



Développeur web & consultant digital mairie des Belleville (73440)



Résumé

Dans ce rapport je présente l'organisme dans lequel j'ai travaillé durant 10 semaines. Je détaille l'histoire de la commune, son organisation mais aussi ses problématiques. Je présente les difficultés que j'ai rencontrées durant la crise sanitaire, mais aussi les solutions qui ont été trouvées.

Ensuite, j'expose les missions qui m'ont été confiées. Je détaille la rédaction de documents propres à la communication sur les réseaux sociaux. Dans un second temps, je présente mon travail de développeur web. Il est question des moyens techniques que j'ai mis en œuvre pour répondre à la problématique, ainsi que de ma méthode de programmation. Je détaille le fonctionnement du code ce qui permet de rendre compte de ma démarche. Dans un troisième temps, j'expose les fonctionnalités les plus innovantes de ce site internet.

Enfin, j'effectue une analyse financière et humaine de mon travail. Je conclus sur les difficultés rencontrées mais aussi sur les enseignements que je tire de cette expérience.

Resume

In this report I present the municipality in which I made my internship during 10 weeks. I detail municipality's history, its organisation but also these problematics. I present the difficulties I encountered during the COVID crisis, but also the solutions that were found.

Then, I expose my tasks. I detail social network communication documents rédaction. In a second way, I present my developper web work. I speak about technicals ways used in order to respond to the problematic, and my programming method. I detail code operation which helps to understand my approach. In a third way, I expose some most innovating fonctionnalités.

Finally, I make a financialy and human analyse. I conclude about difficulties encountered, but also about what I have learned during my internship.

Resumen

En este informe presento el municipio en el que hice mi pasantía durante 10 semanas. Detallo la historia del municipio, su organización, pero también las problemáticas. Presento las dificultades que encontré durante la crisis de COVID, pero también las soluciones que se encontraron.

Luego, expongo mis tareas. Detallo la redacción de documentos de comunicación de las redes sociales. En segundo lugar, presento mi trabajo de desarrollador web. Hablo de las técnicas utilizadas para responder a la problemática, y mi método de programación. Detallo el funcionamiento del código que ayuda a entender mi enfoque. En una tercera forma, expongo algunas de las funcionalidades más innovadoras.

Finalmente, hago un análisis financiero y humano. Concluyo sobre las dificultades encontradas, pero también sobre lo que he aprendido durante mi pasantía.



Remerciements

Je remercie toute l'équipe de la mairie des Belleville pour m'avoir fait confiance durant mon stage. Je leur suis reconnaissant d'avoir accepté le télétravail ainsi que d'avoir maintenu la durée du stage à dix semaines. Je tiens à remercier tout particulièrement Valérie Hudry qui a consacré beaucoup de temps à me guider dans la réalisation du site internet, j'aimerais aussi la remercier pour l'offre d'embauche de trois semaines à l'issue du stage. Cela, afin que je puisse continuer et terminer le développement de leur plateforme internet. Merci de l'opportunité que cela représente.

Pour la technique, je tiens à remercier la communauté de développeurs web qui fournit, depuis de nombreuses années, une aide précieuse, complète, et détaillée sur de multiples forums. Leurs topics m'ont été très utiles lorsque j'ai été confronté à des problèmes techniques.

Enfin, je tiens à remercier Ricardo Uribe Lobello ainsi que Aldo Gonzalez Lorenzo pour leur aide sur certains points précis.

Table des matières

RESUME	1
RESUME	1
RESUMEN	2
REMERCIEMENTS	3
TABLE DES MATIERES	4
INTRODUCTION	6
PRESENTATION DE L'ORGANISME	7
Implantation géographique et modèle économique	7
Historique	7
Situation économique	8
Secteurs d'activités	8
Priorités d'investissement	8
Organisation interne	8
SWOT	9
Avant COVID	9
Après COVID	9
Moyens informatiques	10
SUJET DU STAGE	10
PRESENTATION DU TRAVAIL ACCOMPLI	12
Communication sur les RS	12
Problématique	12
Démarche & Réalisation	12
Facebook	12
Instagram	14
Développement web en interne	15
Problématique	15
Le cahier des charges	16

Démarche	18
Architecture du logiciel	19
Réalisation	20
Fonctionnement du modèle	20
L'erreur de ma POO	20
Structure	21
Les redirections	22
Les constantes	22
La connexion	23
Les autorisations	23
Vérification des entrées	24
Les frameworks	25
La gestion des associations multiples	26
Fonctionnalités	28
Ajout d'un document	28
Modification du style	30
Modification du contenu	31
Les notifications	33
Le formulaire de recherche par libellé du document	33
Analyse économique	34
Analyse humaine	34
 CONCLUSION	 35
 Difficultés rencontrées	 35
 Ce que j'ai acquis pendant ce stage	 35
 ANNEXES TECHNIQUES	 37
 DIAGRAMME DE GANTT	 37
 TABLE DES FIGURES	 37
 BIBLIOGRAPHIE	 38

Introduction

Mon stage s'est déroulé en télétravail, du 13 Avril au 19 Juin, pour une durée de dix semaines. Mon organisme d'accueil a été la commune nouvelle des Belleville, et plus spécifiquement le service communication de cette dernière. Ma responsable était Valérie Hudry, responsable de ce service. C'est elle qui a supervisé mes travaux.

Mon stage s'est déroulé en deux temps, le premier qui a occupé mes deux premières semaines consistait en la réalisation de rapports justificatifs, ou non, de l'intérêt de communiquer sur Instagram. Il a aussi fallu que j'expose une stratégie de communication sur Facebook afin de rendre compte des bonnes pratiques sur ce réseau social, ainsi que de justifier de l'intérêt qu'il faut lui porter. J'ai effectué une étude de la « concurrence¹ », afin de mettre en évidence les stratégies que les communes adoptent sur ce RS (réseau social). Enfin, j'ai aidé à la mise en page d'un document portant sur la taxe d'habitation, taxe qui est gérée en grande partie par les communes.

Dans un second temps, ma mission a été de réaliser un site internet à destination des élus locaux ainsi que des services. Ce site a pour objectif de faciliter la communication entre les différents élus (maire, adjoints, élus de différentes commissions) mais aussi entre les différents services (technique, communication, administration). Un cahier des charges officieux a été défini en amont de mon stage afin de gagner du temps sur le développement. Il a été ajusté au cours du stage via différentes réunions. La demande étant extrêmement spécifique, un développement complet et le plus souple possible est de mise. Le choix des technologies et procédés utilisés seront détaillés au cours de ce rapport ainsi que les raisons de ces choix.

¹ Peut-on vraiment utiliser ce mot lorsqu'on parle de commune ?

Présentation de l'organisme

Implantation géographique et modèle économique

La commune des Belleville se situe en Savoie et est incluse dans l'intercommunalité Cœur de Tarentaise, le maire fraîchement élu est Claude Jay. Cette commune est dite « commune nouvelle », elle forme ainsi la commune des Belleville le 1^{er} Janvier 2019. Ces anciennes communes deviennent des communes déléguées avec leurs propres maires délégués.

Comportant les plus grandes stations françaises, voire mondiales, comme Saint-Martin de Belleville, Les Menuires ou encore Val-Thorens, la population varie énormément d'une saison à l'autre. En hiver la commune a une capacité estimée à 55 000 lits. Ces trois stations font partie intégrante des Trois Vallées, réunissant les Belleville, Méribel et Courchevel. Elle compte néanmoins 3 611 habitants à l'année au dernier recensement de 2019.

Les Belleville, c'est un territoire très vaste (13^{ème} commune la plus vaste de France métropolitaine² (Wikipédia, s.d.)) avec 226km² et une densité de 15.9 habitants au km², et une altitude variant de 509m à 3 564m, c'est une commune majoritairement rurale, très touristique et dont certains villages peuvent être très isolés, voire inaccessible en hiver.

Grâce au tourisme de masse en hiver et dans une moindre mesure en été, la commune des Belleville est plutôt privilégiée sur le point de vue financier.

Historique

Cette commune est dite « commune nouvelle », créée par un arrêté préfectoral le 5 Novembre 2015, elle fusionne les communes de Villarlurin et de Saint-Martin-de-Belleville dans un premier temps. Dans un second temps, la commune de Saint-Jean-de-Belleville est intégrée, formant ainsi la commune des Belleville le 1^{er} Janvier 2019. Ces anciennes communes deviennent des communes déléguées avec leurs propres maires délégués.

Dans un passé plus lointain, durant le XX^{ème} siècle, la commune a connu un essor fulgurant de son activité économique avec la construction des premières stations de ski françaises dans les années 1960. Elle est passée d'une activité majoritairement paysanne à un tourisme de sport d'hiver tourné vers la masse.

² Source : https://fr.wikipedia.org/wiki/Liste_de_communes_fran%C3%A7aises_par_superficie à noter que, vraisemblablement, les superficies n'ont pas été mises à jour avec la création des communes nouvelles.

Situation économique

La situation économique du service est accessible, via un tableur, dans les annexes du rapport ([annexe 1](#)). Ce tableur détaille le budget annuel du service communication qui est de 163 000€. Les types de dépenses sont multiples :

- Maintenance du site internet : 4 000€
- Bulletin municipal : 47 919€
- Réception / cérémonies protocolaires : 7 000€
- Communication interne : 6 700€
- Réalisation de goodies : 4 500€
- Signalétique : 20 000€
- Licences : 1 600€
- Impression : 4 000€
- Traitement salarial : 68 000€

Secteurs d'activités

Une présentation power point est disponible détaillant les missions qu'effectue le service en annexe ([annexe 2](#)).

Priorités d'investissement

La priorité en ce début de mandat était de disposer d'une plateforme permettant une meilleure communication des élus. Il était aussi urgent de produire des documents détaillant les intérêts du réseau social Instagram. Mon organisme d'accueil était une mairie, il est compliqué de détailler des priorités d'investissement.

Organisation interne

Un organigramme est disponible en annexe ([annexe 3](#)), il regroupe toute l'organisation de la municipalité.

SWOT

Avant COVID

Forces	Faiblesses
<ul style="list-style-type: none"> - Appareils photographiques. - Abonnement à une banque d'image. - Abondance de contenu. - Environnement naturel favorable à la communication sur les RS (photos, vidéos). - Moyens financiers importants en comparaisons d'autres communes. - Télé-alerte par SMS permettant un échange d'information rapide. 	<ul style="list-style-type: none"> - Manque de temps pour s'occuper des RS. - Communication inter-élus fragile.
Opportunités	Menaces
<ul style="list-style-type: none"> - Compte Instagram suivi par 200 personnes alors qu'il n'y a pas de contenu. - Compte Facebook dynamique. - Compte Instagram prometteur. - Nouvelle plateforme pour dynamiser le conseil municipal. 	<ul style="list-style-type: none"> - Peut-être les autres pages municipales.

Après COVID

Forces	Faiblesses
<ul style="list-style-type: none"> - Page Facebook plus active et régulière que ses concurrents. - La page a généré de l'entraide. 	<ul style="list-style-type: none"> - Budgets revus à la baisse (potentialité). - Manque de temps pour s'occuper des RS.
Opportunités	Menaces
<ul style="list-style-type: none"> - Les réseaux sociaux ont été beaucoup utilisés pendant le confinement, il faut continuer sur cette voie. - Utiliser les comptes pour promouvoir des vacances françaises. - Promouvoir l'entraide ainsi qu'une vie écoresponsable. - Utiliser la nouvelle plateforme web pour dynamiser la communication inter-élus. 	<ul style="list-style-type: none"> - Peut-être les autres pages municipales.

Moyens informatiques

La commune travaille avec un prestataire de service qui gère l'hébergement ainsi que la maintenabilité et la sécurité de leurs différentes plateformes web. Les technologies utilisées sont très proches de celles vues pendant ma formation, à savoir du WordPress pour les sites internet, du PHP pour l'ancien site que j'ai refondu, et des licences Adobes pour les besoins graphiques (InDesign et Photoshop en majorité).

Relativement aux moyens informatiques de la commune, elle dispose d'environ 50 postes informatiques, d'autant de licences. Les logiciels sont multiples, environ 25 différents : l'état civil, la comptabilité, le suivi des cantines (inscription et paiement), la taxe de séjour, les RH, la communication, le suivi des travaux, la commande publique, le cadastre, la gestion du courrier, la communication avec la suite Adobe. Le service dispose aussi de 2 appareils photographiques dont un récent qui fait de la vidéo ainsi que du matériel audiovisuel tel qu'un pied une perche et un micro.

Les prestataires étant très nombreux, je vais ici détailler ceux qui travaillent pour le service communication. [Newquest](https://www.newquest-group.com/)³ pour le site internet, [Groupe Calliope](https://www.groupe-calliope.com/)⁴ pour l'informatique (matériel, hébergement, sécurisation site...), agence de communication [Atelier Confiture Maison](https://atelier-confituremaison.com/)⁵, différents imprimeurs dont [Couleurs Montagne](http://www.couleurs-montagne.com/)⁶ pour les bulletins municipaux, [Groupe SI2A](http://www.groupe-si2a.com/)⁷ pour les licences Adobe et formations. Ils travaillent également avec différents traiteurs ([Seb Gourmet Traiteur](https://sebgourmet-traiteur.com/)⁸, [Le Petit Marmiton](https://www.facebook.com/Le-Petit-Marmiton-1380197468954218/)⁹) ou restaurants pour les cérémonies protocolaires.

Disposant de la licence Adobe, d'un environnement de développement complet ainsi que d'un serveur web sur ma machine, je n'ai pas eu besoin de travailler avec les moyens de la commune. Au moment où j'écris ces lignes, le site est hébergé sur mon serveur personnel car le site internet est en phase de test. Il sera prochainement mis en ligne sur les serveurs de la commune, via leur prestataire de service.

Sujet du stage

Comme détaillé dans l'introduction, mon stage s'est déroulé en deux temps. J'étais tout d'abord attendu pour une étude critique relative à leur manière de communiquer sur les RS, et plus particulièrement sur Instagram. En effet, la commune, cherchant davantage à communiquer avec les jeunes, souhaite s'implanter sur ce réseau social. C'est en cela que mon aide leur a été utile puisque, même si je ne suis pas un utilisateur d'Instagram, j'ai travaillé sur des projets similaires durant mon DUT.

³ <https://www.newquest-group.com/>

⁴ <https://www.groupe-calliope.com/>

⁵ <https://atelier-confituremaison.com/>

⁶ <http://www.couleurs-montagne.com/>

⁷ <http://www.groupe-si2a.com/>

⁸ <https://sebgourmet-traiteur.com/>

⁹ <https://www.facebook.com/Le-Petit-Marmiton-1380197468954218/>

De la même manière, la mairie souhaitait que je réalise une étude sur leur page Facebook. Un rapport qualitatif présentant l'importance et les bases de la communication sur ce réseau social d'une part, un second portant sur l'étude quantitative de l'activité de différentes pages municipales d'autre part.

L'objectif étant de donner des informations claires et concises aux services afin d'orienter leurs choix et d'optimiser leur campagne de communication. J'ai aussi travaillé sur la forme du document, créer un déroulé précis et expliciter les raisons de mes choix, afin de construire une démarche logique et cohérente. J'ai également voulu fournir des données chiffrées aux services en leur proposant une étude statistique ainsi qu'une interprétation de ces chiffres. En effet, les données étant très nombreuses, proposer des chiffres clefs était pour moi une priorité.

Ensuite, j'ai effectué le cœur de mon stage, ce pourquoi j'ai véritablement été pris, à savoir le site internet. En effet, dispose de plus de villages en son sein. Il y a donc davantage d'espace entre eux, de plus, ces informations sont multiples, cela peut être des comptes-rendus de conseils municipaux, des arrêtés ou encore d'autres comptes-rendus de diverses commissions. Dans ce cadre, il est indispensable que l'information circule de manière ciblée et centralisée. Les services souhaitaient une solution souple et personnalisable au possible, du fait de la nouvelle équipe municipale qui pourrait modifier le cahier des charges, ils ont opté pour un développement personnalisé. Les CMS (Content Management System ou Système de Gestion de Contenu) étant trop peu souples pour ce genre de logiciel il a fallu qu'ils s'orientent vers du développement. Un développeur étant très cher pour ce type de logiciel sur-mesure, ils ont choisi de me prendre en stage pour que je réalise cette plateforme web ambitieuse.

En substance, cela se présente sous la forme d'un drive où les documents sont mis en ligne par certains utilisateurs (souvent par une responsable du service communication) à destination d'autres utilisateurs, à savoir les élus et/ou les employés municipaux. Ces utilisateurs sont regroupés sous forme de listes et chaque liste dispose d'accès à certaines catégories de documents. Pour chaque document mis en ligne, des mails sont envoyés de façon ciblée aux personnes concernées pour les avertir de l'ajout d'un document les concernant. Le site dispose également d'une interface administrateur qui détermine les droits de chaque utilisateur, à savoir leur capacité à mettre un document en ligne, à le modifier, à le supprimer, à notifier d'autres personnes etc. Les permissions sont multiples et elles seront détaillées plus loin dans ce rapport. L'interface administrateur dispose, à la façon de WordPress, de la possibilité de modifier le contenu du site comme les couleurs, les tailles de typographie mais aussi le corps du mail ou encore le contenu affiché sur le site. Cette partie sera aussi détaillée plus avant.

Présentation du travail accompli

Communication sur les RS

Problématique

Afin d'assurer une communication cohérente et efficace dans un environnement morcelé et dont les habitants sont parfois éloignés de plusieurs kilomètres avec la mairie, il devient cohérent d'utiliser les nouveaux moyens de transmissions de l'information. En effet ces derniers possèdent deux avantages que n'ont pas les moyens de communication traditionnels.

Ils sont centralisés, ce qui permet une circulation de l'information bien plus rapide par rapport à l'affichage qui nécessite le déplacement des employés communaux dans toute la commune, d'une part. L'impact de cette communication est mesurable via les statistiques des différents réseaux sociaux, voire celles du trafic du site web, d'autre part.

L'enjeu est donc de proposer un nouveau canal d'information pour les administrés, comme cela est déjà le cas sur Facebook, mais aussi de s'adresser aux jeunes. En effet, la commune met un point d'orgue à essayer de garder ses jeunes, souvent tentés d'aller dans les centres urbains. La problématique est donc de concilier les impératifs d'une communication officielle qui peut être rigide, tout en la rendant attirante pour le plus grand nombre, particulièrement pour les jeunes.

C'est dans ce contexte que le service communication a décidé de créer un compte Instagram, réputé plus proche des jeunes, ainsi que de développer leur communication sur Facebook.

Démarche & Réalisation

Facebook

Il m'a été confié dans un premier temps de réaliser des documents justifiant de l'importance de Facebook pour une mairie. J'y ai détaillé les nouveaux usages des internautes, qui sont majoritairement sur mobile, mais aussi justifié l'importance d'être présent sur les RS.

Dans un second temps j'ai traité des points clefs d'une campagne réussie sur ces médias. J'y ai détaillé l'importance de l'engagement d'une communauté ainsi que le fonctionnement de l'algorithme. Ensuite, j'ai présenté les nouveaux médias d'info-divertissement qui peuvent être un exemple à suivre, ainsi que d'autres exemples pour susciter l'engagement de sa communauté. J'y ai justifié de l'importance du réseau social sur lequel on communique, il faut adapter sa communication en fonction de la plateforme et in fine en fonction des personnes qui la fréquentent. Enfin, une étude critique sur les publications Facebook de la mairie a été effectuée, ce, en tenant compte de l'aspect institutionnel du contenu, autant sur la forme que sur le fond.

J'ai rédigé un aparté sur Messenger dans le but d'ouvrir le champ des possibles, peut-être même de toucher un autre public que celui qui irait voir les publications. En effet, Messenger peut-être un allié

très puissant pour répondre aux questions des habitants et peut donner un sentiment de proximité avec les dirigeants.

Pour terminer avec Facebook, une analyse approfondie du contenu, mais aussi des pages d'autres communes a été diligentée. Cette analyse est suivie d'une étude statistique afin de donner des clefs de compréhension au service. Cette étude a été effectuée sur trois pages de différentes communes voisines des Belleville et sensiblement de même taille. Elle a été réalisée à la fin du confinement et elle couvre toute cette période (17 Mars-11 Mai 2020 inclus). Sur les 55 jours qu'a duré le confinement en France, toutes les données ont été relevées, et ce, pour les trois pages (nombre de publications par jour, titre de la publications, nombre de j'aime, commentaire et partage).

Après ce travail de collecte, j'ai traité ces données sur un tableau Excel pour en extraire des tendances. J'ai tout d'abord effectué les sommes de toutes les interactions ainsi que de toutes les publications, afin d'avoir des valeurs absolues, qui peuvent représenter le poids de chaque page. Ensuite j'ai calculé les nombres moyens de ces différentes interactions sur toute la période de confinement. Divers diagrammes ont été créés grâce à ces données, ils rendent compte du nombre de publication par jour, de la part des différentes interactions pour chaque page ou encore de l'évolution de ce nombre d'interaction sur la durée de l'étude. J'ai continué cette étude statistique en intégrant l'écart-type de ces différentes valeurs. Cet écart-type représente la dispersion moyenne des valeurs d'une série, il décrit, de façon absolue, l'écart de performance moyen de chaque publication. Grâce à cette donnée on peut rendre compte de la longévité des performances de la page. En effet, un écart-type important signifie que certaines publications, en moyenne, suscite beaucoup plus d'intérêt que d'autre. Cet écart de performance peut être intéressant à mesurer au cas par cas ; il peut rendre compte de l'intérêt des habitants pour des thématiques particulières.

Cependant, cette donnée à un défaut important lorsqu'il s'agit de comparer plusieurs séries entre elles : l'écart-type donne une valeur absolue. Pour bien comprendre le problème que cela pose lorsque l'on veut comparer deux séries entre elles, comme c'est le cas ici, illustrons ce problème avec deux pages Facebook. L'une a en moyenne 5 000 likes et l'autre 500 likes par publications, dans ce cas là on comprend aisément qu'un écart moyen de 10 likes par publications n'a pas la même importance pour la page à 5 000 likes en moyenne, que pour celle qui n'en compte que 500. Cela peut être négligeable lorsque les pages sont similaires, ce qui n'est pas tout à fait le cas pour les trois pages que j'ai étudié.

Il a donc fallu que j'utilise une autre valeur, le coefficient de variation (appelé aussi l'écart-type relatif), qui exprime la dispersion moyenne d'une série mais de façon relative cette fois. Elle est donc exprimée en pourcentage. Pour bien comprendre de quoi il s'agit, prenons une valeur présente dans mon étude (annexe) : pour la page de la commune des Belleville, le CV est de 198% pour les likes. Cela signifie qu'en moyenne il y a une différence, d'un facteur trois, du nombre de likes entre deux publications.

On peut en tirer la conclusion suivante, il existe des publications qui fonctionnent très bien et d'autres qui ne sont que très peu likées. On pourrait faire de même avec le nombre d'impression, mais lors de mon étude statistique, je n'avais pas accès aux données de la page des Belleville, et je n'aurai pas non plus eu accès à ce type de données pour les autres pages.

Pour terminer sur ce point, j'ai compilé dans un tableur Excel toutes les données récoltées, et, conscient du fait que cette masse de chiffre ne signifie pas grand-chose, j'en ai fait des graphiques et autres diagrammes afin d'en faire une représentation plus visuelle et donc plus parlante pour la majorité. Cette étude pourra être réitérée dans six mois ou un an afin de quantifier l'évolution de la page Facebook.

Instagram

Dans un second temps j'ai effectué le même travail sur Instagram. A la différence que, même si la commune possède déjà un compte Instagram, il n'a jamais été alimenté. Ceci, afin d'éviter qu'une autre personne ne se fasse passer pour le compte officiel, comme cela a été le cas pour d'autres villes. Tout d'abord j'ai débuté par énoncer les différences entre Facebook, déjà connu du service, et Instagram. Cela, surtout pour le public présent sur ce réseau social, le contenu présent mais aussi les différences de mœurs. En effet, pour parler à un public plus jeune il faut adopter ses codes, sans pour autant tomber dans le cliché. Pour éviter ce vice qui pourrait s'avérer contre-productif, il peut être pertinent de regarder du côté des influenceurs, ces stars de la plateforme qui sont totalement en phase avec le contenu type de ce réseau social. J'ai aussi exposé quelques exemples qui témoignent de la force de frappe de ces personnes. En me basant sur des articles qui traitent des « do and don't » d'Instagram, puisque je rappelle que je ne suis pas un utilisateur de ce réseau social, j'ai ainsi pu documenter mes conseils et mon analyse.

J'ai détaillé l'importance de l'image sur ce réseau social, qu'elle soit fixe ou animée, l'importance d'être proche de sa communauté, mais aussi la puissance des stories Instagram qui sont aujourd'hui parmi les plus suivies au monde, loin devant celles de Facebook et en concurrence directe avec celles de Snapchat. Comme pour son homologue Facebook, j'ai détaillé l'importance de l'algorithme ainsi que la part non négligeable des likes. Après avoir listé ce qui fait de ce réseau social un incontournable, surtout lorsque l'on souhaite s'adresser aux jeunes, j'ai effectué une étude de comptes Instagram similaires à celui que souhaiterai lancer la mairie. J'ai repris les mêmes municipalités que lors de l'étude Facebook, mais devant le très faible nombre de communes présentes sur cette plateforme, j'ai revu à la hausse le nombre de page à étudier.

En effet, après avoir épluché les communes limitrophes des Belleville je me suis aperçu que très peu d'entre-elles possèdent un compte Instagram. Ce qui est étonnant quand on sait que les stations correspondantes à ces communes ont presque toutes un compte Instagram (Tignes, Val-Thorens, Les Menuires, La Plagne, Meribel, Courchevel ...). Je n'ai trouvé que deux comptes à exploiter, la mairie de Moûtiers qui possède deux comptes dont un inactif et la commune de Tignes qui est inscrite mais qui est inactive. En voyant ces maigres résultats j'ai fait le choix de comparer avec Facebook, j'ai donc pris un panel de commune limitrophes avec celles des Belleville et j'ai présenté un tableau comportant la présence sur Facebook et Instagram. Quand celles-ci étaient présentes, j'ai détaillé leur nombre de publication ainsi que leur nombre d'abonné, leur activité sur les deux réseaux sociaux ainsi que la présence d'une certification.

Le constat est disponible dans les annexes (préciser quelle annexe), seule une des deux pages de Moûtiers est active régulièrement. Il y a donc une place à prendre sur ce réseau social pour donner une image branchée de la commune des Belleville. C'est une réelle opportunité pour la commune, d'autant plus que le service communication dispose du matériel photographique qui permettrait une abondance de contenu audiovisuel. J'ai également incité la commune à essayer d'obtenir la certification sur les deux réseaux sociaux, ce qui serait un gage de professionnalisme et un atout supplémentaire pour eux.

Ensuite, en m'appuyant sur le compte Instagram de Moûtiers, j'ai développé une critique et, en parallèle, j'ai expliqué le fonctionnement de ce réseau social. J'ai débuté en précisant que, comme pour Facebook, les blasons des pages sont maintenant affichés dans des cercles, il faut donc en tenir compte pour que le logo de la commune ne soit pas coupé. J'ai ensuite expliqué l'importance d'une

fréquence de publication la plus régulière possible, puis l'importance des médias natifs en reprenant les thèmes les plus en vogue de la plateforme (voyage, nourriture ...). Enfin, j'ai poursuivi en expliquant le principe puis l'importance des hashtags pour conclure en exposant les intérêts d'être proche de sa communauté.

Développement web en interne

Problématique

Comme développé dans l'introduction, avec la création de la commune nouvelle, l'espacement géographique entre les élus a augmenté. De plus, la mairie était en recherche d'un moyen de centraliser et de fluidifier la transmission d'informations pour répondre à des problèmes internes. Ces problèmes pouvaient être la mauvaise transmission des documents aux élus, certains ne recevaient pas les documents qu'ils devaient pourtant avoir, alors que d'autres recevaient des fichiers qui ne leurs étaient pas destinés. De plus, cela représentait un travail administratif systémique et lourd, une automatisation de ce processus était devenue une nécessité. Enfin, un archivage des fichiers sur serveur, ainsi qu'une gestion massive de ces derniers permet un gain de temps, et une sécurisation de ces derniers en cas de perte de données.

Il est important de préciser que la mairie disposait déjà d'une plateforme web-interne développée par un de leur ancien prestataire de service similaire à celle-ci. Cependant, elle n'était plus adaptée aux besoins de la commune. En effet elle était vieille, pas responsive, et ne disposait pas de suffisamment d'options pour l'hébergement d'une multitude de document et ne supportait pas différentes catégories d'utilisateurs, entre autres problèmes.

De plus, il faut ajouter à cela un agenda centralisé, visible par les élus et dans lequel sont notés tous les événements de la mairie. Cet agenda n'était pas clair pour certains élus et il convenait de le rendre accessible d'une manière simple et ergonomique sur ce site internet. Pour terminer, les élus possédant chacun une tablette pour consulter leurs mails ainsi que leur agenda, il était nécessaire que le site internet soit responsive. Je vais détailler ici le cahier des charges de ce site internet. Il est important de noter que ce cahier des charges a été ajusté au fur et à mesure du stage, il a été en perpétuel mouvement, ce qui a permis d'adapter le site au fur et à mesure des réunions. Je parlerai de ce point dans les difficultés rencontrées.

Le cahier des charges

Le cahier des charges, a été défini comme suit, il a ensuite été retouché sur le tas. Il n'est donc pas exhaustif de tout ce qui a été fait pour le site internet. Il est disponible en annexe ([annexe 4](#)).

- Présence d'un portail de connexion
- Inscription sur la plateforme par une personne qui a les droits de créer un compte
- Catégories d'utilisateurs (super-administrateur, élus, adjoints ...)
 - Catégories gérées par un ou plusieurs comptes qui en ont les droits
 - Chaque catégorie a un accès à différentes sous-catégories de document
- Autorisations pour chaque utilisateur
 - Autorisations par défaut à l'inscription (abandonné en cours pour des raisons de sécurité)
 - Autorisations par défaut au changement de statut d'un utilisateur (abandonné en cours pour des raisons de sécurité)
 - Autorisations personnalisables pour chaque utilisateur (créer, modifier, supprimer un compte, un document, une catégorie [document et utilisateur], notifier des utilisateurs)
- Compte individuel et différencié pour chaque utilisateur
 - Choix d'être notifié par mail
 - Envoie d'un mail au responsable du site (arrive dans une boîte mail modifiable par le super-administrateur)
 - Personnalisation de l'affichage du site (mode sombre [à l'étude])
- Interface d'administration
 - Envoie de mail ciblé ou par catégorie d'utilisateur
 - Gestion des catégories des documents
 - Ajout
 - Modification
 - Modification en cascade sur les documents
 - Suppression
 - Gestion en cascade sur les documents
 - Liste de tous les utilisateurs du site
 - modification des autorisations
 - visualisation des informations de l'utilisateur
 - modification des informations d'un utilisateur
 - informations personnelles (mail, nom-prénom, statut, mail)

- réinitialisation du mot de passe
- suppression d'un utilisateur
- Administration du design du site (couleurs, ajout de CSS), de son contenu (titre, textes ...) Ainsi que de son paramétrage (contenu du mail de notification, objet)
- Documents
 - Ajout
 - notification des utilisateurs par mail
 - notifications ciblées et automatique
 - ajout d'utilisateurs/listes d'utilisateurs à notifier
 - Modification
 - Suppression
 - Date de validité
 - Consultation et/ou téléchargement des documents
 - Système de vue sur certains documents pour avoir un fil d'actualité propre et dynamique
- Sécurisation des mots de passe (hashage)
- Mode mobile
- Fichier configuration qui permettra de configurer les paramètres essentiels du site (adresse, mot de passe et login de la base de données).

Démarche

J'ai tout d'abord analysé l'ancienne plateforme, puisque Valérie Hudry m'a conseillé, dans un souci de gain de temps, de partir de ce qui existait déjà. Les technologies étaient compatibles puisque j'avais prévu de développer un site internet en PHP pour le traitement des données, ce même langage était utilisé pour l'ancienne plateforme. Mais ce site internet rencontre plusieurs problèmes :

- La version de php utilisée est très vieille (au moins 8 ans) il se peut donc que des failles soient présentes.
- Le site n'est pas adapté à une utilisation sur mobile, il faut donc revoir le HTML dans sa globalité.
- Il n'a pas été conçu pour accueillir différentes listes d'utilisateurs, il était à la base fait pour archiver des documents, et pouvoir les retrouver facilement.

Il faut rajouter à cela le temps de compréhension du code de mon prédécesseur, ce qui, selon moi, aurait été une perte de temps inutile aux vues des inconvénients listés plus haut. J'ai donc décidé de repartir de zéro, de développer entièrement le back-end ainsi que de me baser sur le framework Bootstrap, afin de faciliter le mode responsif. Cette solution a été approuvée par Valérie Hudry.

J'ai ensuite commencé le développement du site internet, sans passer par une modélisation UML, j'ai fait ce choix pour deux raisons :

- Je savais que le cahier des charges serait amené à évoluer, ma modélisation pourrait donc être rapidement obsolète.
- J'ai considéré cela comme une perte de temps, aux vues du type de site que cela représentait. En effet, j'ai déjà créé de tels sites et je ne pensais pas avoir besoin d'une modélisation UML pour en avoir une vue d'ensemble durant le développement.
- Le site internet a effectivement beaucoup évolué durant le stage, et même si le principe est resté le même, les associations entre les différents modules ont énormément évolués.

Avec du recul, je pense que j'aurai dû prendre du temps pour créer une modélisation UML. En effet, même si je ne pense pas que cela m'aurait fait gagner du temps, elle m'aurait simplifiée la tâche lorsque des modifications devaient être faites à l'issue des réunions. Je ne pense pas qu'un développeur plus expérimenté aurait pris le temps de le faire, mais pour un débutant comme je le suis il aurait été préférable de poser de bonnes bases.

Dès le départ j'avais dans l'idée qu'il faudrait que le site soit le plus souple possible, et ce, afin de justifier de mon refus d'utiliser un CMS. C'est pourquoi j'ai intégré dès le début un motif d'architecture logiciel que nous avons vu en cours lors du dernier semestre, le modèle MVC (Modèle-Vue-Contrôleur). J'ai en effet estimé que le nombre de fonction étant tel, qu'il faudrait au moins un fichier, voir un répertoire qui centralise tous les appels à la BDD (Base De Donnée) ainsi que le traitement de ces données. Il me fallait aussi un autre fichier qui me serve à afficher les données traitées par le Contrôleur, et entre les deux, un répertoire qui gère les permissions, les redirections mais aussi la gestion des erreurs. Ce répertoire peut s'apparenter au Modèle.

Je me suis directement inspiré du framework PHP CodeIgniter que nous avons vu en cours lors du 4^{ème} Semestre. En effet, j'ai trouvé qu'il était particulièrement adapté au développement de logiciel de cette envergure, mais je ne disposais pas des connaissances suffisantes pour le développer via cet outil.

Cette architecture induit naturellement l'utilisation de la POO (Programmation Orientée Objet), ce qui a été étudié lors du Semestre 3 et 4.

Architecture du logiciel

Je suis tout à fait conscient que l'architecture de mon logiciel ne reprend que partiellement, et grossièrement, le concept MVC. L'ayant trop peu étudié pour que j'en comprenne les subtilités, je me suis permis de le remanier pour essayer de concevoir une architecture suffisamment robuste et fiable dans le but de gagner du temps lors du développement, mais aussi afin d'avoir des répertoires aux fonctions bien définies. Le but était surtout de ne pas mélanger tous les fichiers php, ce qui peut se traduire par une perte de temps considérable au fur et à mesure que le logiciel se complexifie. Je suis aussi conscient du fait que mon fichier qui se veut être Contrôleur est en réalité un mix entre les fichiers Modèles et Contrôleur de l'architecture initiale. De même que mes fichiers scripts ne sont pas à proprement parlé des fichiers Modèles. Il en va de même pour la Vue qui effectue aussi quelques calculs parfois, ce qui est normalement à proscrire.

Développer un logiciel de cette envergure est pour moi une première, j'ai donc essayé d'anticiper au maximum, en amont, les difficultés d'organisation que je pourrai rencontrer. J'y suis parvenu pour la majorité, m'évitant bien des contre-temps, mais j'ai aussi énormément appris sur l'organisation rigoureuse qui doit être la pierre angulaire dans ce type de réalisation.

Réalisation

Fonctionnement du modèle

On peut schématiser le fonctionnement de mon modèle comme suis :

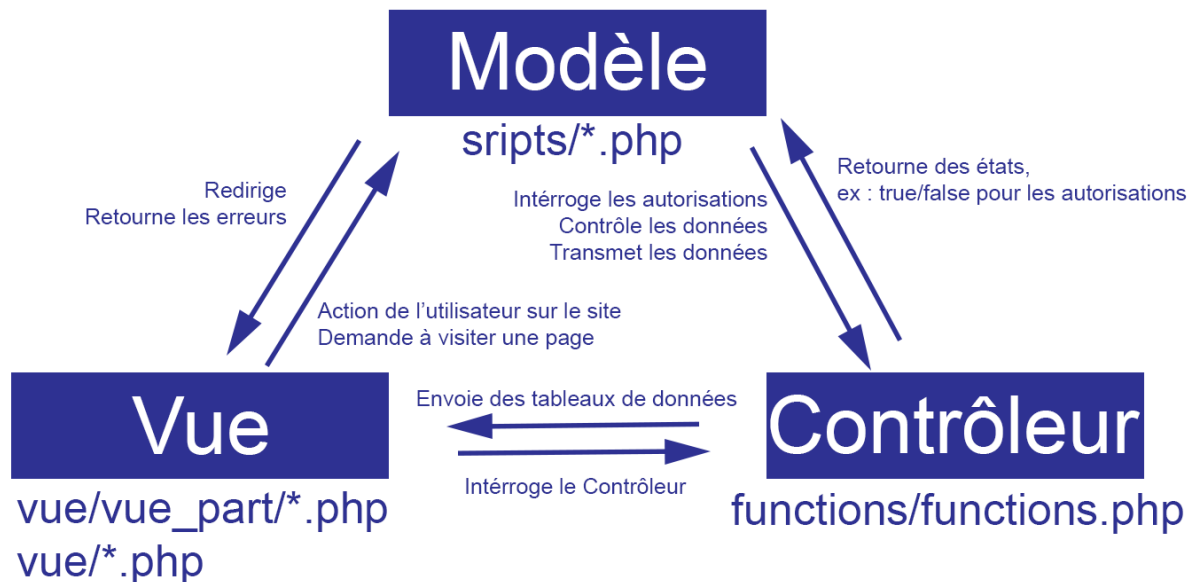


Figure 1 : schématisation du fonctionnement de mon code

On constate donc que mon modèle diffère de celui d'un MCV classique. En effet, ici la vue n'agit pas directement sur le Contrôleur, elle ne fait que demander des données par le biais des méthodes de la classe fonctions. De plus, pour l'ajout, la suppression ou la modification d'entités, on passe d'abord par les fichiers scripts, le Modèle. Il interroge ensuite les droits pour que le Contrôleur retourne les données à la vue.

L'erreur de ma POO

J'ai énormément appris en POO pendant la réalisation de ce logiciel. J'avais un niveau plutôt faible à l'issue du troisième Semestre, il a été renforcé avec l'utilisation du framework CodeIgniter. Cependant, je jugeais mon niveau encore en deçà du seuil nécessaire pour mon orientation, à savoir une licence d'informatique. Il est évident que j'ai énormément progressé durant ce stage, mais j'ai fait aussi des erreurs sur la POO.

En effet, comme décrit plus haut, j'ai créé un seul fichier « functions.php » faisant office de Contrôleur. Ce fichier regroupe une seule class appelée « functions » qui elle-même regroupe toutes les méthodes nécessaires aux fonctionnalités du site. Avec le recul, il est évident que j'aurai dû créer plusieurs autres classes, une qui aurait permis l'ajout d'une entité dans la BDD, une autre qui l'aurait supprimée etc. Cela, agencé avec des héritages et des constructeurs intelligemment disposés. J'estime avoir une POO

relativement propre pour ce site internet, ma méthode de programmation pourrait être meilleure, cependant, je suis satisfait des progrès que j'ai effectué durant ce stage.

Structure

Une fois que j'ai terminé de poser les bases théoriques sur papier, j'ai commencé le code. J'ai tout d'abord créé les répertoires contenant la structure MVC. L'architecture se structure comme suis :

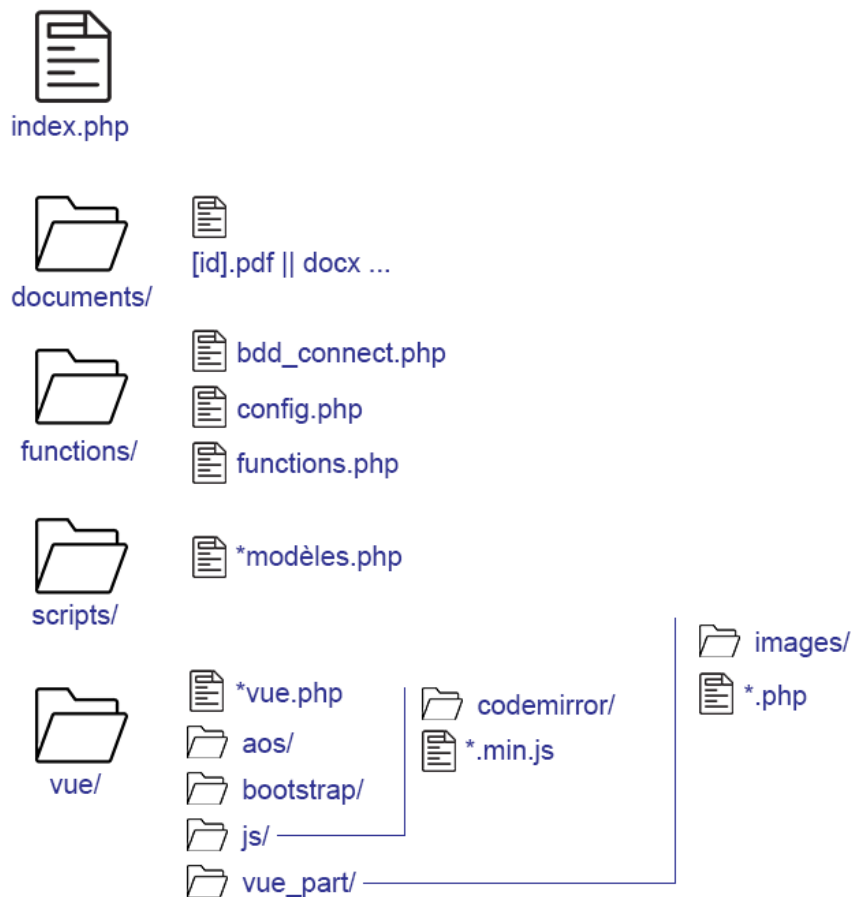


Figure 2 : structure des répertoires & fichiers du site en vue simplifiée

La vue est simplifiée, elle ne liste pas tous les fichiers php d'une part, et ne liste pas les fichiers ni sous-répertoires issus des frameworks que j'ai intégrés d'autre part. Ces frameworks seront détaillés dans la suite du rapport. Je vais donc expliciter, en me basant sur ce schéma, le contenu ainsi que la fonctionnalité de chaque élément. Je vais aussi en profiter pour expliquer comment ont été conçues certaines fonctionnalités.

Le fichier `index` qui est la page sur laquelle l'internaute arrive lorsqu'il veut atteindre le site. Cette page ne sert qu'à la redirection, en fonction de l'état de l'internaute. En effet, si celui-ci est connecté, il est redirigé vers le dossier `vue/` puis dans le fichier `index` de ce répertoire. De là, il a accès, en fonction de ses autorisations, au contenu du site internet. S'il n'est pas connecté au moment d'arriver au premier `index`, il est redirigé dans la vue, mais cette fois sur la page de connexion.

Les redirections

Au départ, j'utilisais la fonction [header](#)¹⁰('location : [adresse absolue ou relative]') qui permet l'envoi d'une en-tête HTTP. Avec des conditions, je redirige ou non les internautes en fonctions de leur état de connexion ainsi que de leur poste, et donc de leurs autorisations.

Cependant, au moment de l'envoi sur le serveur, d'abord sur le mien pour la phase de test, je me suis aperçu que les redirections PHP ne fonctionnaient pas. Après des recherches, j'ai compris qu'il fallait que j'utilise la fonction [ob_start\(\)](#)¹¹ afin d'éviter d'envoyer des données, ce qui peut bloquer les envois d'en-tête HTTP. Profitant qu'il faille que je revoie le fonctionnement de mes redirections, j'ai donc fait le choix de créer une fonction `redirect()` dans mon contrôleur afin de centraliser ce processus. Grâce aux constantes définies dans ce même fichier le chemin de redirection est absolu.

A chaque fois qu'une redirection a lieu, elle est suivie de l'instruction `exit;` qui permet l'arrêt de l'exécution du code. En effet, il arrive que les redirections soient ignorées dans certaines conditions. L'arrêt du script est donc obligatoire pour garantir le bon fonctionnement du site ainsi que sa sécurité.

Les constantes

Grâce à la fonction [define\(\)](#)¹², j'ai défini plusieurs constantes dans mon fichiers Contrôleur. Au moment d'écrire ce rapport, elles sont au nombre de 4 dans mon fichiers Contrôleur, 5 dans mon fichiers config.

Contrôleur -> `functions.php`

- `ROOT_HTTP` : chemin absolu vers la vue
- `MAIL_MESSAGE` : adresse mail par laquelle on peut contacter l'administrateur du site via un formulaire
- `OBJET_NOTIFICATION` : objet du mail de notification
- `MESSAGE_NOTIFICATION` : corps du mail envoyé pour notifier une ou plusieurs personnes

Toutes ces constantes (sauf `ROOT_HTTP`) sont modifiables par le super-administrateur via une interface graphique.

Config -> `config.php`

- `HOST` : adresse de la BDD
- `DB_NAME` : nom de la BDD
- `DB_LOGIN` : login de la BDD
- `DB_PASSWORD` : mot de passe de la BDD
- `DB_CHARSET` : jeu de caractère pour encoder la BDD

J'ai repris les mêmes variables que pour WordPress. Ces valeurs ne sont pas modifiables, étant donné leur nature. En effet on ne peut pas, via un appel à la BDD, récupérer les logs pour se connecter à cette dernière.

Ces constantes ont pour but de simplifier le code, d'avoir des valeurs directement visibles comme étant constantes plutôt que d'en faire des variables.

¹⁰ <https://www.php.net/manual/fr/function.header.php>

¹¹ <https://www.php.net/manual/fr/function.ob-start>

¹² <https://www.php.net/manual/fr/function.define.php>

La connexion

Pour se connecter, les utilisateurs disposent d'un couple identifiant-mot de passe créé par le super-administrateur (ou des personnes ayant l'autorisation de créer un compte, ce point est détaillé juste après). Les mots de passe seront ensuite chiffrés grâce à la fonction [`crypt\(\)`](#)¹³ ou avec l'API [`Book.Password`](#)¹⁴. Il existe aussi la fonction [`password_hash\(\)`](#)¹⁵ qui est décrite comme plus robuste. N'ayant jamais utilisé d'API en PHP, cette solution est encore à l'étude. Je ne connais que très peu les moyens de chiffrer des mots de passe, c'est pourquoi aujourd'hui je ne dispose pas encore de solution définitive. Ce qui est certain, c'est que les mots de passe seront chiffrés.

Le login est l'adresse mail du compte, en effet plusieurs personnes disposent du même nom de famille parmi les élus, et il peut être possible de trouver des homonymes. L'adresse mail, elle, est unique. Une fois le couple login-mot de passe rentré, il est comparé avec l'entrée en BDD, et s'il est correct une variable de session est créée et atteste de sa connexion. Cette variable sert donc à vérifier, par le biais d'une fonction `is_connected()` qui prend en paramètre la variable session, l'état de l'utilisateur. Si l'utilisateur n'est pas connecté, c'est que cette variable n'existe pas ou quelle est à l'état false, alors la fonction retourne false, si elle existe et qu'elle a l'état true, alors la fonction retourne true.

Les autorisations

Tous les utilisateurs ont une liste d'autorisations, il y a 10 autorisations au total :

- Créer un compte
- Modifier un compte
- Supprimer un compte
- Ajouter un document
- Modifier un document
- Supprimer un document
- Ajouter une catégorie (documents et utilisateurs)
- Modifier une catégorie (documents et utilisateurs)
- Supprimer une catégorie (documents et utilisateurs)
- Notifier un/des utilisateur(s)

Toutes ces autorisations sont répertoriées dans la BDD, et chaque ligne correspond à un utilisateur. Le super-administrateur possède par défaut toutes les autorisations à l'état « oui ». Lui seul peut définir les autorisations de toutes les autres personnes, mais pas les siennes. A la manière de WordPress, la définition des autorisations se présentent comme suit :

¹³ <https://www.php.net/manual/fr/function.crypt.php>

¹⁴ <https://www.php.net/manual/fr/book.password.php>

¹⁵ <https://www.php.net/manual/fr/function.password-hash.php>

Les utilisateurs

Admin Suppression			
administrateur@gmail.com			
Créer cpte <input type="checkbox"/>	Modifier cpte <input type="checkbox"/>	Suppr. un cpte <input type="checkbox"/>	Ajouter doc <input type="checkbox"/>
Modifier doc <input type="checkbox"/>	Suppr. doc <input checked="" type="checkbox"/>	Ajouter cat. <input type="checkbox"/>	Modif. cat <input type="checkbox"/>

contributeur1			
contributeur1@gmail.com			
Créer cpte <input type="checkbox"/>	Modifier cpte <input type="checkbox"/>	Suppr. un cpte <input type="checkbox"/>	Ajouter doc <input checked="" type="checkbox"/>
Modifier doc <input checked="" type="checkbox"/>	Suppr. doc <input checked="" type="checkbox"/>	Ajouter cat. <input type="checkbox"/>	Modif. cat <input type="checkbox"/>

Admin Ajout			
test@gmail.com			
Créer cpte <input checked="" type="checkbox"/>	Modifier cpte <input type="checkbox"/>	Suppr. un cpte <input type="checkbox"/>	Ajouter doc <input checked="" type="checkbox"/>
Modifier doc <input type="checkbox"/>	Suppr. doc <input type="checkbox"/>	Ajouter cat. <input type="checkbox"/>	Modif. cat <input type="checkbox"/>

Figure 3 : capture d'écran du panneau de contrôle des autorisations pour les différents utilisateurs

Le super-administrateur peut donc définir à tout moment, au cas par cas, les autorisations de tel ou tel utilisateur. Afin de centraliser ce processus, la fonction `is_autorisation(id utilisateur, autorisation à tester)` retourne `true` si l'utilisateur peut effectivement effectuer cette action et `false` s'il ne dispose pas du droit. L'affichage dans la vue est directement lié à cette fonction, dans le but de ne pas afficher des boutons qui ne concernent pas tel utilisateur. De plus, dans les fichiers scripts cette fonction est appelée systématiquement, redirige si la personne n'a pas les droits et arrête l'exécution du programme tout en générant un message d'erreur. Cela, afin d'éviter d'accéder aux scripts en rentrant l'URL de ce dernier.

Les autorisations peuvent aussi se faire via la catégorie d'utilisateur à laquelle appartient l'internaute. En effet, comme détaillé dans le cahier des charges, un utilisateur ne peut pas nécessairement voir tous les documents. C'est pourquoi la fonction `is_poste(id utilisateur, catégorie utilisateur)` retourne `true` si l'utilisateur fait bien partie de cette catégorie.

Chaque document dispose des ids des sous-catégories auxquelles il appartient. Ces sous-catégories disposent elles-mêmes des ids des catégories d'utilisateurs qui ont accès à ces dernières. Grâce à plusieurs fonctions créées par mes soins, il est possible de savoir si tel utilisateur peut voir tel document.

Vérification des entrées

Afin d'éviter les attaques par injection de script, il convient de contrôler les entrées textuelles des utilisateurs. Pour cela, j'ai créé la fonction `verify_entry(texte)` qui combine deux fonctions PHP. La première est [htmlspecialchars\(\)](https://www.php.net/manual/fr/function.htmlspecialchars.php)¹⁶ qui convertit les caractères spéciaux en entités HTML, comme par exemple le caractère « < » qui devint « < ». Cela afin d'éviter que HTML n'interprète ce symbole comme l'ouverture d'une balise, mais plutôt comme un « inférieur à », et l'affiche. Elle permet aussi de bloquer les attaques par injection de JS, elle rend inoffensive du code qui pourrait être entré via les inputs.

La seconde fonction est [addslashes\(\)](https://www.php.net/manual/fr/function.addslashes.php)¹⁷ qui ajoute des antislashes dans une chaîne de caractères. Elle échappe les caractères qui peuvent servir à injecter du SQL via les inputs et ainsi permettre de récupérer des données dans la BDD. De façon plus générale, si un simple ou double guillemet est entré dans un input il peut alors faire planter une requête SQL. Cette fonction permet d'éviter cela.

¹⁶ <https://www.php.net/manual/fr/function.htmlspecialchars.php>

¹⁷ <https://www.php.net/manual/fr/function.addslashes.php>

Les frameworks

Trois frameworks ont été ajoutés au site afin de gagner du temps et d'améliorer l'ergonomie :

- [AOS](#)¹⁸ permet d'animer du contenu au défilement de la page. Il apporte du dynamisme et de la fraîcheur. Il n'est pas encore utilisé sur le site, puisqu'il est toujours en développement. Il sera intégré à la toute fin, une fois toutes les fonctionnalités opérationnelles.
- [Bootstrap](#)¹⁹ a été vu en cours cette année, il permet un gain de temps considérable pour le style d'un site internet par le biais de ses composants ainsi que de son style initial. Il permet aussi et surtout la mise en place du mode responsive grâce à son système de grille.
- [CodeMirror](#)²⁰ qui permet la mise en place d'un éditeur de code directement sur le site internet. En effet j'ai intégré la possibilité d'ajouter du CSS pour permettre une plus grande personnalisation, comme le propose WordPress avec son « CSS additionnel ». J'ai considéré qu'une zone de texte ne permettait pas un ajout de CSS simple, les erreurs ne sont pas marquées, il n'y a pas de couleurs... Je reviendrai en détail sur la modification du style dans la suite du rapport.

Voici ce que permet CodeMirror :

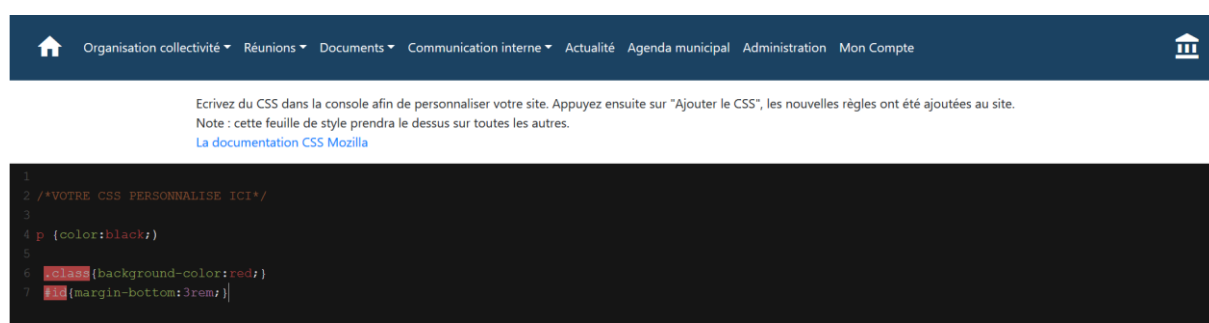


Figure 4 : capture d'écran de l'éditeur de code avec CodeMirror

Il propose une interface digne d'un éditeur de code classique, il marque même les erreurs. Il propose plusieurs langages et permet même d'en compiler certains, en plus de proposer plusieurs thèmes. L'installation est un peu fastidieuse mais il n'a besoin ensuite d'aucune configuration, en plus d'être très léger.

- [Material Icon](#)²¹ permet l'ajout d'icônes par le biais d'une l'API Google, avec un simple lien vers cette API, et en ajoutant ce code : `<i class="material-icons">[nom de l'icône disponible dans la documentation]</i>`.

¹⁸ <https://michalsnik.github.io/aos/>

¹⁹ <https://getbootstrap.com/>

²⁰ <https://codemirror.net/>

²¹ <https://material.io/resources/icons/?style=baseline>

La gestion des associations multiples

Initialement, le site internet ne devait pas pouvoir supporter certaines associations multiples. Par exemple, un document ne devait appartenir qu'à une seule catégorie et les sous-catégories n'existaient pas. Idem pour les listes d'utilisateur, un utilisateur était initialement prévu dans une seule liste. Je détaillerai lorsque j'exposerai les difficultés rencontrées dans la conclusion, il est seulement question ici de détailler comment je me suis adapté à cette modification du cahier des charges en cours de développement.

Afin d'éviter de revoir complètement la structure de ma BDD ainsi que celui de mon code, j'ai essayé de trouver la solution la moins chronophage tout en garantissant que ces fonctionnalités seraient intégrées au site. Je me suis inspiré du fonctionnement de certains codes-barres.



Figure 5 : schéma du fonctionnement du code-barre

Un code-barre ne se comporte pas comme un identifiant unique pour chaque produit, il comporte en réalité bien plus d'information. Comme on peut le constater sur la figure ci-dessus, un code barre se découpe en plusieurs segments, tous comportant une information différente. Le premier étant le pays d'origine, puis le pays du fabricant et ainsi de suite. Cela permet de stocker un grand nombre d'information dans un nombre relativement petit.

Si je reprends l'exemple d'un document qui est associé à plusieurs sous-catégories (donc à plusieurs catégories par association), disposer d'un « code-barre » pour chaque document me permet de stocker toutes ces informations dans une seule cellule. Cela m'évite la création d'une nouvelle table et de modifier en cascade toute ma base de données ainsi qu'une partie importante de mes fonctions. J'ai donc créé mes propres codes-barres afin qu'ils contiennent toutes l'information d'un document directement dans la table documents. Leur structure est bien plus simple que celle d'un code-barre classique.

categorie	sous_categories
003 002	006 007

Figure 6 : capture d'écran montrant la structure de donnée qui permet de créer une association multiple

Le principe est le suivant : à la mise en ligne on sélectionne les sous-catégories de ce document. Un document doit avoir au minimum une sous-catégorie et au maximum trois. Il a donc les mêmes contraintes du point de vue des catégories. Une fois les sous-catégories sélectionnées, de façon automatique, le document est associé aux bonnes catégories (une sous-catégorie n'appartient qu'à une seule catégorie). Une fonction concatène ensuite les ids des sous-catégories en ajoutant entre chaque id un séparateur, ici « | ». Cette fonction est native de php, [implode\(glue, tableau\)](#)²². Cela en résulte une chaîne de caractère qui est enregistrée dans la BDD.

Pour lire ce code et en extraire les données, il convient de dérouler le processus dans le sens inverse. Grâce à la fonction [explode\(délimiteur, chaîne\)](#)²³ qui scinde une chaîne de caractère en tableau à chaque fois que la fonction repère le délimiteur (ici « | »), on obtient un tableau qui contient tous les ids d'un document. A partir de ce tableau on peut retrouver à quelles sous-catégories appartient le document.

Ce processus est répété plusieurs fois dans le logiciel, toujours dans le même but.

²² <https://www.php.net/manual/fr/function.implode.php>

²³ <https://www.php.net/manual/fr/function.explode.php>

Fonctionnalités

Je détaille ici les fonctionnalités du site, ainsi que de leur aspect technique. Dans un souci de concision, je ne vais pas m'attarder sur les capacités « classiques » d'un tel site telles que sont l'ajout, la modification ainsi que la suppression d'entités (documents, utilisateurs, catégories ...). Je fais le choix de ne présenter que les aspects innovants dans le détail.

Le site, ainsi que toutes ses possibilités, est disponible en annexe ([annexe 5](#)), il suffit pour cela d'installer le code sur un serveur local, d'importer la BDD ainsi que de se connecter en tant que super-administrateur. Une fois cela fait, le site est disponible tel qu'il était à l'issue de mon stage. Comme détaillé dans les remerciements, ce n'est pas la version définitive, elle est encore en cours de développement. Je vais travailler trois semaines à l'issue de mon stage dans le but de leur rendre un site fonctionnel. Des pans du site ne sont pas encore terminés. Pour se référer, dans le détail, au travail que j'ai effectué, je vous invite à consulter le code pour avoir une vision exhaustive des fonctionnalités intégrées.

Ajout d'un document

L'ajout d'un document, ou l'upload, est intéressant à détailler. Cela représente une fonction que nous n'avons pas vu en cours, d'une part, ainsi qu'un travail méticuleux et organisé afin de garder le site synchronisé, d'autre part. Le fichier `upload.php` se trouve dans le répertoire `scripts` (ou `Modèle` dans mon architecture). Il gère la mise en ligne d'un fichier grâce à la présence d'un formulaire comportant l'attribut : « `enctype="multipart/form-data"` » qui permet la mise sur le serveur d'un fichier. Cette mise sur le serveur n'est pas à proprement parler la mise en ligne, puisque, dans un premier temps, le fichier est présent dans le répertoire `tmp/` du serveur, là où sont stockés les fichiers temporaires. Il faut ensuite préciser, depuis le répertoire `tmp/`, où doit migrer ce fichier. Sans quoi, il sera écrasé automatiquement. C'est cette étape de migration qui permet au fichier d'être pérenne.

Avant de gérer la migration de ce fichier, plusieurs informations doivent être testées. En effet, mettre un fichier depuis un client sur un serveur n'est pas anodin, outre le fait que cela peut présenter des failles de sécurités importantes, il faut contrôler aussi son nom, son poids etc.

Dans l'ordre sont testés :

- Si l'utilisateur est connecté
- Si un fichier a bien été envoyé grâce à la variable superglobale `$FILES`²⁴, plus spécifiquement, puisque cette variable est un tableau, grâce à la variable `$_FILES['file']['errors']` dont le code d'erreur spécifique au fait qu'il n'y a aucun document mis en ligne vaut 4.
- Si le fichier ne dépasse pas la taille maximale autorisée : 10Mo
- Si le fichier ne répond pas aux critères de catégories : par exemple s'il est dans trop de catégorie ou s'il se trouve dans deux catégories incompatibles entre elles.
- Si un fichier existe déjà avec le même nom. Le fait que deux fichiers aient le même nom n'est pas un problème pour la plateforme. En effet, le nom du fichier permanent situé dans le répertoire `documents/` est généré via l'id en BDD. Il est donc unique dans tous les cas. Cependant, sur la demande de Valérie Hudry, et pour des raisons de lisibilité, le nom du fichier qui est affiché sur le site (on appellera ça le libellé du document) doit être unique, d'où cette vérification.
- Si le libellé du fichier contient des caractères de contrôles comme « `/` » ou « `\` » par exemple.

²⁴ <https://www.php.net/manual/en/features.file-upload.post-method.php>

- Si le fichier, pour une raison quelconque ne se trouve pas dans le répertoire temporaire, tmp/. En d'autres termes, qu'il n'a pas été uploadé.
- Si le fichier, pour une raison quelconque ne peut pas être déplacé du dossier temporaire tmp/ au fichier permanent documents/. Cette vérification se fait via la fonction [move_uploaded_file\(\)](#)²⁵ qui, si elle retourne FALSE, indique que le fichier n'a pas pu être déplacé.

Comme décrit plus haut, le nom du fichier originel, présent sur la machine de l'utilisateur, est intégré à la BDD comme étant le libellé du document. En réalité, pour l'internaute, tout se déroule comme si c'était le véritable nom du document : l'affichage du nom sur le site, la recherche par « nom » du fichier, et même la prédiction de recherche par « nom » du fichier (cette partie sera détaillée dans la suite du rapport).

C'est seulement au moment de télécharger un fichier que le nom de ce fichier est [id].pdf, ce qui n'a pas été jugé dérangeant dans le cahier des charges.

Plusieurs autres données sont enregistrées dans la BDD, l'emplacement du fichier, la date de mise en ligne, la date du document (si on ajoute un document en mars 2021 alors qu'il date de mars 2020), l'id de l'utilisateur qui l'a mis en ligne, sa date de validité (optionnel) et par qui ce document est-il visible.

Pour comprendre comment un fichier est défini visible ou non, il faut comprendre que : chaque fichier est rangé dans chaque catégorie, en fonction des sous-catégories. En effet, Valérie Hudry souhaitait qu'on ne puisse sélectionner que les sous-catégories dans lesquelles se trouverait le fichier. En fonction des sous-catégories dans lesquelles le document se trouve, il est possible de remonter jusqu'aux utilisateurs qui ont accès à cette sous-catégorie. Il est donc possible de leur envoyer un mail pour les prévenir de l'ajout d'un document les concernant.

Ensuite, est stocké si le fichier est visible ou non (je détaille ce point plus bas) et si le fichier a été supprimé ou non.

Un fichier est supprimé par le biais de la fonction [unlink\(\)](#)²⁶ qui supprime un fichier. Ensuite, l'entrée en BDD correspondante à ce fichier est définie sur supprime : « oui ». Ce qui la rend invisible, même pour le super-administrateur. J'ai préféré une suppression douce pour l'entrée en BDD afin de garder une trace mais aussi pour ne pas avoir de trous dans mes id de documents.

Un fichier défini sur visible = oui est visible de tout ceux qui ont accès à ce document, c'est sa valeur par défaut. Cependant, à l'ajout du document on peut ajouter un paramètre en plus, la date de validité. En effet, il peut arriver que l'on veuille mettre un document en ligne et qu'il disparaisse à une date précisée à l'avance. C'est pour cela que cette colonne existe, quand visible = non le fichier est visible uniquement par le super-administrateur. Il peut alors choisir de modifier le document, soit en avançant sa date de validité, soit en la supprimant pour que ce dernier redevienne visible aux internautes, ou de ne rien faire, auquel cas le document reste invisible.

²⁵ <https://www.php.net/manual/en/function.move-uploaded-file.php>

²⁶ <https://www.php.net/manual/fr/function.unlink.php>

<div> <div>Organisation collective</div> <div>Réunions</div> <div>Documents</div> <div>Communication interne</div> <div>Actualité</div> <div>Agenda municipal</div> <div>Administration</div> <div>Mon Compte</div> </div>				
fichiers.sql				
Date doc.	Auteur	Dépôt		
17/06/2020	super-administrateur	17/06/2020		
drive.sql				
Date doc.	Auteur	Dépôt	Validité	
17/06/2020	super-administrateur	17/06/2020	05/04/2002	
index.html				
Date doc.	Auteur	Dépôt		
17/06/2020	super-administrateur	17/06/2020		
logd ftp.txt				
Date doc.	Auteur	Dépôt		
17/06/2020	super-administrateur	17/06/2020		
testV2.php				
Date doc.	Auteur	Dépôt		
19/06/2020	super-administrateur	19/06/2020		
PNG_transparency_demonstration_1.png				
Date doc.	Auteur	Dépôt	Validité	
19/06/2020	super-administrateur	19/06/2020	25/06/2020	

Figure 7 : capture d'écran montrant la vue du super-administrateur sur les documents dont la date de validité a été dépassée, ces documents sont en rouges.

Il est important de noter que les fichiers associés ne sont pas supprimés. Il est seulement impossible de s'y rendre si on n'est pas le super-administrateur.

Modification du style

Via CodeMirror, qui fournit un éditeur de code en ligne, il est possible de créer une interface intuitive dans le but d'intégrer du code au site web, cette interface est intégrée via deux fonctions afin d'éviter de surcharger la vue. L'interface est illustrée Figure 4 : capture d'écran de l'éditeur de code avec CodeMirror. Une fois le code rentré, un script vérifie le contenu du code, et grâce à la fonction [file_put_contents\(\)](#)²⁷ qui permet d'écrire des données dans un fichier, le CSS est ajouté dans un fichier CSS prévu à cet effet. Cette feuille de style étant la dernière à être intégrée au site, elle prend le dessus sur toutes les autres. Le contenu de ce fichier est intégré à l'interface de développement afin de pouvoir reprendre l'écriture du CSS en cours de route.

Dans le même esprit que la modification du style, j'ai développé une interface qui permet de modifier dynamiquement une feuille de style spécialement prévue à cet effet. Ce fichier CSS est inclus juste avant la feuille vue précédemment afin qu'elle prenne le dessus sur toutes les autres. Cela fonctionne comme suis, en amont j'ai créé une table spéciale dans ma BDD qui me permet de définir des valeurs pour ce que je souhaite rendre modifiable. Parmi ces valeurs, on peut trouver :

- La taille des différents titres
- La couleur des différents titres
- Trois couleurs de fond
- La couleur des tableaux
- Couleur des items du menu
- La couleur de bouton
- La couleur de boutons au survol

Chaque ligne de ma BDD se présente comme suis :

²⁷ <https://www.php.net/manual/fr/function.file-put-contents.php>

id	nom	valeur
1	couleur1	#1B4262

Figure 8 : capture d'écran représentant une entrée dans la table style

Grâce à une fonction que j'ai créé, `get_style_part(nom, donnée à afficher)`, il est possible de retourner le champ spécifié en précisant quelle donnée on souhaite. Par exemple `get_style_part('couleur1', 'valeur')` retourne `#1B4262`. Il est donc possible d'obtenir la valeur pour afficher dans l'interface quelle couleur est active pour le site.

Ensuite, il est possible de modifier cette valeur grâce à la fonction `modifier_style_par_id(nouvelle valeur, id)`. Enfin, un fichier PHP qui se trouve dans le répertoire Bootstrap, fait office de feuille de style dynamique. Pour que ce fichier soit interprété comme étant une feuille de style, il convient de réutiliser la fonction `header`, mais d'une autre manière : `header('content-type: text/css');`. Cette ligne fait donc comme si le fichier n'était pas PHP mais CSS, ce qui permet de l'inclure dans le header. Dans ce fichier il suffit d'afficher toutes les valeurs de style grâce à la fonction `get_style_part(nom, donnée à afficher)` vue précédemment. En combinant cela avec du CSS il est possible de créer une feuille de style dynamique. Dans un souci de réduire le nombre d'appel à la BDD, j'ai utilisé le [root CSS](#)²⁸. Cela permet de définir des variables en CSS en début de feuille pour pouvoir les réutiliser tout au long de la feuille de style. J'ai donc appelé toutes mes variables au début de la feuille de style, ne faisant qu'un appel pour chaque variable, elles sont ensuite réutilisées dans le fichier. Il ne reste ensuite qu'à combiner cela avec du CSS et la feuille de style est générée.

Modification du contenu

De la même manière que pour la modification du style graphiquement, j'ai intégré la possibilité de modifier le contenu du site internet directement depuis ce dernier. Cela, grâce à la fonction `get_content_part(balise, class, nom)` qui permet l'affichage du contenu du site stocké en BDD, mais aussi, lorsque l'on est super-administrateur, la modification de ce contenu.

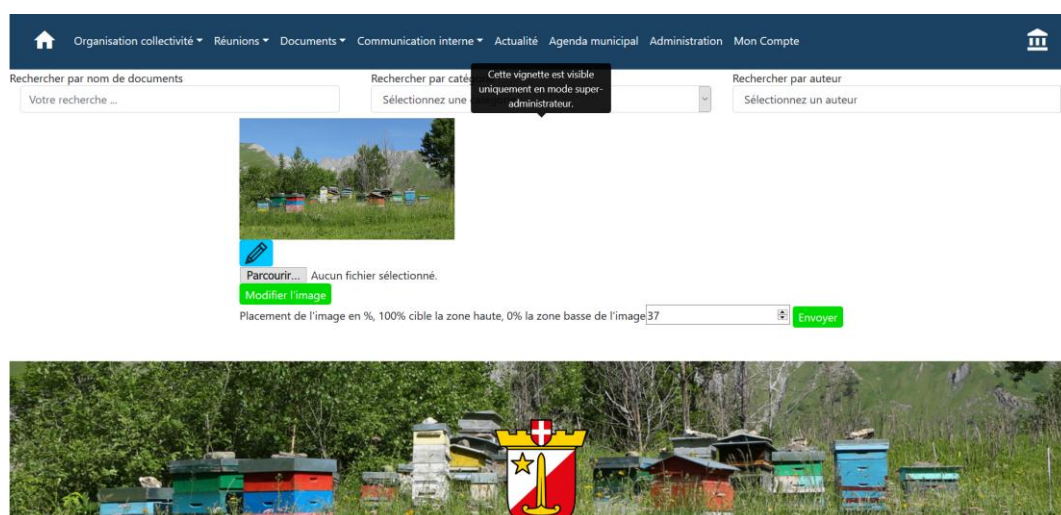


Figure 9 : capture d'écran montrant la modification du contenu en super-administrateur lors du survol d'une zone modifiable

²⁸ <https://developer.mozilla.org/fr/docs/Web/CSS/:root>

Lorsque l'on survole une zone modifiable et que l'on est super-administrateur, un crayon bleu apparaît. Si on clique dessus, cela déroule un formulaire qui permet de modifier le contenu. Ici, il est possible de modifier l'image qui apparaîtra en dessous, ainsi que de modifier sa zone visible. L'interface est toujours la même, que ce soit pour modifier un texte ou une image.

Toutes les données sont présentes dans la table « contenu ». Elle se structure comme suis :

id	nom	texte	href	class	groupe
17	footer_item	https://www.valthorens.com/fr/	vue_part/images/footer3.png	content_part	footer_item

Figure 10 : capture d'écran représentant la structure de la table « contenu »

Je détaille plus loin la fonction de l'id, le nom sert à appeler depuis la fonction get_content_part() la bonne ligne, le texte est ici le lien vers lequel renvoie le logo dans le footer, href est le chemin pour aller chercher l'image du logo, tous les deux modifiables, la class permet de styliser tous les éléments qui sont modifiable, et le groupe permet de cibler les éléments d'un même groupe, ici toutes les images du footer.



Figure 11 : capture d'écran du footer du site

L'id me sert à modifier le contenu avec un seul script. En effet, il arrive que le contenu soit parfois du simple texte, parfois des images, parfois des images avec liens. Afin de permettre une telle souplesse, j'utilise l'id de chaque contenu pour le modifier en fonction de sa nature.

Par exemple si le contenu est seulement du texte, les vérifications faites dans le cas de la mise en ligne d'une image sont à proscrire, et réciproquement. Dans le cas d'une image, le traitement est un peu plus subtil. A titre d'exemple l'id 27 correspond au favion du site internet, il faut donc le mettre en ligne avec un nom différent que pour l'id 15 qui est la première image du footer.

Les notifications

Le système de notification étant central dans l'efficacité du site internet, il est important de détailler son fonctionnement. Tout d'abord il est possible, lorsqu'on est super-administrateur d'envoyer un mail ciblé à une ou plusieurs listes d'utilisateurs, et/ou de cibler un ou plusieurs utilisateurs. Cependant, ce ne sont pas ces notifications-là qui sont le plus intéressantes à détailler.

Il est à noter que les utilisateurs peuvent choisir ou non d'être notifiés par mail lors de l'ajout d'un document les concernant. Par défaut, la case « Accepter les notifications par mail » est cochée, chacun peut la décocher.

Lors de l'ajout d'un document, les sous-catégories sélectionnées sont reliées aux catégories d'utilisateurs qui peuvent les consulter. Une fois celles-ci récupérées par le biais d'une fonction PHP, il devient possible de lister tous les mails des utilisateurs présents dans toutes les listes utilisateurs et qui acceptent les notifications par mail. La fonction [`mail\(destinataire, sujet, message\)`](#)²⁹ permet d'envoyer un mail. Les variables sujet et message sont des constantes définies dans le fichier `functions.php` via un appel à la BDD récupérant la configuration du site internet.

Le formulaire de recherche par libellé du document

Le nombre de documents étant important, il était nécessaire d'ajouter des fonctions de recherche. J'ai donc intégré des fonctions de recherche classiques comme une recherche par catégorie ou encore par auteur du document. Cependant, j'ai trouvé intéressant de pousser la fonctionnalité plus loin. En effet, à la manière de Google lorsque l'on fait une recherche et que celle-ci comporte une faute, le moteur de recherche nous propose un résultat proche de ce dernier afin de tenter de corriger la recherche. Je me suis donc renseigné afin de savoir si PHP proposait une fonction permettant de genre de chose. J'ai trouvé la fonction [`similar_text\(chaine1, chaine2\)`](#)³⁰ qui permet de calculer la similarité entre deux chaînes.

En faisant de cette fonction le cœur de mon moteur de recherche, j'ai créé la mienne. Dans un premier temps, elle liste tous les documents visibles et non supprimés (elle devra dans le futur comporter la condition de ne lister que les documents accessibles à celui qui fait la recherche). Pour chaque document elle compare un libellé avec la recherche de l'internaute. En ajoutant un troisième paramètre à `similar_text()`, la fonction va calculer la similarité en pourcentage des deux chaînes de caractère. Cette donnée est ensuite comparée à une valeur arbitraire (en %). Si la ressemblance est supérieure à la valeur arbitraire, alors la ligne en BDD qui correspond à ce document est intégrée dans un tableau. Ce tableau est ensuite classé par ordre décroissant, pour que les résultats les plus pertinents apparaissent en premier, puis envoyé à la vue.

²⁹ <https://www.php.net/manual/fr/function.mail.php>

³⁰ <https://www.php.net/manual/fr/function.similar-text>

Analyse économique

Les retombées économiques d'un tel logiciel étant difficilement quantifiables, j'ai préféré regarder du côté du prix de conception de ce genre de système. Selon [neuvoo](https://neuvoo.fr/salaire/?job=Developpeur%20Php)³¹, un développeur PHP débutant gagne environ 19.2€ par heure. En considérant que les développeurs PHP gèrent la totalité du site, back-end et front-end, que je suis considéré comme un débutant, et que le chiffre donné plus haut correspond à un salaire net horaire (ce qui peut compenser le fait que je ne sois pas considéré encore comme un développeur PHP débutant) alors que l'organisme paie un salaire horaire brut (+25% environ), on peut garder cette donnée pour une estimation grossière.

Pour un développeur débutant, ce site leur aurait donc coûté $9 \times 35h = 315h$ $315 \times 19.2 = 6\,048\text{€}$.

Cependant, le site internet n'est pas encore terminé. Je pense estimer à encore deux semaines le temps de développer les dernières fonctionnalités, corriger les bugs, terminer l'interface graphique, mettre le site internet en ligne et rédiger une petite documentation. Il faut donc ajouter à ce prix $2 \times 70h = 140h$ $140h \times 19.2 = 2\,688\text{€}$ $2\,688\text{€} + 6\,048\text{€} = 8\,736\text{€}$. Ce site devrait donc être facturé de l'ordre de 8 000-9 000€.

Afin de compléter cette analyse économique, il peut être intéressant de montrer ce que la commune a déboursé pour ce site internet. J'ai été payé au taux horaire de 3.75€ net de l'heure pour une durée d'exactly 315h, $315 \times 3.75 = 1\,181.25\text{€}$. A cela, puisque le site n'est pas terminé, il faut ajouter deux semaines de plus au smic soit environ 10€/heure brut. On arrive à $1\,181.25 + 700 = 1\,881.25\text{€}$. C'est ce que la commune a déboursé pour la réalisation de ce site internet.

Néanmoins il faut prendre ces chiffres avec beaucoup de recul, étant donné ma situation de stagiaire, du temps que j'ai pu perdre par manque d'expérience, ainsi que des multiples approximations dans mes calculs. Il faut aussi souligner que ce stage m'a permis de m'améliorer, il n'avait pas pour but de générer de gros revenus. Mes estimations peuvent être largement critiquables, cependant, il était stipulé dans le cahier des charges qu'une analyse financière était obligatoire.

Analyse humaine

Mes relations avec les personnes qui m'ont entouré dans le service ont toujours été bonnes. J'ai déjà pu rencontrer certaines de ces personnes par le passé, mais j'ai véritablement appris à connaître Valérie Hudry durant ce stage. La collaboration s'est toujours très bien passée avec elle, puisque c'est avec elle que j'ai travaillé durant la grande majorité du stage. J'ai aussi rencontré d'autres personnes du service à qui, je l'espère, j'ai donné une bonne image. Les rencontres avec monsieur le maire ainsi que les personnes qui l'entouraient durant les réunions se sont toujours bien déroulées. Le contact d'une administration est toujours intéressant, il permet de mieux comprendre ses rouages. De plus, cette plateforme sera utile aux élus, ce qui valorise mon travail.

J'ai appris à communiquer avec des personnes qui ne sont pas du même milieu professionnel que moi, ni de la même génération, il a fallu que j'expose et explique mes travaux, et cela a été une bonne expérience. Le fait que je sois plus jeune a permis d'apporter une nouvelle vision sur les problématiques que le service rencontrait. J'espère y avoir apporté un œil neuf et frais. Dans le même temps, cette différence d'expérience m'a aussi permis de me confronter à des personnes expérimentées dans leur domaine professionnel. J'ai beaucoup appris à leur contact et je les en remercie.

³¹ <https://neuvoo.fr/salaire/?job=Developpeur%20Php>

Conclusion

Difficultés rencontrées

La difficulté la plus importante a été, sans aucun doute, le changement de cap en cours de développement. En effet, après une réunion avec le maire, quelques élus ainsi que des agents des services, plusieurs changements ont été décidés. La majorité de ces ajustements étaient bénins pour moi, cela se résumait à des modifications de la mise en page, de style ou encore de fonctionnalités. Cependant certaines de ces modifications insinuaient de revoir complètement la structure de ma BDD ainsi que celle de mon code. Par exemple, un document n'était plus associé à une seule catégorie mais à plusieurs, il fallait aussi créer des sous-catégories qui doivent être liées à leur catégorie mère. Enoncer toutes les modifications structurelles serait fastidieux, ce qu'il faut retenir c'est que cela m'a demandé de m'adapter rapidement et de revoir complètement mon organisation. J'ai dû faire preuve d'imagination, de patience ainsi que d'écoute afin d'être certain que j'allais dans la bonne direction. C'est sans conteste la difficulté la plus importante que j'ai eu à surmonter.

Ensuite, la communication avec les élus, pas toujours familiers avec les nouvelles technologies, s'est parfois révélée plus compliquée que ce que j'aurai imaginé. Il a fallu surmonter cette barrière afin que nous soyons tous sûrs que le développement du logiciel se poursuivait dans la bonne direction. Le confinement a aussi été une barrière supplémentaire, il induisait des échanges plus lents entre les personnes. Cependant, Valérie a toujours été réactive pour moi et ça n'a donc pas été une grosse difficulté.

Enfin, les délais du rapport de stage étaient très courts. En effet, j'ai terminé mon stage le 19 Juin, le rapport étant à rendre 10 jours après. Initialement le délai était de plus de deux semaines ce qui était un laps de temps confortable. Cependant, mon stage ayant été décalé d'une semaine, j'ai eu moitié moins de temps pour rédiger un rapport conséquent. J'espère avoir produit un contenu de qualité et bien documenté durant le temps que j'y ai consacré. Cela, en plus de devoir régler des points de détails durant la phase de test du site internet qui s'est déroulée du 22 au 26 Juin.

Ce que j'ai acquis pendant ce stage

Pendant que j'ai rédigé les documents relatifs à la communication sur les RS, je n'ai pas beaucoup appris. En effet, les données que j'ai recherché sur le web ont validé ce que je connaissais déjà étant donné les cours que j'ai suivis. Cependant j'ai progressé sur la suite Office, j'ai découvert de nouvelles fonctionnalités que j'ai réutilisé pour ce rapport.

Néanmoins, le constat est radicalement différent pour la partie développement web. J'ai en effet passé beaucoup de temps à lire la documentation PHP afin de découvrir de nouvelles fonctions. J'ai beaucoup appris sur les possibilités de PHP qui offre tellement plus qu'un simple langage de programmation serveur. J'ai aussi beaucoup progressé en POO, la réalisation de ce site m'a permis de mettre en pratique mes cours dans ce domaine mais aussi de comprendre ce qui m'avait été enseigné. La pratique m'a beaucoup aidé à comprendre le fonctionnement d'un programme orienté objet.

Au niveau de l'organisation, j'ai aussi pu retenir certaines choses. Il y a des erreurs que je ne referai pas, comme l'absence de modélisation UML ainsi que des schémas pour la BDD. J'ai parfois développé

la partie graphique avant le traitement de donnée, ce qui n'est pas optimal. Je ne me suis pas assuré que « le client » avait la même vision du logiciel que moi, ce qui a entraîné des difficultés supplémentaires de mon côté.

La méthode télétravail était une grande première pour moi et je suis satisfait de ce que j'ai proposé. Cela suppose une discipline plus dure mais aussi une organisation de son travail plus exigeante.

Pour conclure, j'ai adoré travailler pour la mairie. J'ai le sentiment d'avoir construit quelque chose d'utile et qui servira pendant longtemps. Mettre mes compétences à disposition des élus est pour moi un honneur et me dire que mon code sera utilisé afin de faciliter l'organisation de cette institution est pour moi une fierté. Le stage est terminé mais la mairie m'a proposé de m'embaucher du 29 Juin au 17 Juillet. C'est une proposition que j'ai naturellement acceptée et qui va me permettre de terminer le projet sur lequel je travaille depuis maintenant 9 semaines. Il me sera demandé de terminer la plateforme, de rédiger de la documentation mais aussi de présenter le logiciel à l'ensemble du conseil municipale, de l'expliquer et bien sûr de le mettre en ligne. J'attends le moment où je pourrai me remettre à travailler dessus avec impatience.

Annexes techniques

Annexe 1 : [Budget communication 2020.xls](#)

Annexe 2 : [Presentation-service-com-juin2020.pptx](#)

Annexe 3 : [Organigramme complet juin 2020-BAT.pdf](#)

Annexe 4 : [Cahier des charges intranet de la commune.docx](#)

Annexe 5 : [mairie_smb/](#)

Annexe 6 : [administratif.pdf](#)

Diagramme de GANTT



Figure 12 : diagramme de GANTT

Ce diagramme illustre parfaitement le fait que plus de 80% de mon stage a été du développement web, comme initialement prévu.

Table des figures

Figure 1 : schématisation du fonctionnement de mon code	20
Figure 2 : structure des répertoires & fichiers du site en vue simplifiée	21
Figure 3 : capture d'écran du panneau de contrôle des autorisations pour les différents utilisateurs	24
Figure 4 : capture d'écran de l'éditeur de code avec CodeMirror	25
Figure 5 : schéma du fonctionnement du code-barre	26
Figure 6 : capture d'écran montrant la structure de donnée qui permet de créer une association multiple	27
Figure 7 : capture d'écran montrant la vue du super-administrateur sur les documents dont la date de validité a été dépassée, ces documents sont en rouges.	30
Figure 8 : capture d'écran représentant une entrée dans la table style	31
Figure 9 : capture d'écran montrant la modification du contenu en super-administrateur lors du survol d'une zone modifiable	31
Figure 10 : capture d'écran représentant la structure de la table « contenu »	32
Figure 11 : capture d'écran du footer du site	32
Figure 12 : diagramme de GANTT	37

Bibliographie

(s.d.). Récupéré sur

https://fr.wikipedia.org/wiki/Liste_de_communes_fran%C3%A7aises_par_superficie

bootstrap.com. (s.d.). Récupéré sur <https://getbootstrap.com/>

codemirror.net. (s.d.). Récupéré sur <https://codemirror.net/>

<https://neuwoo.fr/salaire/?job=Developpeur%20Php>. (s.d.).

material.io. (s.d.). Récupéré sur <https://material.io/resources/icons/?style=baseline>

michalsnik. (s.d.). Récupéré sur <https://michalsnik.github.io/aos/>

mozilla. (s.d.). Récupéré sur <https://developer.mozilla.org/fr/docs/Web/CSS/:root>

neuwoo. (s.d.). Récupéré sur <https://neuwoo.fr/salaire/?job=Developpeur%20Php>

php.net. (s.d.). Récupéré sur <https://www.php.net/manual/fr/function.header.php>

php.net. (s.d.). Récupéré sur <https://www.php.net/manual/fr/function.header.php>

php.net. (s.d.). Récupéré sur <https://www.php.net/manual/fr/function.define.php>

php.net. (s.d.). Récupéré sur <https://www.php.net/manual/fr/function.crypt.php>

php.net. (s.d.). Récupéré sur <https://www.php.net/manual/fr/book.password.php>

php.net. (s.d.). Récupéré sur <https://www.php.net/manual/fr/function.password-hash.php>

php.net. (s.d.). Récupéré sur <https://www.php.net/manual/fr/function.htmlspecialchars.php>

php.net. (s.d.). Récupéré sur <https://www.php.net/manual/fr/function addslashes.php>

php.net. (s.d.). Récupéré sur <https://www.php.net/manual/en/features.file-upload.post-method.php>

php.net. (s.d.). Récupéré sur <https://www.php.net/manual/en/function.move-uploaded-file.php>

php.net. (s.d.). Récupéré sur <https://www.php.net/manual/fr/function.unlink.php>

php.net. (s.d.). Récupéré sur <https://www.php.net/manual/fr/function.file-put-contents.php>

php.net. (s.d.). Récupéré sur <https://www.php.net/manual/fr/function.mail.php>

php.net. (s.d.). Récupéré sur <https://www.php.net/manual/fr/function.similar-text>

php.net. (s.d.). Récupéré sur <https://www.php.net/manual/fr/function.implode.php>

php.net. (s.d.). Récupéré sur <https://www.php.net/manual/fr/function.explode.php>

Wikipédia. (s.d.). Récupéré sur

https://fr.wikipedia.org/wiki/Liste_de_communes_fran%C3%A7aises_par_superficie

L'organisme d'accueil atteste de la lecture de ce rapport par la présente signature

Fait à :

Le :

Signature

