvénient de ne pas permettre une juste appréciation typographique du résultat final. Il circule des centaines de versions différentes du Lorem ipsum, mais ce texte aurait originellement été tiré de l'ouvrage de Cicéron, De Finibus Bonorum et Illorum	Voir un autre exemple...	Début : 01/02/22 Fin : 03/02/22	Afficher	Modifier
Boîte à Livres ou Bibliothèque de rue		Généralement, on utilise un texte en faux latin (le texte ne veut rien dire, il a été modifié), le Lorem ipsum ou Lipsum, qui permet donc de faire office de texte d'attente. L'avantage de le mettre en latin est que l'opérateur sait au premier coup d'œil que la page contenant ces lignes n'est pas valide, et surtout l'attention du client n'est pas dérangée par le contenu, il demeure concentré seulement sur l'aspect graphique. Ce texte a pour autre avantage d'utiliser des mots de longueur varia500ble, essayant de simuler une occupation normale. La méthode simpliste consistant à copier-coller un court texte plusieurs fois (« ceci est un faux-texte »)750	Pour en savoir plus...	Début : 10/03/22 Fin : 24/03/22	Afficher	Modifier
Jardin		Généralement, on utilise un texte en faux latin (le texte ne veut rien dire, il a été modifié), le	photo	Début :		Afficher

Une nouvelle Action, ou la modification, se présente sous cette forme. L'administrateur peut ajouter différentes photos, le texte, et d'autres informations. L'enregistrement se fait par l'action sur bouton Valider.

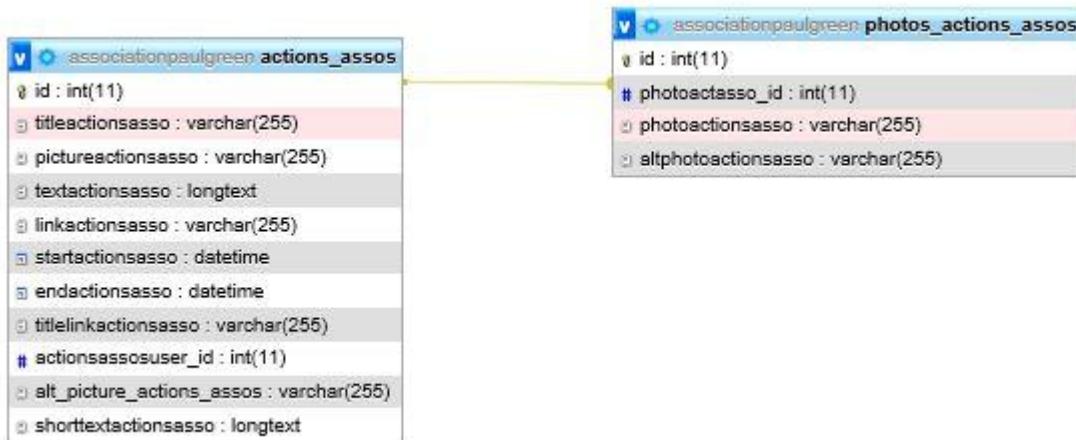
Titre :												
<input type="text" value="Garder à insectes"/>												
Texte de présentation court :												
Generally, on utilise un texte en faux latin (le texte ne veut rien dire, il a été modifié), le Lorem Ipsum ou Lipsum, qui permet donc de faire office de tel normale. La méthode consiste à copier-coller un court texte plusieurs fois (= ceci est un faux-titre cela est un faux-titre cela est un faux-titre)												
Photo de présentation:												
<input type="button" value="Choisir un fichier"/> aucun fichier choisi												
Texte alternatif Photo / sous-Titre :												
Construire et installer un hôtel à insecte												
Texte long :												
<p>Pourquoi vouloir installer un hôtel à insecte ? A quoi cela va-t-il servir ???</p> <p>En général, 80% des insectes sont dépourvus de vol, nécessitant d'installer un hôtel à insecte, afin d'aider les espèces à se déplacer pour trouver un partenaire, protéger d'éventuelles agressions ou échapper aux prédateurs. Les insectes sont également très sensibles aux changements climatiques, nécessitant pour les survie, une certaine stabilité dans leur habitat. Les insectes sont également très sensibles au temps, avec une température supérieure à 30-35 °C pour la mort des 20 mm d'âge.</p> <p>La découpe</p> <p>Les insectes ont une croissance segmentée. Pour l'essai, nous avons choisi de planter en longueur des séquences alternées (10 cm en 8 m de planter).</p> <ul style="list-style-type: none"> - 8 planentes de 22 cm - 4 planentes de 30-35 cm (pour une planente d'âge moyen 22 mm - 30 cm) - 8 planentes de 22-30 cm (d'âge moyen 10-12 cm pour une planente de 22 mm) - 4 planentes de 22 cm (planente avec section d'âge moyen 10-12 cm) - Des trous de 10-12 mm de diamètre et de 20-25 mm pour laisser le bois de 20 mm d'âge. <p>Le montage</p> <p>Date de début :</p> <p>01/02/2022</p> <p>Date de fin :</p> <p>01/02/2022</p> <p>lien du lien d'accès :</p> <p>Voir un autre exemple...</p> <p>lien :</p> <p>http://www.jardin-maintenance.fr/construire-hotel-a-insectes/</p> <p>Thème de l'action:</p> <p><input checked="" type="checkbox"/> Forêt</p> <p><input checked="" type="checkbox"/> Rivières</p> <p><input type="checkbox"/> Insectes</p> <p><input type="button" value="Sélectionner"/> aucun fichier choisi</p> <p>Images:</p> <p></p> <p>Détails de l'hôtel à insectes</p> <p>Photos:</p> <table border="1"> <tr> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> </tr> <tr> <td>Supprimer</td> <td>Supprimer</td> <td>Supprimer</td> <td>Supprimer</td> <td>Supprimer</td> <td>Supprimer</td> </tr> </table>							Supprimer					
Supprimer	Supprimer	Supprimer	Supprimer	Supprimer	Supprimer							
Valider Recadrer Supprimer												



Gestions de photos

Dans la beaucoup de mes entités principales, le thème est exposé avec une photo principale mais les étudiants peuvent ajouter par la suite une multitude de photos rentrant dans des carrousels. Une méthode m'a été proposée par un tutoriel.

Pour notre exemple, nous allons utiliser la table ActionsAssos, et exposer la fonction de modification d'une action de l'association. Le principe est le même pour la partie création.



Dans notre base de données, la relation entre les deux tables est du type « One To Many ». Dans la table regroupant les photos il sera possible ensuite de rajouter le texte alternatif indispensable pour les référencements. La relation entre les deux entités a été définie grâce à la commande « php bin/console make:entity » suivi des commandes de migration précédemment.

Dans le cas où les créations des fichiers de gestion générés par le CRUD ont déjà été réalisées, il faut modifier les différents fichiers « à la main ».

```
<?php

namespace App\Form;

use App\Entity\Themes;
use App\Entity\ActionsAssos;
use Symfony\Component\Form\AbstractType;
use FOS\CKEditorBundle\Form\Type\CKEditorType;
use Symfony\Component\Form\FormBuilderInterface;
```

```

use Symfony\Bridge\Doctrine\Form\Type\EntityType;
use Symfony\Component\Validator\Constraints\File;
use Symfony\Component\OptionsResolver\OptionsResolver;
use Symfony\Component\Form\Extension\Core\Type\DateType;
use Symfony\Component\Form\Extension\Core\Type\FileType;
use Symfony\Component\Form\Extension\Core\Type\TextType;
use Symfony\Component\Form\Extension\Core\Type\TextareaType;

class ActionsAssosType extends AbstractType
{
    public function buildForm(FormBuilderInterface $builder, array $options): void
    {
        $builder
            ->add('titleactionsasso', TextType::class, [
                'label' => 'Titre : ',
                'required'=> true,
                'attr' => [
                    'maxlength' =>'250',
                ]
            ])
            ->add('shorttextactionsasso', TextareaType::class, [
                'label' => 'Texte de présentation court : ',
                'required'=> false,
                'attr' => [
                    'rows' => '10',
                    'maxlength' => '1000'
                ],
            ])
            ->add('pictureactionsasso', FileType::class, [
                'label' => 'Photo de présentation: ',
                'mapped' => false,
                'required' => false,
                'constraints' => [
                    new File([
                        'maxSize' => '1024k',
                        'mimeTypes' => [
                            'image/*',
                        ],
                        'mimeTypesMessage' => 'Veuillez entrer un format de document valide',
                    ])
                ],
            ])
            ->add('altPictureActionsAssos', TextType::class, [
                'label' => 'Texte alternatif Photo / sous-Titre : ',
                'required'=> false,
                'attr' => [

```

```

        'maxlength' =>'250',
    ]
])
->add('textactionsasso', CKEditorType::class, [
    'label' => 'Texte long : ',
])
->add('startactionsasso', DateType::class, [
    'label' => 'Date de début : ',
    'widget' => 'single_text',
    'required' => false,
])
->add('endactionsasso', DateType::class, [
    'label' => 'Date de Fin : ',
    'widget' => 'single_text',
    'required' => false,
])
->add('titlelinkactionsasso', TextType::class, [
    'label' => 'Titre du lien ci-dessous : ',
    'label_attr' => [
        'class' => 'fst-italic text-decoration-underline'
    ],
    'required'=> false,
    'attr' => [
        'maxlength' =>'250',
    ]
])
->add('linkactionsasso', TextType::class, [
    'label' => 'Lien : ',
    'label_attr' => [
        'class' => 'fst-italic'
    ],
    'required'=> false,
    'attr' => [
        'maxlength' =>'250',
    ]
])
->add('actionsassostheme', EntityType::class,[

    'label' => "Theme de l'action:",
    'class' => Themes::class,
    'choice_label' => 'nameTheme',
    'multiple' => True,
    'expanded' => true,
])
->add('actionsassosphoto', FileType::class,[

    'label' => false,
    'multiple' => true,
])

```

```

        'mapped' => false,
        'required' => false,
    ])
;
}

public function configureOptions(OptionsResolver $resolver): void
{
    $resolver->setDefaults([
        'data_class' => ActionsAssos::class,
    ]);
}
}

```

L'ajout des photos se fait par un select multiple de fichiers.

Les photos spécifiques à l'association sont enregistrées dans un répertoire particulier.

Cette information nous allons l'indiquer dans le fichier « config/services.yaml » dans la section « parameters ». Il suffit à chaque besoin pour afficher une photo d'utiliser le paramètre «photos_directoryAssos ».

```

parameters:
    photos_directoryAdherents: '%kernel.project_dir%/public/photosAdherents'
    photos_directoryParams: '%kernel.project_dir%/public/photosParams'
    photos_directoryPartners: '%kernel.project_dir%/public/photosPartners'
    photos_directoryNews: '%kernel.project_dir%/public/photosNews'
    photos_directoryAssos: '%kernel.project_dir%/public/photosActionsAssos'
    photos_directoryMasters: '%kernel.project_dir%/public/photosActionsMasters'
    photos_directoryFieldtrips: '%kernel.project_dir%/public/photosFieldtrips'

```

Dans notre « controller » ActionsAssosController.php, la méthode « edit » ci-dessous, et dans la partie des Photos multiples, nous constatons que la récupération des données se fait par un système de boucle avec « foreach », copie la photo dans un répertoire spécifique et enregistre seulement un nouveau nom généré aléatoirement, dans la base de données.

```

/**
 * @Route("/{id}/edit", name="actions_assos_edit", methods={"GET", "POST"})
 */
public function edit(Request $request,
                    ActionsAssos $actionsAsso,
                    EntityManagerInterface $entityManager,
                    SluggerInterface $slugger,
                    Users $users ): Response

```

```

{

$form = $this->createForm(ActionsAssosType::class, $actionsAsso);
$form->handleRequest($request);

if ($form->isSubmitted() && $form->isValid()) {

    // ===== Photo

    $photo = $form->get('pictureactionsasso')->getData();
    if ($photo) {
        $originalFilename = pathinfo($photo->getClientOriginalName(),
PATHINFO_FILENAME);
        // ceci est nécessaire pour inclure en toute sécurité le nom de fichier dans
l'URL
        $safeFilename = $slugger->slug($originalFilename);
        $newFilename = $safeFilename . '-' . uniqid() . '.' . $photo->guessExtension();
        // Déplacez le fichier dans le répertoire où les brochures sont stockées
        try {
            $photo->move($this->getParameter('photos_directoryAssos'), $newFilename);
        } catch (FileException $e) {
            // ... gérer l'exception si quelque chose se produit pendant le
téléchargement du fichier
        }
        // met à jour la propriété 'photo' pour stocker le nom du fichier au lieu de
son contenu
        $actionsAsso->setPictureactionsasso($newFilename);
    }
    // =====

    // ===== Photos Multiples

    // On récupère les images transmises
    $photos = $form->get('actionsassosphoto')->getData();

    // On boucle sur les images
    foreach($photos as $photo){
        // On génère un nouveau nom de fichier
        $fichier = md5(uniqid()).'.'.$photo->guessExtension();
        // On copie le fichier dans le dossier uploads
        $photo->move(
            $this->getParameter('photos_directoryAssos'),
            $fichier
        );
        // On crée l'image dans la base de données
        $img = new PhotosActionsAssos();

```

```

        $img->setPhotoactionsasso($fichier);
        $actionsAsso->addActionsassosphoto($img);
    }
    // =====

    $actionsAsso->setActionsassosuser($this->getUser());
    $entityManager->flush();
    return $this->redirectToRoute('actions_assos_index', [], Response::HTTP_SEE_OTHER);
}

// -----
// Partie commentaires
// On crée le commentaire "vierge"
$comment = new DocActionsAssos;
// On génère le formulaire
$commentForm = $this->createForm(DocActionsAssosType::class, $comment);
$commentForm->handleRequest($request);
// Traitement du formulaire
if($commentForm->isSubmitted() && $commentForm->isValid()){
    // On récupère le contenu du champ parentid
    $parentid = $commentForm->get("docactasso")->getData();
    // On va chercher le commentaire correspondant
    if($parentid != null){
        $parent = $entityManager
            ->getRepository(DocActionsAssosType::class)
            ->find($parentid);
    }
    return $this->render('actions_assos/edit.html.twig', [
        'actions_asso' => $actionsAsso,
        'form' => $form->createView(),
        'commentForm' => $commentForm->createView()
    ]);
}

```

Pour supprimer les photos, j'ai utilisé une méthode « deleteImage » dans le même controller.

```

/**
 * @Route("/supprime/photo/{id}", name="actions_assos_delete_photo", methods={"DELETE"})
 */
public function deleteImage(Request $request, PhotosActionsAssos $photo,
EntityManagerInterface $entityManager){

    $data = json_decode($request->getContent(), true);

    // On vérifie si le token est valide
    if($this->isCsrfTokenValid('delete' . $photo->getId(), $data['_token'])){


```

```

    // On récupère le nom de l'image
    $nom = $photo->getPhotoactionsasso();
    // On supprime le fichier
    unlink($this->getParameter('photos_directoryAssos').'/'.$nom);

    // On supprime l'entrée de la base
    $entityManager->remove($photo);
    $entityManager->flush();

    // On répond en json
    return new JsonResponse(['success' => 1]);
} else{
    return new JsonResponse(['error' => 'Token Invalide'], 400);
}
}

```

J'associe à cette méthode du JavaScript couplé avec une requête Ajax, avec l'appui sur un bouton. J'ai créé un fichier photos.js que j'ai intégré dans mon projet avec le code suivant :

```

window.onload = () => {
    // Gestion des boutons "Supprimer"
    let links = document.querySelectorAll("[data-delete]")

    // On boucle sur links
    for (link of links) {
        // On écoute le clic
        link.addEventListener("click", function(e) {
            // On empêche la navigation
            e.preventDefault()

            // On demande confirmation
            if (confirm("Voulez-vous supprimer cette photo ?")) {
                // On envoie une requête Ajax vers le href du lien avec la méthode DELETE
                fetch(this.getAttribute("href"), {
                    method: "DELETE",
                    headers: {
                        "X-Requested-With": "XMLHttpRequest",
                        "Content-Type": "application/json"
                    },
                    body: JSON.stringify({ "_token": this.dataset.token })
                }).then(
                    // On récupère la réponse en JSON
                    response => response.json()
                ).then(data => {
                    if (data.success)
                        this.parentElement.remove()
                })
            }
        })
    }
}

```

```

        else
            alert(data.error)
        }).catch(e => alert(e))
    )
}
}
}

```

J'ai intégré le lien à ce fichier avec une balise script.

```
<script src="{{asset('js/photos.js')}}" defer></script>
```

Dans tous les fichiers templates Twig, ou je gère ces photos j'intègre le code par exemple suivant, qui m'affiche chaque photo avec le bouton supprimer :

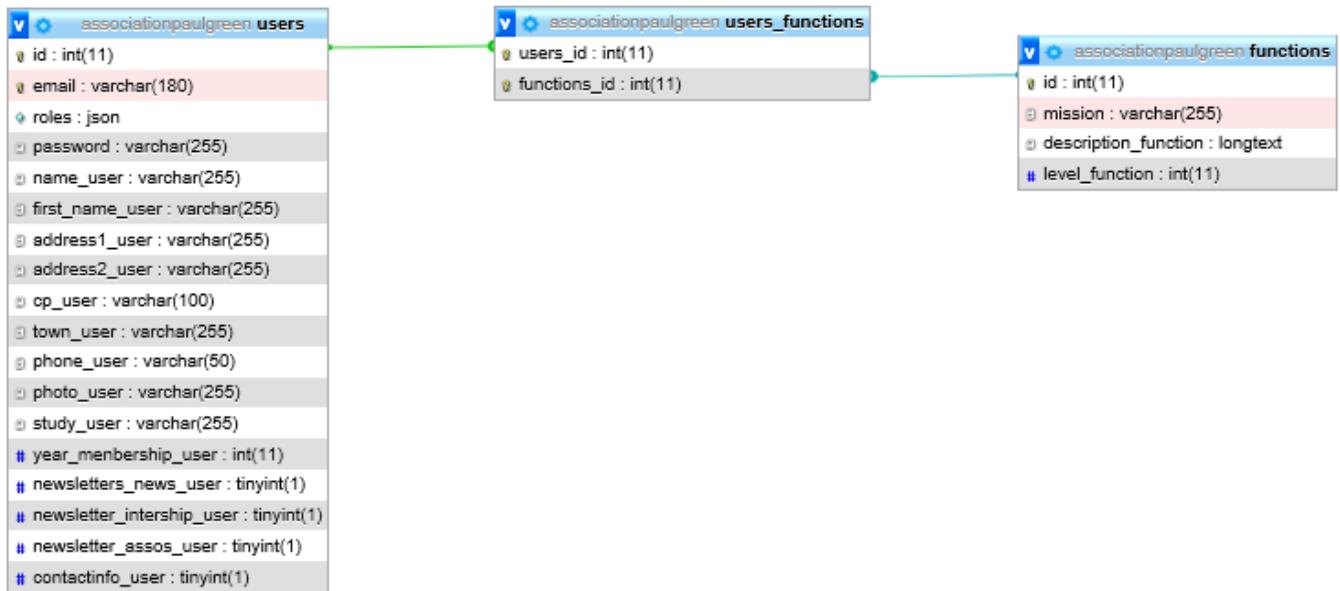
```

<p>Photos</p>
<div class="d-flex flex-row">
    {% for photo in actions_asso.actionsassosphoto %}
        <div class="card me-2" style="width: 150px;">
            
            <div class="card-body p-0 d-flex flex-column justify-content-between">
                <p class="card-text textAlt">{{ photo.altphotoactionsasso }}</p>
                <div class="card-footer">
                    <a href="{{ path('actions_assos_delete_photo', {id: photo.id})}}"
                        data-delete data-token="{{ csrf_token('delete' ~ photo.id )}}"
                        class="btn btn-grad btn-danger2">
                        Supprimer
                    </a>
                </div>
            </div>
        {% endfor %}
    </div>

```

Affichage du Bureau de l'association :

Le but ici est d'afficher un bureau, c'est à dire les Users qui ont une fonction au sein de l'association et suivant un ordre précis défini dans l'entité Functions.



La relation entre les tables users et functions est une « Many To Many », d'où la table pivot users_functions.

J'ai recherché à traduire une requête SQL classique, réalisé avec la partie SQL dans phpMyAdmin, dans le système de requête Symfony. J'ai donc réalisé une fonction dans le repository UsersRepository.php.

```
// SELECT users.*, function.* FROM users
// INNER JOIN users_functions ON users.id=users_functions.users_id
// INNER JOIN functions ON users_functions.functions_id=functions.id
// ORDER BY functions.level_function, users.name_user, users.first_name_user

public function findOffice()
{
    $entityManager = $this->getEntityManager();
    $query = $entityManager->createQuery(
        'SELECT u, f
         FROM App\Entity\Users u
         INNER JOIN u.functionUser f
         ORDER BY f.levelFunction, u.nameUser, u.firstNameUser'
    );

    return $query->getResult();
}
```

Dans le controller, je fais appel à cette fonction pour l'affichage du bureau.

```
/**  
 * @Route("/Office", name="office_index", methods={"GET"})  
 */  
  
public function office(UsersRepository $usersRepository,  
                      ParamsRepository $paramsRepository,  
                      SocialNetworksRepository $socialNetworksRepository ): Response  
{  
    return $this->render('users/office.html.twig', [  
        'office' => $usersRepository->findOffice(),  
        'params' => $paramsRepository->findAll(),  
        'social_networks' => $socialNetworksRepository->findAll(),  
    ]);  
}
```

Puis j'ai créé un Template pour afficher les membres du bureau avec ce code ci-dessous :

```
<div class="container clBureau">  
    <h1 class="h-grad bg-blueGreen">Les membres du Bureau</h1>  
    <hr class="mb-3">  
    <div class="row row-cols-1 row-cols-md-3 g-4">  
        {% for user in office %}  
            <div class="col">  
                <div class="card text-center ombre">  
                    <div class="d-flex justify-content-center">  
                        {% if user.photoUser %}  
                              
  
                        {% else %}  
                              
                        {% endif %}  
                    </div>  
                    <div class="card-body d-flex flex-column align-items-center">  
                        <h5 class="card-title">{{ user.nameUser }} {{ user.firstNameUser }}</h5>  
                        <a class="clRond"  
                            href="mailto:{{ user.email }}  
                            ?subject=Sujet du mail...  
                            &cc={{ params[0].emailParam }}>
```

```

&body=Bonjour {{ user.nameUser }} {{ user.firstNameUser }}

...
"
    rel="" title=""
<i class="fas fa-envelope"></i>
</a>
<a class="card-text text-decoration-none"
  href="mailto:{{ user.email }}"
  ?subject=Sujet du mail...
  &cc={{ params[0].emailParam}}
  &body=Bonjour {{ user.nameUser }} {{ user.firstNameUser }}

...
"
    rel="" title=""
<i class="fas fa-envelope"></i> {{ user.email }}
</a>
<br>
<a href="tel:+{{ user.phoneUser }}" class="card-text text-
decoration-none">
    <i class="fas fa-phone-square-alt"></i> {{ user.phoneUser }}
</a>
</div>
<div class="card-footer">
    {% set xFonction = null %}
    {% for xFct in user.functionUser %}
        {% if xFonction is null %}
            {% set xFonction = xFct.mission %}
        {% else %}
            {% set xFonction = xFonction ~ ' / ' ~ xFct.mission %}
        {% endif %}
    {% endfor %}
    <small class="text-muted">{{ xFonction }}</small>
</div>
</div>
</div>
{% else %}
    <p>Aucun membre du Bureau</p>
    {% endfor %}
</div>
<hr class="mb-3">
<a class="btn btn-grad btn-secondary2 me-2" href="{{ path('users_index') }}>Retour</a>
<hr class="mb-3">
</div>

```



Mise en Production



Hébergement

Le projet étant toujours en développement, pour les besoins de l'examen, le site a été installé sur Arinfo par notre formateur.

O2Switch est l'hébergeur français utilisé par Arinfo.

Afin de transférer les données locales de l'ordinateur, le client SFTP (SSH File Transfer Protocol) a été utilisé.

WinSCP.

The screenshot shows the WinSCP interface with two panes. The left pane shows the local directory structure on C:\Jacques\Projet\AssociationPaulGreen\, including subfolders like .git, Base, bin, config, DocumentationsProjet, migrations, public, src, templates, tests, translations, var, and vendor, along with various configuration files such as .env, .env.local, .env.test, .gitignore, .phpunit.result.cache, composer.json, composer.lock, docker-compose.ove..., docker-compose.yml, phpunit.xml.dist, README.md, and symfony.lock. The right pane shows the remote directory structure on jlemoine@jlemoine.asnprojets.fr@ftp.boel7182.odns.fr, which is identical to the local structure. Both panes include columns for Nom (Name), Taille (Size), Date de modification (Last modified), Droits (Permissions), and Pro (Properties). The status bar at the bottom indicates 0 B de 409 KB dans 0 de 25 on the left and 0 B de 410 KB dans 0 de 25 on the right, with a total transfer time of 0:02:22.

Après avoir importé la base de données via cPanel, le fichier '.env' est modifié pour la nouvelle base :

```
DATABASE_URL=mysql://boel7182:'fpHB4x-  
DxUJI'@109.234.162.86:3306/boel7182_jlemoine
```

Le fichier '.htaccess' a été ajouté à la racine de mon dossier public afin de gérer les redirections.

Optimisation des performances et Référencement

Lighthouse



Afin d'évaluer les performances du site internet, j'ai utilisé l'outil Google Lighthouse, qui permet de mesurer la qualité des pages web.

Avec le premier test, nous avons les résultats suivants :



Values are estimated and may vary. The [performance score is calculated](#) directly from these metrics. [See calculator.](#)

▲ 0–49 ■ 50–89 ● 90–100



METRICS

[Expand view](#)

■ First Contentful Paint

2.9 s

▲ Time to Interactive

8.4 s

■ Speed Index

4.7 s

● Total Blocking Time

0 ms

■ Largest Contentful Paint

3.7 s

● Cumulative Layout Shift

0.097

[View Treemap](#)



Show audits relevant to: [All](#) [FCP](#) [TBT](#) [LCP](#) [CLS](#)

Il est important de lancer des axes de progrès du site afin d'améliorer ces évaluations.

SEO (Search Engine Optimisation) ou Optimisation pour les moteurs de recherche. :

Pour améliorer le référencement, j'ai complété les informations de métadonnées dans le 'head' de ma page base.html.twig.

```
<!DOCTYPE html>
<html lang="fr">

<html>
    <head>
        <meta charset="UTF-8">
        <meta http-equiv="X-UA-Compatible" content="IE=edge">
        <meta name="viewport" content="width=device-width, initial-scale=1.0">
        <meta name="description"
            content="{{render(controller('App\\Controller\\MainController::paramRecup'))}} &
            " || Campus BIODIV+ || Faire de l'Université Paul Valéry un campus à
biodiversité positive" &
            " || Université Paul Valéry Montpellier 3 || Master Gestion de l'Environnement"
        <title>
            {% block title %}
                {{render(controller('App\\Controller\\MainController::paramRecup'))}}
            {% endblock %}
        </title>
```

J'ai également pris soin de remplir l'attribut de texte alternatif 'alt' de l'ensemble des images du site. En sachant que les images rentrées par les utilisateurs pour les différentes entités, il y a la possibilité de saisir eux même les textes alternatifs.

En exemple :

```
<div class="d-flex justify-content-center">
    
</div>
```

Recherche en Anglais

Dans mes recherches pour résoudre des problèmes, pour trouver des bouts de code et autres, j'ai effectué des recherches et nombreux les résultats trouvés sont en langue anglaise. Tout d'abord les documentations Symfony et du Twig, CKEditor, et autres sont en anglais.

Dans le cadre de l'amélioration des performances, ayant entendu parler du « loading lazy » pendant mon stage, je me suis mis à la recherche d'informations et je suis arrivé sur des explications en anglais avec le paragraphe suivant :

Images and iframes

Very often, webpages contain many images that contribute to data-usage and how fast a page can load. Most of those images are off-screen (non-critical), requiring user interaction (an example being scroll) in order to view them.

Loading attribute The loading attribute on an element (or the loading attribute on an <iframe>) can be used to instruct the browser to defer loading of images/iframes that are off-screen until the user scrolls near them.

```

<iframe src="video-player.html" title="..." loading="lazy"></iframe>
```

The load event fires when the eagerly-loaded content has all been loaded; at that time, it's entirely possible (or even likely) that there may be lazily-loaded images that are within the visual viewport that haven't yet loaded.

Pour la traduire en français :

Images et Iframes

Très souvent, les pages Web contiennent de nombreuses images qui contribuent à l'utilisation des données et à la vitesse de chargement d'une page. La plupart de ces images sont hors écran (non-critique), nécessitant une interaction de l'utilisateur (un exemple étant le scroll ou défilement) afin de les voir ou de les visualisés.

Attribut de chargement L'attribut loading sur l'élément (ou l'attribut loading sur un <iframe>) peut être utilisé pour demander au navigateur de différer le chargement des images / iframes qui sont hors écran jusqu'à ce que l'utilisateur les fasse défiler.

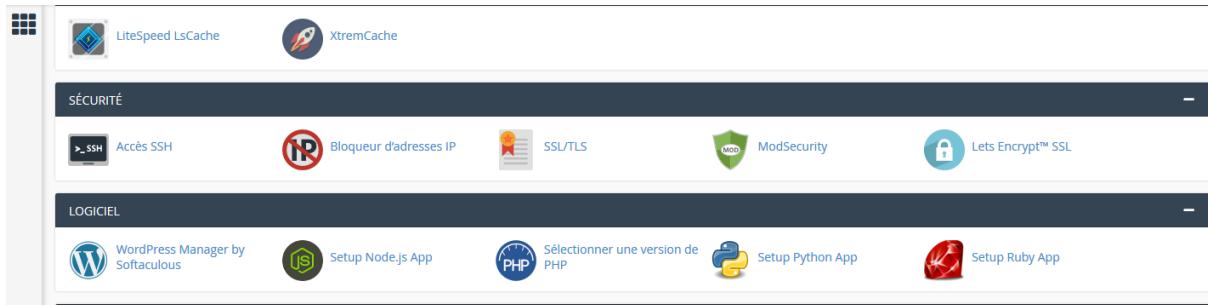
```

<iframe src="video-player.html" title="..." loading="lazy"></iframe>
```

L'événement de chargement se déclenche lorsque le contenu chargé avec « empressement » à été chargé; à ce moment-là, il est tout à fait possible (ou même probable) qu'il y ait des images chargées « paresseusement » qui se trouvent dans l'écran visible qui ne sont pas encore chargées.

Bonnes Pratiques

A partir du panneau de configuration de cPanel, pour augmenter la sécurité du site, et améliorer son référencement, l'application est configuré avec le protocole ‘https’.



The screenshot shows the cPanel interface. At the top, there are icons for LiteSpeed LSCache and XtreMCache. Below that is a 'SÉCURITÉ' section with icons for Accès SSH, Bloqueur d'adresses IP, SSL/TLS, ModSecurity, and Lets Encrypt™ SSL. Under 'LOGICIEL', there are links for WordPress Manager by Softaculous, Setup Node.js App, Sélectionner une version de PHP, Setup Python App, and Setup Ruby App.

Vos domaines avec des certificats Let's Encrypt

Vos domaines avec des certificats Let's Encrypt						
Show 10 entries	Search: <input type="text"/>					
Nom de domaine	Hôtes alternatifs	État	Validation	Expiration	Actions	
compagniedesjeux.emmanuel-mahe-developpeur.fr	www.compagniedesjeux.emmanuel-mahe-developpeur.fr	Installé	http-01	02 Jun 2021	Supprimer	Réinstaller Voir
emmanuel-mahe-developpeur.fr	mail.emmanuel-mahe-developpeur.fr www.emmanuel-mahe-developpeur.fr	Installé	http-01	10 May 2021	Supprimer	Réinstaller Voir

Showing 1 to 2 of 2 entries

Previous 1 Next

Générer un nouveau certificat

Choisissez parmi l'un de vos noms de domaine ci-dessous. Une nouvelle clé et un nouveau certificat seront rajoutés au gestionnaire SSL/TLS.

Générer un nouveau certificat		
Show 10 entries	Search: <input type="text"/>	
Nom de domaine	Hôtes alternatifs	Actions
aghs3607.odns.fr	aghs3607.odns.fr, mail.aghs3607.odns.fr, www.aghs3607.odns.fr,	Générer
compagniedesjeux.emmanuel-mahe-developpeur.fr	compagniedesjeux.emmanuel-mahe-developpeur.fr, www.compagniedesjeux.emmanuel-mahe-developpeur.fr,	Générer

Cookies/RGPD

Tarte au citron

Pour pouvoir gérer les cookies sur mon site, j'ai mis en place le système de gestion appelé ‘Tarte au citron’.

Après avoir téléchargé les fichiers nécessaires et les avoir inclus dans mon projet, j'ai copié ces scripts qui me permettent notamment de gérer les options de ce système :

```
{# ===== Tarteaucitron ===== #}
<script type="text/javascript" src="{{ asset('tarteaucitron/tarteaucitron.js') }}></script>
<script>
    tarteaucitron.init({
        "privacyUrl": "", /* URL de la page de la politique de vie privée */
        "hashtag": "#tarteaucitron", /* Ouvrir le panneau contenant ce hashtag */
        "cookieName": "tarteaucitron", /* Nom du Cookie */
        "orientation": "bottom", /* Position de la bannière (top - bottom) */
        "showAlertSmall": false, /* Voir la bannière réduite en bas à droite */
        "cookieslist": true, /* Voir la liste des cookies */
        "adblocker": false, /* Voir une alerte si un bloqueur de publicités est
détecté */
```

```

    "AcceptAllCta": true, /* Voir le bouton accepter tout (quand highPrivacy est à
true) */

    "highPrivacy": true, /* Désactiver le consentement automatique */
    "handleBrowserDNTRequest": false, /* Si la protection du suivi du navigateur est
activée, tout interdire */
    "removeCredit": false, /* Retirer le lien vers tarteaucitron.js */
    "moreInfoLink": true, /* Afficher le lien 'voir plus d'infos' */
    "useExternalCss": false, /* Si false, tarteaucitron.css sera chargé*/
    // "cookieDomain": '.my-multisite-domaine.fr', /* Cookie multisite */
    "readmoreLink": "", // '/cookiespolicy' /* Lien vers la page 'Lire plus' */
    // =====
    "showIcon": false, /* Show cookie icon to manage cookies */

});

</script>

```

Ceci permet d'afficher la bandeau sur les cookies au démarrage du site.



RGPD

Depuis le bas de page du site (Footer), l'utilisateur peut accéder à une page de mentions légales. Celle-ci est entièrement modifiable et adaptable par les administrateurs.

Association Paul Green

Accueil Association - Master - Créeation d'un compte Se Connecter Contacts A Propos

Informations légales

1. Présentation du site.

En vertu de l'article 6 de la loi n° 2004-575 du 21 juillet 2004 pour la confiance dans l'économie numérique, il est précisé aux utilisateurs du site www.AssociationPaulGreen.fr l'identité des différents intervenants dans le cadre de sa réalisation et de son suivi :

Propriétaire : Association Paul Green - Association loi 1901 - MONTPELLIER
Créateur : Association Paul Green
Responsable publication : Association Paul Green - <xxxxxxxxxxxx@xxxx.fr>
Le responsable publication est une personne physique ou une personne morale.
Webmaster : Jacob - <jacobxxxx@xxxxxx.fr>
Hébergeur : <xxxxxxxxxxxx@xxxxxx.fr> France

Credits :
Le modèle de mentions légales est offert par Subdellium.com [Générateur de mentions légales](#)

2. Conditions générales d'utilisation du site et des services proposés.

L'utilisation du site www.AssociationPaulGreen.fr implique l'acceptation pleine et entière des conditions générales d'utilisation ci-après décrites. Ces conditions d'utilisation sont susceptibles d'être modifiées ou complétées à tout moment, les utilisateurs du site www.AssociationPaulGreen.fr sont donc invités à les consulter de manière régulière.

Ce site est normalement accessible à tout moment aux utilisateurs. Une interruption pour raison de maintenance technique peut être toutefois décidée par La comédie Ponthotchatalane, qui s'efforce alors de communiquer préalablement aux utilisateurs les dates et heures de l'intervention.

Le site www.AssociationPaulGreen.fr est mis à jour régulièrement par La comédie Ponthotchatalane. De la même façon, les mentions légales peuvent être modifiées à tout moment ; elles s'imposent néanmoins à l'utilisateur qui est invité à s'y référer le plus souvent possible afin d'en prendre connaissance.

3. Description des services fournis.

Le site www.AssociationPaulGreen.fr a pour objectif de fournir une information concernant l'ensemble des activités de la société.

L'Association Paul Green s'efforce de fournir sur le site www.AssociationPaulGreen.fr des informations aussi précises que possible. Toutefois, il ne pourra être tenu responsable des omissions, des inexactitudes et des carences dans la mise à jour, qu'elles soient de son fait ou du fait des tiers partenaires qui lui fournissent ces informations.

Tous les informations indiquées sur le site www.AssociationPaulGreen.fr sont données à titre indicatif, et sont susceptibles d'évoluer. Par ailleurs, les renseignements figurant sur le site www.AssociationPaulGreen.fr ne sont pas exhaustifs. Ils sont donnés sous réserve de modifications ayant été apportées depuis leur mise en ligne.

4. Limitations contractuelles sur les données techniques.

Le site utilise la technologie JavaScript.

Le site Internet ne pourra être tenu responsable de dommages matériels liés à l'utilisation du site. De plus, l'utilisateur du site s'engage à accéder au site en utilisant un matériel recent, ne contenant pas de virus et avec un navigateur de dernière génération mis à jour.

5. Propriété intellectuelle et contrefaçons.

L'Association Paul Green est propriétaire des droits de propriété intellectuelle ou détient les droits d'usage sur tous les éléments accessibles sur le site, notamment les textes, images, graphismes, logo, tömes, sons, logotypes.

Toute reproduction, représentation, modification, publication, adaptation de tout ou partie des éléments du site, quel que soit le moyen ou le procédé utilisé, est interdite, sauf autorisation écrite préalable de : L'Association Paul Green .

Toute exploitation non autorisée du site ou de l'un quelconque des éléments qu'il contient sera considérée comme constitutive d'une contrefaçon et poursuivie conformément aux dispositions des articles L.335-2 et suivants du Code de Propriété Intellectuelle.

6. Limitations de responsabilité.

L'Association Paul Green ne pourra être tenue responsable des dommages directs et indirects causés au matériel de l'utilisateur, lors de l'accès au site www.AssociationPaulGreen.fr, et résultant soit de l'utilisation d'un matériel ne répondant pas aux spécifications indiquées au point 4, soit de l'apparition d'un bug ou d'une incompatibilité.

L'Association Paul Green ne pourra également être tenue responsable des dommages indirects (tels par exemple qu'une perte de marché ou perte d'une chance) consécutifs à

Intégration de Google Analytics

Google Analytics est un service gratuit d'analyse d'audience d'un site Web ou d'applications.

Pour profiter de ce service, il faut posséder un compte Google et s'inscrire le site Projet sur Google Analytics.

ADMINISTRATION 	UTILISATEUR Détails du compte Nom du compte (obligatoire) Les comptes peuvent contenir plus d'un ID de suivi. <input type="text" value="E-mail-syndic"/> Paramètres de partage des données du compte  Les données que vous recueillez, traitez et stockez à l'aide de Google Analytics ("Données Google Analytics") sont sécurisées et restent confidentielles. Elles peuvent servir aussi bien à assurer la maintenance et la protection du service Google Analytics qu'à effectuer des opérations critiques au niveau du système. À de rares exceptions, nous pouvons également être utilisées à nos fins légales, conformément à nos Règles de confidentialité.
Les options de partage des données vous permettent de mieux contrôler le partage de vos données Google Analytics. En savoir plus	
<input checked="" type="checkbox"/> Produits et services Google Partagez vos données Google Analytics avec Google afin de nous aider à améliorer nos produits et services. Si vous activez ce paramètre, Google Analytics sera en mesure de fournir les meilleures services possibles en matière de renseignements et d'insight, d'améliorer les services de détection des spams qui profitent à tous les produits et les utilisateurs, et d'offrir des rapports améliorés sur les données démographiques et les contrats d'intérêt (si les départs Google sont activés). Si vous désactivez cette option, des connexions seront toutefois maintenues aux autres produits Google associés à votre propriété. Pour consulter ou modifier nos paramètres, accédez à la section Fonctionnalités de produits dans chaque propriété .	
<input checked="" type="checkbox"/> Analyse comparative Apportez votre contribution en nous envoyant des données anonymes qui seront comparées à celles d'autres utilisateurs. Ces données permettront d'activer des fonctionnalités (telles que l'analyse comparative et les publications) qui vous aideront à mieux comprendre les tendances relatives aux données. Toutes les informations permettant d'identifier votre site seront supprimées et seront combinées à d'autres données anonymes avant d'être partagées avec d'autres utilisateurs. Afficher l'exemple	
<input checked="" type="checkbox"/> Assistance technique Autorisez les représentants de l'assistance technique Google à accéder, si nécessaire, à vos données et à votre compte Google Analytics afin de résoudre des problèmes techniques.	
<input checked="" type="checkbox"/> Account Specialists Autorisez les spécialistes Google du marketing et de la vente à accéder à vos données et à votre compte Google Analytics afin qu'ils puissent trouver des moyens d'améliorer votre configuration et vos analyses, et vous apporter leurs conseils d'optimisation. Si vous n'êtes pas en contact avec des spécialistes dédiés, accordez给他们 accès aux représentants Google directement.	

Cette inscription est suivie d'une phase de configuration, en fonction des objectifs du site internet et de l'identité du destinataire du site.

Pour finaliser la configuration il faut intégrer le code ci-dessous dans le <head>.

× Détails du flux Web

The screenshot shows the 'Détails du flux Web' (Web Flow Details) section in Google Tag Manager. It displays the following information:

- URL DE FLUX:** https://jlemoine.asnprojets.fr
- NOM DU FLUX:** AssociationPaulGreen
- ID DE MESURE:** G-HJBPYR1GB9
- ID DE FLUX:** 3468151556
- ÉTAT:** Aucune donnée reçue au cours des dernières 48 heures. [En savoir plus](#)
- Mesures améliorées:** Mesurez automatiquement les interactions et le contenu sur vos sites en plus des pages vues. This option is checked.
- Mesure:** Pages vues, Défilements, Clics sortants, + 3 autre(s)

```
{# ===== Global site tag (gtag.js) - Google Analytics ===== #
<script async src="https://www.googletagmanager.com/gtag/js?id=G-HJBPYR1GB9"></script>
<script>
  window.dataLayer = window.dataLayer || [];
  function gtag(){dataLayer.push(arguments);}
  gtag('js', new Date());

  gtag('config', 'G-HJBPYR1GB9');
</script>
```

Google Analytics nous permet alors, via un tableau de bord personnalisable, de consulter et de pouvoir analyser l'activité de notre site.

The screenshot shows the Google Analytics dashboard with several reports displayed:

- Accueil (Home):** Shows real-time traffic, engagement, monetization, and acquisition metrics. It includes a chart of new users over the last 14 days and a list of top pages.
- Analyses (Analyses):** Shows the source of new users, top campaigns, and visitor distribution by country.
- Temps réel (Real-time):** Shows active users over time, user retention, and a list of users currently active.
- Acquisition (Acquisition):** Shows the breakdown of users by acquisition channel and specific campaign details.
- Engagement (Engagement):** Shows the number of events per session and conversion rates.
- Monétisation (Monetization):** Shows revenue and average order value.
- Fidélisation (Retention):** Shows user retention over time.
- Données démographiques (Demographic data):** Shows user demographics.
- Technologie (Technology):** Shows device and browser usage.
- Événements (Events):** Shows event data.
- Conversions (Conversions):** Shows conversion data.
- Analyses (Analyses):** Shows the top products sold.

Le site n'étant en ligne depuis très peu de temps les résultats sont bien sûr non significatifs.



Ce projet a été pour moi une expérience enrichissante et a répondu à mes attentes à plusieurs égards :

D'abord, il m'a permis d'avoir une approche concrète des métiers du web en particulier celui de développeur web et web mobile. En effet, la prise d'initiative, le respect des délais et l'opportunité de travailler en autonomie sur un projet et d'être force de proposition sont des aspects essentiels de ce métier que j'ai pu mettre en œuvre à travers ce projet.

De plus, il m'a permis d'appliquer d'une façon concrète les connaissances théoriques et pratiques vues lors de la formation et de réaliser la conception d'un site réel dans sa globalité. Travailler sur un projet de création de site de bout en bout et maîtriser chaque étape de production m'a permis de rapidement progresser et également de voir beaucoup plus clair vis à vis d'une conception similaire dans le monde professionnel.

J'ai également pu prendre conscience des difficultés et des contraintes de la conception d'un site et j'ai pris un réel plaisir à rechercher des solutions et des améliorations nécessaires. J'ai aussi appris qu'il faut faire preuve de flexibilité afin de s'adapter à toutes les demandes du donneur d'ordre pour satisfaire à ses exigences.

Je peux donc affirmer que ce projet m'a beaucoup apporté et a été un travail agréable car le résultat est concret et en bonne voie de finalisation.

Cependant, une formation telle que celle que je viens de suivre est intense, et courte pour acquérir des connaissances complètes, pour concevoir et livrer un tel site aussi dense en contenu. J'aurais aimé améliorer certaines fonctionnalités et en intégrer de nouvelles comme des recherches de documents, la validation de l'inscription par email..., et aussi travailler mieux le design.

Je voulais la réalisation d'un site réel et cette recherche n'a pas été facile, le choix définitif du sujet s'est fait peu de temps avant le début des délais qui nous étaient accordés pour la conception.

Le projet de site de l'association était ambitieux avec des demandes nombreuses de la part de l'Association Paul Green et il était difficile de collecter toutes les informations de la part des étudiants qui étaient plus ou moins disponibles. De plus, en début de projet, la demande manquait réellement de précisions et il a fallu les relancer plusieurs fois pour avoir des éléments concrets . La compréhension des demandes était primordiale pour pouvoir les adapter dans le cadre du développement.

Si ce site a une raison concrète à son existence, faire connaître une association et l'aider à recruter de nouveaux membres, il était avant tout le projet qui clôturait ma formation, et donc un

moyen pour pratiquer et acquérir des connaissances d'un framework aussi populaire et polyvalent que Symfony.

Le projet est encore en phase de réalisation. Les activités de l'association sont à l'arrêt pendant cette période de stage pour les étudiants du Master GE et il est difficile pour ses membres de s'organiser afin de produire le contenu qui agrémentera au final le site pour le moment. Cependant, il est déjà fonctionnel et continuera à évoluer vers la livraison définitive.

L'association Paul Green apprécie mon dévouement pour leur projet et mon investissement dans la poursuite pour la finalisation du site au-delà de la formation à laquelle je me suis engagé afin de leur apporter des améliorations et fonctionnalités plus développées.

Alors que j'arrive à l'issue de ma formation, je suis d'ores et déjà à la recherche d'un emploi, tout en continuant à me former de manière personnelle. Le métier de développeur Web présente l'avantage de laisser à portée de clic toutes les ressources et les aides que l'on peut avoir besoin si l'on souhaite se perfectionner ou apprendre de nouveaux langages et de nouveaux outils. J'ai notamment la volonté et l'espoir d'approfondir mes connaissances, entre autres, dans l'apprentissage du langage Java, par une formation complémentaire spécifique.



Annexe 1 : Définition de Projet fourni par les étudiants de l'association

Création d'un site Internet pour l'association étudiante Paul Green

Paul Green est une association étudiante située à l'Université Paul Valéry Montpellier 3. Elle a vocation, d'une part, à soutenir le master Gestion de l'Environnement (GE) et assurer la cohésion entre les différents étudiants des deux promotions. Elle a pour objectif, d'autre part, de porter des projets associatifs sur des sujets environnementaux et écologiques, et ce, auprès de la communauté universitaire.

L'association est ainsi constituée des étudiants du master GE ainsi que de tout autre étudiant souhaitant y adhérer. Ces derniers ont un statut particulier, ne leur permettant pas d'accéder à certains aspects de l'association (comme les fonds financiers générés par les projets collectifs réalisés dans le cadre du master). L'association souhaite animer et mettre à jour un site Internet dédié à son activité et son champ d'action d'année en année. Il doit donc être facile d'utilisation afin que la prise en main par les nouveaux membres du bureau puisse se faire sans perte de temps et sans compétence spécifique en informatique. De cette manière, le site devra permettre une modification libre par les membres ayant droit.

Objectifs du site pour le master GE et les étudiants :

- Rendre les informations de l'association et du master GE accessibles et permettre la compréhension des actions menées par le plus grand nombre.
- Permettre aux adhérents d'accéder à des informations précises selon leurs statuts. Il ne s'agira pas que d'un site vitrine.
- Assurer de manière pérenne la transmission des supports, données et informations de l'association aux nouvelles promotions du master et aux nouveaux membres de l'association année après année.

Éléments de mise en oeuvre du site Internet :

Le site vitrine

- Présentation de l'association : ses actions (passées, présentes et futures), ses membres.
- Présentation du master GE.
- Présentation des actions menées par l'association.
- Un espace "Agenda" afin de présenter les actions programmées, les rendez vous de l'association...
- Un espace "Contact", présentant les coordonnées utiles pour joindre l'association (notamment celles du Président et l'adresse mail générique de l'association) et qui redirigera directement l'internaute vers les réseaux sociaux.
- Un espace présentant nos engagements et partenaires (exemple : L'Université Paul Valéry ou le RESES : Réseau Étudiant pour une Société Écologique et Solidaire).
- Un espace dédié aux actualités environnementales : possibilité de tenir un "journal" de l'environnement mis à jour régulièrement, permettant de promouvoir les actions locales pour l'environnement.

L'espace "adhérents"

- Mettre en place trois types de comptes "Adhérents" pour lesquels la connexion via les identifiants se fera automatiquement.

Exemple : un adhérent simple aura automatiquement des accès bloqués par rapport à un membre du bureau sans pour autant s'être connecté à un espace en particulier.

1- Les **adhérents membres du bureau** de l'association, qui auront accès aux diverses données financières et auront la possibilité de les modifier/ajouter. Les membres du bureau pourront accéder aux données concernant les différents membres de l'association, gérer les statuts, gérer les paramètres des comptes adhérents, gérer les droits de ceux-ci...

2- Les **adhérents simples membres du master GE**, pour lesquels les informations financières ne seront pas accessibles mais pour lesquels le reste des informations concernant le master seront disponible (offres de stage, anciennes promotions, projets collectifs...)

3- Les **adhérents simples non membres du master GE** pour lesquels toutes les informations concernant l'association autre que celles concernant le Master seront disponibles.

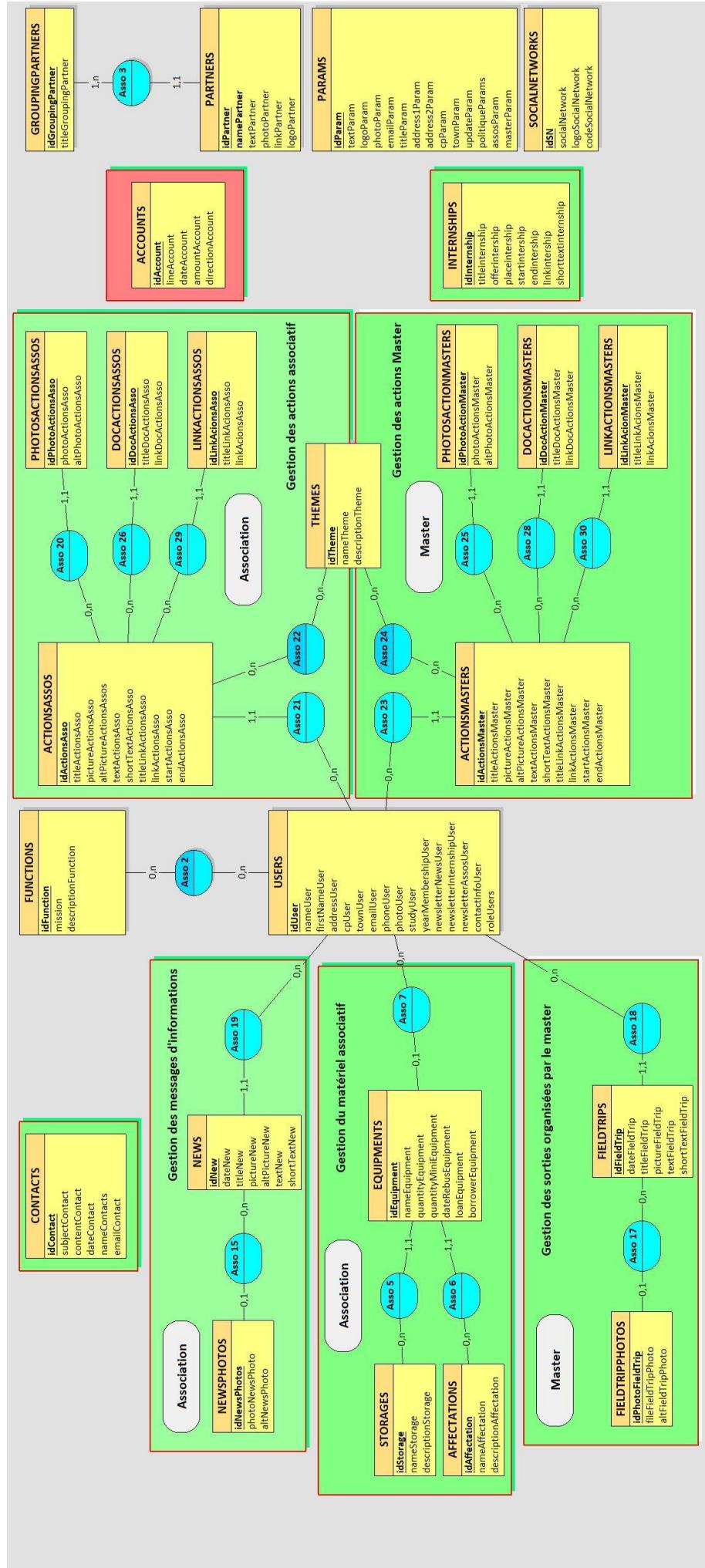
- Mettre en ligne la possibilité d'y stocker les anciens travaux réalisés dans le cadre des projets collectifs du master GE ainsi que les données brutes qui y sont liées (exemple : dossiers, supports...). Il s'agit généralement de documents assez lourds en termes de stockage.
- Possibilité de partager des offres de stage pour les membres du bureau et les étudiants du master GE.
- Possibilité de tenir un inventaire des divers matériels détenus par l'association, notamment ceux du jardin de permaculture afin d'en assurer un suivi précis et des renouvellements optimaux.
- Possibilité de mettre en place un système "d'alerte" que tout adhérent pourrait déployer afin de signaler des éléments nouveaux mis en ligne sur le site (volonté de développer un projet, problème...). Ces alertes permettraient de coordonner les membres de l'association. Si possible, ces alertes pourraient être liées à l'adresse mail des membres afin que chacun d'entre eux soient alertés par mail lorsqu'une nouvelle alerte est proposée.

Exemple de sites :

Ancien Site du Master : <https://univ-verte.asso-web.com>

Association le Globe : <https://afneg.org/lafneg/associations/le-globe-montpellier/>:

Annexe 2 : MCD



Annexe 3: MLD

