

GRAPHISTE & CODEUR

Nicolas Martineau
ASIW7 07/2018

LA RECHERCHE D'UN ÉQUILIBRE



Nicolas Martineau
ASIW7 2016-2018

**GRAPHISTE & CODEUR,
LA RECHERCHE D'UN ÉQUILIBRE**

présenté par Nicolas Martineau
Sous le tutorat de Guillaume Bernon

Mémoire présenté le 13 juillet 2018
Mastère 1 Design Graphique

Attestation d'authenticité

Ce document rempli et signé par l'étudiant doit être inséré dans tous les mémoires après la page de garde.

Je, soussigné(e) : (Nom et Prénom)

Nicolas Martineau

Etudiant(e) de : (formation et année)

ASIW/7 2016-2018

Etablissement :

IPSSI

Certifie que le texte soumis ne comporte aucun passage ou schéma copié sans qu'il soit explicitement référence aux sources.

Fait à **Paris** le **11/06/2018**

Signature :



IPSSI : INSTITUT PRIVÉ SUPÉRIEUR DES SYSTÈMES D'INFORMATION

SARL au capital social : 105 000 € - RCS PARIS : 420 793 705 - CODE NAF 8559A - Déclaration d'existence n° 117 533 621 75 - N° TVA : FR 41 420 793 705

3 Campus : • Paris : 25, rue Claude Tillier – BP 20024 - 75560 Paris Cedex 12 - Tél : 01 55 43 26 65 - Fax : 01 55 43 26 64

• Marne-la Vallée : 14, avenue de l'Europe – 77144 Montévrain • Brest : 297, rue Eugène Berest 29200 Brest – Tél. : 02 57 52 04 31

contact@ip-formation.com - www.ecole-ipssi.com

JE TIENS À REMERCIER L'AGENCE DE COMMUNICATION ARS MAGNA

De m'avoir accueilli durant deux ans.

Pour m'avoir fait confiance en m'offrant une rare liberté d'action.

Et plus particulièrement mon tuteur Guillaume Bernon pour son attention à mon égard.

JE REMERCIE ÉGALEMENT L'ÉQUIPE D'IPSSI

Pour leur mise en relation avec Ars Magna et les intervenants rencontrés.

Ainsi que toute ma promotion ASIW7 pour nos échanges durant ces deux ans,

Et en particulier Clara Nigen pour sa participation au test UX.

ET UNE ATTENTION TOUTE PARTICULIÈRE

À mon chat Réglisse pour sa présence, son amour et sa patience.

AVANT-PROPOS

Après une dizaine d'années dans le secteur audiovisuel, une lourde hospitalisation a fait dérailler ma situation professionnelle, survenue en parallèle d'une importante coupe budgétaire chez mon principal employeur. Un ensemble de traitements cliniques inappropriés ont eu pour effet de limiter mes fonctions cognitives, et impactant mes capacités d'embauche dans un secteur de plus en plus sinistré.

Libéré de cette errance au bout de quelques années, une page se tourne et me propose de choisir une direction nouvelle. C'est dans cet état d'esprit associé au regain de mes capacités intellectuelles que je suis entré à l'IPSSI en formation de designer graphique. Avec une place en entreprise, me voici bien décidé à retrouver le goût de l'apprentissage et de la découverte.

Ce mémoire est le premier auquel je me confronte, il s'avère être une part de cette exploration. Concluant deux années de formation, me rapprocher ainsi de la ligne d'arrivée m'incite à prendre de la distance sur le chemin parcouru. Avec l'impression d'être ressorti d'une tempête de sable il y a déjà 7 ans, le besoin d'alimenter une recherche continue et stimulante se heurte au besoin d'une stabilité générale.

Elle implique de la réflexion sur les possibilités d'évolutions de ma situation professionnelle, héritant d'un bagage graphique aujourd'hui fortement attiré par le code. Cette recherche d'un nouvel équilibre a démarré dans mon entreprise en juillet 2016.

ARS MAGNA	12
NOUVEAU POSTE, NOUVELLES MISSIONS	14
ANALYSE DE LA SITUATION	16
LE SITE ACTUEL	18
REPENSER LE SITE	20
LA MAQUETTE	30
INTRODUCTION AU DÉVELOPPEMENT	38
RÉALISATION DU SITE PUBLIC	42
1) HEADER & FOOTER	42
2) PAGE D'ACCUEIL	44
A) Slider	44
B) Publications	47
C) Projets récents	47
3) PORTFOLIO	48
A) Page générale	48
B) Page projet - générique	50
C) Page projet - motion design	52
D) Page projet - digital	54
E) Page projet - divers détails	56
4) CONTACT	58
5) QUI NOUS SOMMES ?	60
6) RESPONSIVE OU ADAPTATIF ?	61
RÉALISATION DU BACK-OFFICE	64
1) INTRODUCTION	64
2) DESCRIPTION DES RUBRIQUES	66
A) Slider	66
B) Publications	71
C) Portfolio	73
i) Vidéo	74
ii) Images optionnelles	75
iii) Catégories	77
3) DES DÉTAILS SIMPLIFIANT LA VIE	80
A) Date, affichage et utilisation	80
B) Système de pagination	80
C) Découpage du code	82

DÉTAILS DU BACK-END	84
1) BASE DE DONNÉES	84
2) SÉCURITÉ	87
A) Get & Post	87
B) Les injections malveillantes	87
C) Session et mot de passe	88
D) HTTPS	90
E) Un back-office en local ?	90
F) Cacher l'extension .php	91
G) Pour aller plus loin	91
3) CONFIGURATION DU SERVEUR APACHE	91
4) ERREUR 404 :(.....	92
5) OPTIMISATION DES PERFORMANCES	94
A) L'ordre des conditions	94
B) Les requêtes SQL	94
C) Des fichiers images miniatures	96
6) ORGANISATION GLOBALE DU PROJET	96
A) Le code	96
B) Les fichiers	97
ET LE SEO ?	100
TEST UX	102
BILAN	104
GLOSSAIRE	108

ARS MAGNA

Depuis 2014, Ars Magna est une agence de communication ayant des activités majoritairement dans l'industrie de la musique et du spectacle. Sa clientèle s'étend progressivement vers d'autres domaines avec un centre de formation dans le domaine médical, une agence de courtiers en immobilier, une comédienne, un chef de cuisine ou encore un centre de yoga.



Les services proposés par l'agence vont de la réflexion à la conception de packaging pour des albums CD et/ou vinyles, des collections de films en DVD, des affiches de publicité ou des bannières web, des catalogues, des shooting photos et des captations vidéo, l'organisation de journées de célébration et enfin la mise en place et l'entretien de site web.

La clientèle de l'industrie du disque, principalement des majors comme Warner Classics et Sony Music, est héritée de "Vue intégrale", une ancienne agence dont les employés d'Ars Magna ont fait partie durant 15 ans avant sa fermeture. Il y a donc un profond lien relationnel de confiance qui relie Ars Magna à sa clientèle musique. L'ambiance familiale se dégageant des locaux crée souvent une atmosphère accueillante et agréable pour les clients qui nous visitent.

L'agence comprend 4 employés d'orientation Print, ce qui en fait une petite structure, mais où la polyvalence est de mise. Ils sont ainsi en constante formation et ont entre autres appris l'usage du CMS WordPress pour la conception de sites internet. Ils sont donc ouverts à de nouveaux apprentissages tout en constatant les limites d'une trop grande polyvalence et du manque de temps pour l'approfondissement technique.



De 10h à 18h, les horaires sont classiques, mais plutôt flexibles. En fonction des besoins de certains projets, il peut y avoir besoin d'élargir ces horaires. Il m'est arrivé de rester plus tard le soir ou d'arriver plus tôt un vendredi matin pour assurer les délais de quelques rares urgences.

Ayant été intermittent du spectacle durant une dizaine d'années principalement pour Lagardère Active, grand groupe du secteur audiovisuel, mon arrivée chez Ars Magna a nécessité un temps d'adaptation. Autant d'ordre relationnel au sein d'une petite équipe, que de mon emploi du temps fixe et régulier ou de la présence d'un poste de travail dédié.

Le logo représente une pyramide trouvant son équilibre à travers les compétences de chacun de ses membres. Il symbolise donc diversité, richesse et solide coopération.



NOUVEAU POSTE, NOUVELLES MISSIONS

La création de mon poste chez Ars Magna exprime la volonté de proposer à la clientèle un nouveau service dédié à la production vidéo, notamment pour des video lyrics⁽¹⁾ en lien avec l'industrie du disque. Ayant une expérience de monteur et de motion designer d'une petite dizaine d'années, mon profil correspondait parfaitement aux besoins d'Ars Magna. Aussi, ma formation en design graphique enrichira les services que je pourrais apporter, notamment dans l'usage avancé des CMS de leurs projets web.

Ma première mission a été de réaliser une video lyrics pour le dernier single d'Amir, "On dirait" sous la houlette de Guillaume, graphiste de l'agence déjà en charge de la conception du packaging des albums d'Amir. Par la suite, j'ai eu l'occasion d'en produire trois autres dans des conditions de productions plus restreintes et plus proches des intentions de l'agence : emploi de photos issues de shooting et respect de la charte du packaging des albums. Le but étant de produire en une semaine une video lyrics efficace et agréable.



(1) Une séquence vidéo dans laquelle les paroles d'une chanson sont synchronisées sur la musique pour être mises en avant. Cela requiert un travail d'animation typographique et de mise en situation du texte. Il est souvent employé comme un pré-clip à moindres frais, conçu pour entretenir l'intérêt du public avant la diffusion du clip officiel. Les video lyrics étant également souvent conçus par des fans, les maisons de disque préfèrent mettre en avant leurs propres versions qu'ils jugent plus valorisantes.

J'ai également pu répondre à un certain nombre d'autres missions incluant du motion :

- La réalisation fréquente de vidéos corporate à destination des clients.
- La création d'une installation vidéo accompagnant l'inauguration des nouveaux locaux d'une entreprise. Des tablettes de type iPad, déguisées en tableaux dont chacune projetait une photo d'un des membres de leur équipe... se mettant subitement à bouger. Ces petites vidéos en boucle disposées dans les couloirs de l'agence ont égayé la soirée.
- De l'authoring DVD pour Céline Dion. Préparation des menus pour un DVD en respectant l'ambiance graphique de son dernier album pour intégrer un karaoké.
- L'animation des illustrations d'une graphiste de l'agence, pour donner vie à des dessins d'animaux.

Dans un autre registre, j'ai pu également :

- Créer des fichiers de sous-titre pour des vidéos à destination de YouTube et Facebook.
- Concevoir des affiches animées pour des écrans digitaux.
- Produire des bannières animées en GIF ainsi que des landing page sous forme de maquette Photoshop.
- Modéliser des vinyles sous Cinema 4D pour créer des packshots de covers d'albums.
- Assister un graphiste dans la modification d'un catalogue et pour la production de visuels grand format sous Indesign.

Enfin, en lien avec ma formation en webdesign, j'ai pu :

- Prendre en charge le développement front-end d'un site web responsive.
- Écrire en JavaScript un questionnaire dynamique pour conseiller les clients d'une agence de Yoga, nécessitant d'être intégré au sein de WordPress.
- Concevoir des eCard et eMailing.

Pour les besoins de l'agence, j'ai dû acquérir des notions de base de Cinema 4D et produire des séquences en 3D. J'ai ainsi réalisé un exercice de modélisation d'un album CD de l'agence que j'ai ensuite animé en synchronisation sur la musique de l'album. Le résultat a plu autant à mes collègues qu'à de potentiels clients intéressés par la démarche.



ANALYSE DE LA SITUATION

Malgré toutes ses activités, peu de missions m'ont été proposées en l'espace de deux ans. Aussi nous avons élaboré plusieurs solutions afin d'alimenter le service motion :

- Appuyer les propositions de production vidéo aux clients lors des réunions si le projet s'y prêtait, car parfois, ils ne sont simplement pas au courant de nos capacités
- Alimenter l'actuel site de l'agence et les réseaux sociaux avec plus de contenu vidéo
- Produire la communication client de l'agence sous forme de courtes vidéos afin de les sensibiliser
- Ars Magna organise souvent des soirées dans ses locaux aux profits de ses clients pour favoriser la rencontre et les échanges. Des vidéos remémorant les travaux de l'agence sont diffusés en boucle pour créer de l'animation et communiquer sur nos compétences.

Malgré ces tentatives, beaucoup d'espaces blancs se sont étalés entre mes missions. L'idée de rester devant une page Facebook en attendant une activité peut paraître tentante pour certains, mais a tendance à me déprimer.

J'ai investi ce temps libre pour approfondir mes connaissances des principaux langages du web que sont HTML, SCSS, JavaScript, PHP, MySQL et Apache.

Ma motivation première a été un intérêt personnel croissant pour le code. Ayant auparavant eu l'occasion de m'initier au C++⁽¹⁾, les langages du web me sont de fait apparus abordables.

L'idée d'accroître mes compétences afin d'offrir à mes employeurs de nouvelles circonstances opportunes de missions a également été l'une de ses raisons.

Le code est une nécessité évidente dans les métiers du web.

Enfin, j'ai souvent eu une nature autodidacte m'ayant dirigé là où j'éprouvais du plaisir. Ce qui était déjà le cas pour le motion design alors que j'étais originaire d'une formation de monteur.

(1) Le C++ est un langage de programmation apparu en 1983 et destiné à concevoir des logiciels. C'est un langage dit «bas niveau» qui ne fonctionne pas depuis un navigateur internet ou un serveur, car il n'a pas été conçu pour le web et nécessite par ailleurs d'être compilé. Le sigle «++» après le «C» signifie que le C++ est une incrémentation (ou amélioration) du langage C. Tout un vocabulaire prouvant que le code recouvre plusieurs galaxies !

Cette situation m'a conduit vers un projet personnel se résumant à une refonte complète du site web de l'agence. Tout d'abord parce qu'il me permet de tester mes acquis et de me confronter à la réalité du code mais aussi parce qu'il reste dans l'objectif d'aider l'agence à communiquer sur ses capacités. Il y a au travers de ce projet un grand nombre de réflexions qui vont de la stratégie de communication au design, à l'ergonomie et au développement. Difficile de ne pas s'y perdre. J'ai donc laissé mes envies me guider dans le choix des priorités.

Avant de me lancer dans le code, plusieurs questions et réflexions préliminaires ont été nécessaires. Je vous propose donc de découper ce projet en plusieurs parties :

- En premier, un état des lieux de l'actuel site s'impose. De son contenu à l'usage de WordPress qui en fait son socle.
- Une réflexion sur les améliorations de la hiérarchisation du contenu et de l'ergonomie générale et de l'usage du responsive web design.
- La création d'une première maquette.
- Les différentes phases de développement qui m'ont semblé pertinent de vous partager.
- Enfin, une session de test du nouveau site se concluant par un bilan du travail effectué.

Ce projet personnel n'avait pas forcément comme objectif final de remplacer l'actuel site internet de l'agence — d'ailleurs remplacé depuis par un nouveau thème WordPress — mais j'ai envisagé ma motivation sous l'angle d'une visée réaliste, d'un projet concret et palpable pour me confronter aux difficultés que l'aspect théorique ne me permettrait pas d'effleurer.

Bien que cette réalisation concrète implique une exigence de qualité, cette dernière est limitée par mes connaissances en code. L'intérêt de ce projet étant de m'initier plus profondément au développement web, j'ai donc envisagé les étapes de réflexion et production de ce site comme un équilibre entre méthodologie professionnelle et plaisir de découverte.

LE SITE ACTUEL

N. B. Mon état des lieux était basé sur l'ancien site d'Ars Magna qui a été récemment remplacé. Je faisais donc ici référence à l'ancien site. Je ne peux malheureusement pas illustrer mes observations faute de captures d'écran.

Le site présentait alors trois types de contenu :

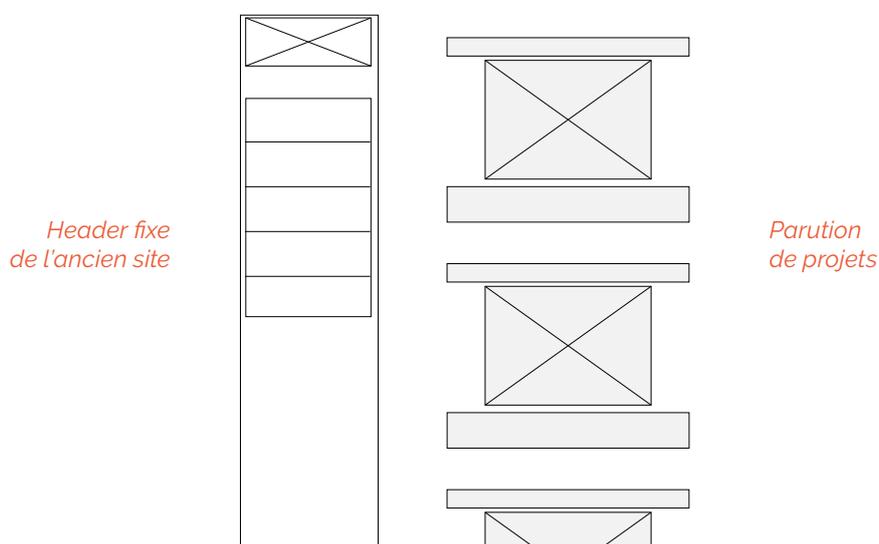
- Le travail de l'agence, le principal contenu, car le site est avant tout un portfolio en ligne.
- Une présentation de l'agence, de ses membres et de l'état d'esprit général.
- Une page de contact.

Beaucoup de sites concurrents disposent également d'une page de tarifs, allant jusqu'à proposer des packs tarifaires. Ars Magna a fait le choix de ne pas en proposer pour appuyer sur une relation client moins mercantile et plus chaleureuse.

Le portfolio se présentait sous la forme d'un blog où chaque publication était un projet. Chaque publication se retrouvait par la suite rangée dans une rubrique Portfolio séparée en plusieurs catégories. Chaque projet dispose donc de sa propre page avec plusieurs visuels. Leur mise en page est générique, qu'importe le type de projet. D'où un éventuel problème de valorisation de certains d'entre eux.

Dans son ensemble, le site ressemble plus à un blog qu'à un portfolio d'agence. Ce qui s'explique en partie, parce qu'il a été conçu sous WordPress.

Un choix justifié par le manque de connaissance dans les langages du web malgré le besoin évident d'un site. Motivé par l'apparente simplicité de WordPress, un membre de l'équipe s'est formé à son utilisation. L'usage d'un CMS comme WordPress a permis d'obtenir rapidement un système de blog prêt à l'emploi. Aux vues de la relative simplicité de mise en place, la capacité de production est séduisante.



Bien que la mise en place d'un site soit simple et rapide, cela ne va pas sans quelques contreparties en fonction des besoins utilisateurs.

Dans le cas d'Ars Magna, j'ai constaté les problèmes suivants :

- Les fonctions de personnalisation esthétique proposées par WordPress ou le thème utilisé ne sont pas toujours suffisantes. L'usage d'un CSS personnalisé devient très vite une obligation. On se retrouve alors dans une situation où l'on tente d'écraser un code préexistant pour atteindre ses objectifs. Ce qui rend l'environnement de travail très brouillon et à la maintenance inutilement compliquée.
- L'ajout d'un slider en page d'accueil sur le thème WordPress choisi par l'agence n'est pas une tâche aisée, ce qui est source de frustration.
- WordPress dispose de sa propre logique UI/UX que tous les concepteurs de thèmes de respectent pas forcément. En utilisant un grand nombre de plug-ins, on se retrouve souvent avec deux (ou plus) logiques d'interfaces différentes qui s'enchevêtrent, rendant l'interface graphique confuse.
- WordPress est un framework puissant, mais également très lourd. Ce qui peut parfois se ressentir lors des requêtes serveur. Nous avons déjà eu l'occasion de recevoir des erreurs 503 chez des hébergeurs bas de gamme en raison de la lourdeur du CMS. Le site étant alors inaccessible pour les internautes le temps du redémarrage du serveur.

En fin de compte, l'usage de WordPress me donne l'impression de remplacer une difficulté par une autre, ce qui est en soi une démarche logique. Avoir le choix de ses difficultés est une manière de s'assurer une progression certaine.

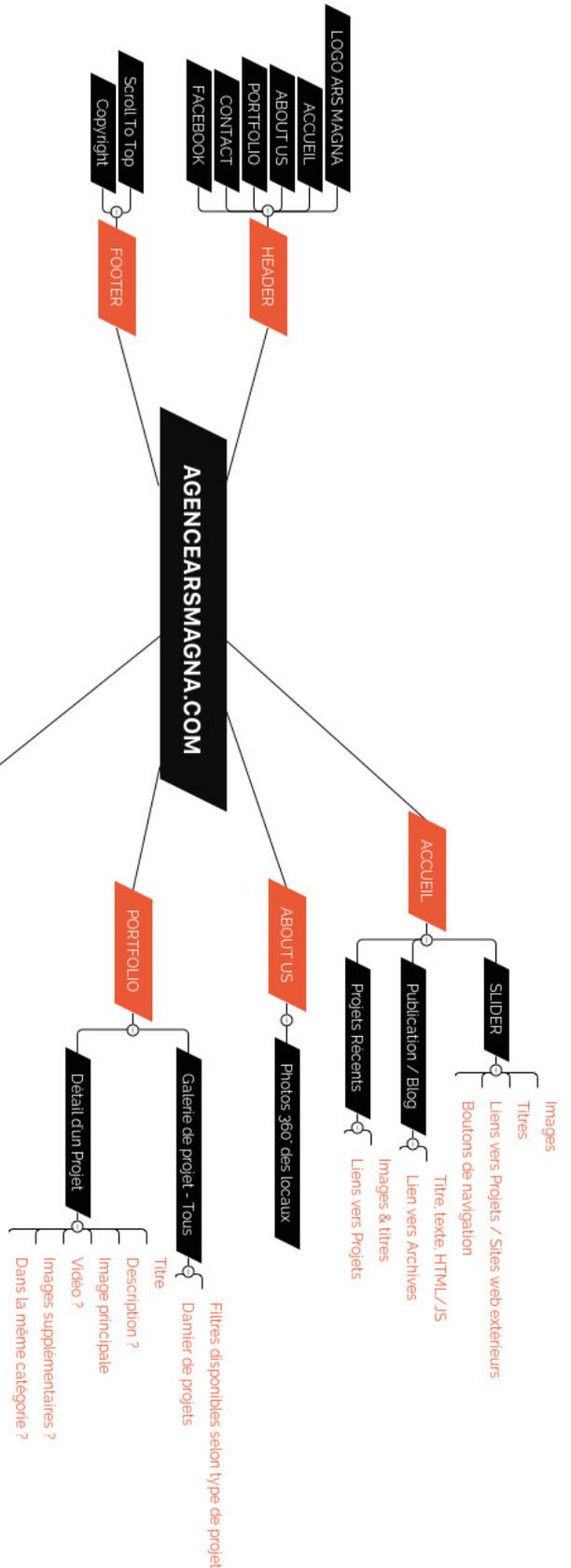
Les connaissances de mes collègues concernant les langages du web étant limitées par manque de temps à y accorder, il m'a semblé naturel de pouvoir leur faire profiter de mon temps d'apprentissage.

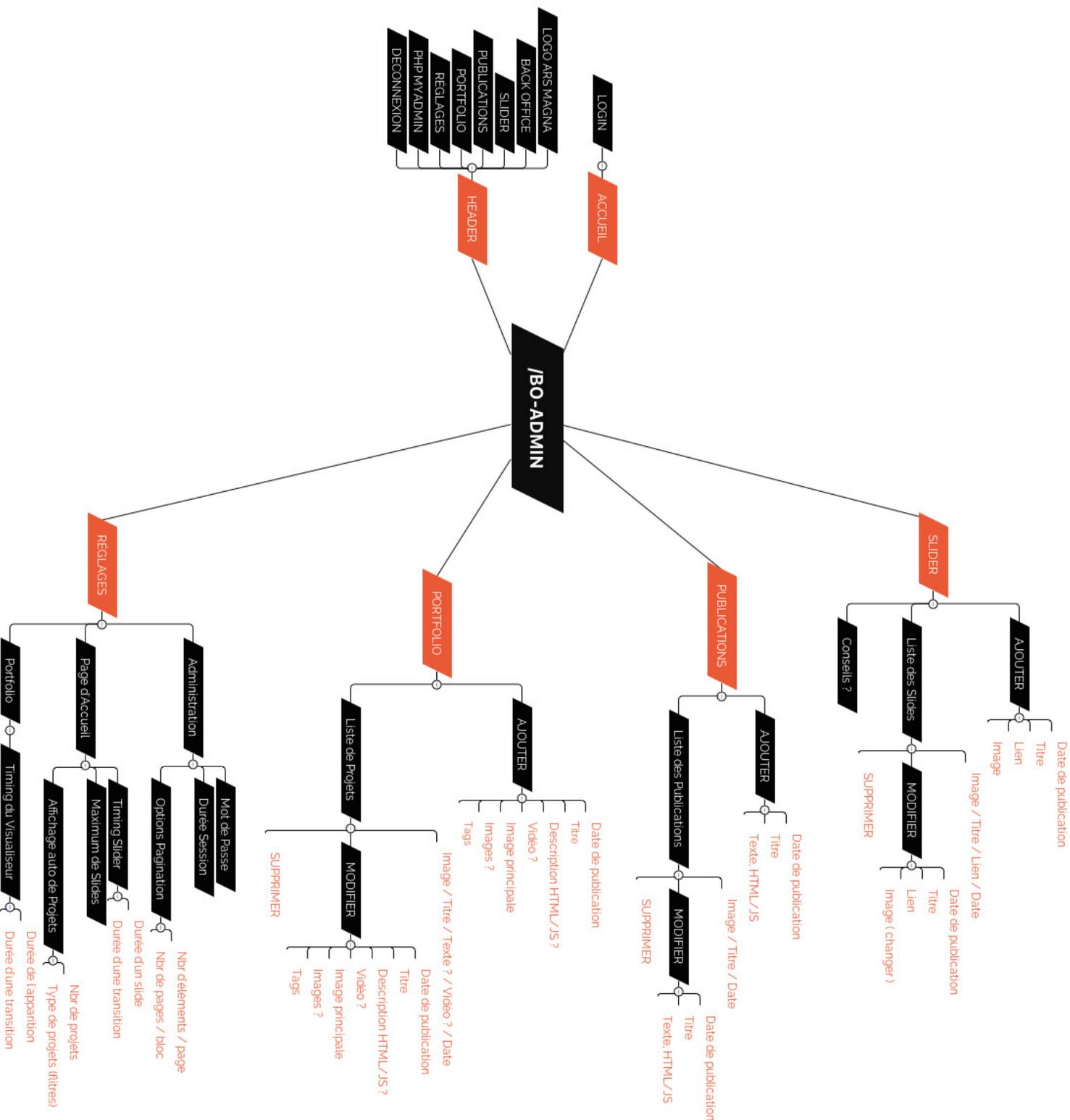
REPENSER LE SITE

Dans l'optique de concevoir un nouveau site, il sera important de conserver les avantages qu'apporte WordPress tout en gommant les défauts relatifs à l'usage de l'agence. D'où l'objectif de fournir un back-office personnalisé, à l'interface claire et minimale pour assurer sa compréhension et son efficacité.

Dès que le contenu d'un site est posé, on conçoit son arborescence en dressant un chemin de fer à l'aide d'une mind map. Voici respectivement celles du site public et du back-office :

Si vous consultez ce mémoire en version PDF, vous pouvez cliquer sur les images pour les visionner sans risquer un torticolis !





Les clients habitués de l'agence utilisent peu le site internet donc ce dernier sert principalement à capter de nouveaux clients. En tant que vitrine numérique, la mise en valeur des projets de l'agence est une priorité qui passe par deux points majeurs : les projets d'envergure et les projets récents de l'agence. Il faut donc penser à une page d'accueil capable de mettre en avant ces deux aspects.

La limitation concernant l'intégration d'un slider n'est plus un problème dans un site fait maison. Se pose tout de même la pertinence de son usage avant toute mise en œuvre. J'ai conservé l'idée ⁽¹⁾, car elle m'a semblé être une bonne occasion pour mettre en avant les projets phares d'Ars Magna. Une gestion manuelle des slides depuis le back-office a donc été prévue.

Les projets récents ⁽²⁾ ont été pensés comme un affichage automatique personnalisable.

Je me suis interrogé sur la présence d'un petit encadré texte situé entre le slider et les projets récents. Un encadré fonctionnant comme l'affichage de la dernière entrée d'un blog ⁽³⁾. L'idée étant de permettre la rédaction d'un petit billet d'humeur servant à humaniser l'agence, lui donner de la personnalité, mais aussi de souligner l'activité de l'agence. Un bouton "Archive" pourrait permettre la lecture des précédents billets.

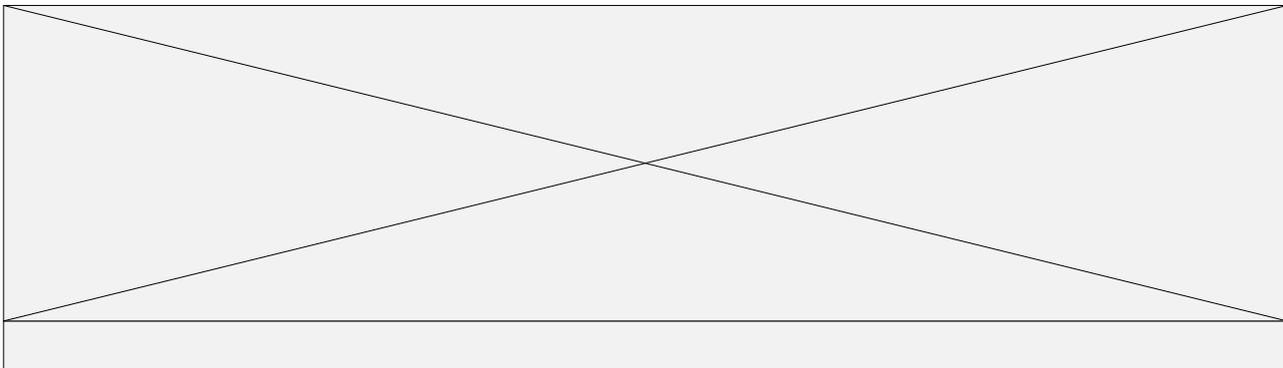
Le tout encadré par un header ⁽⁴⁾ simplifié et clair avec quelques fonctions facilitant l'ergonomie et un footer ⁽⁵⁾ incluant un scrollToTop permettant de remonter en haut de page.

Sur la page de droite, vous pouvez observer un wireframe. C'est une maquette fonctionnelle avec laquelle on décide du placement des principaux composants d'une interface.

(4)



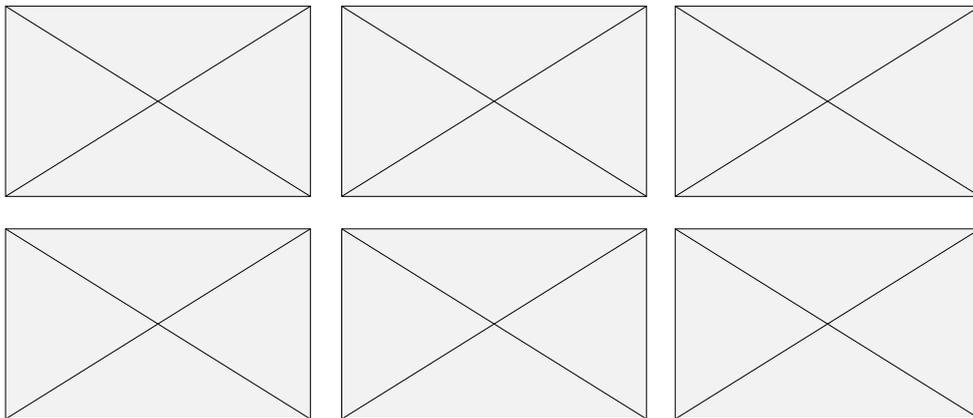
(1)



(3)



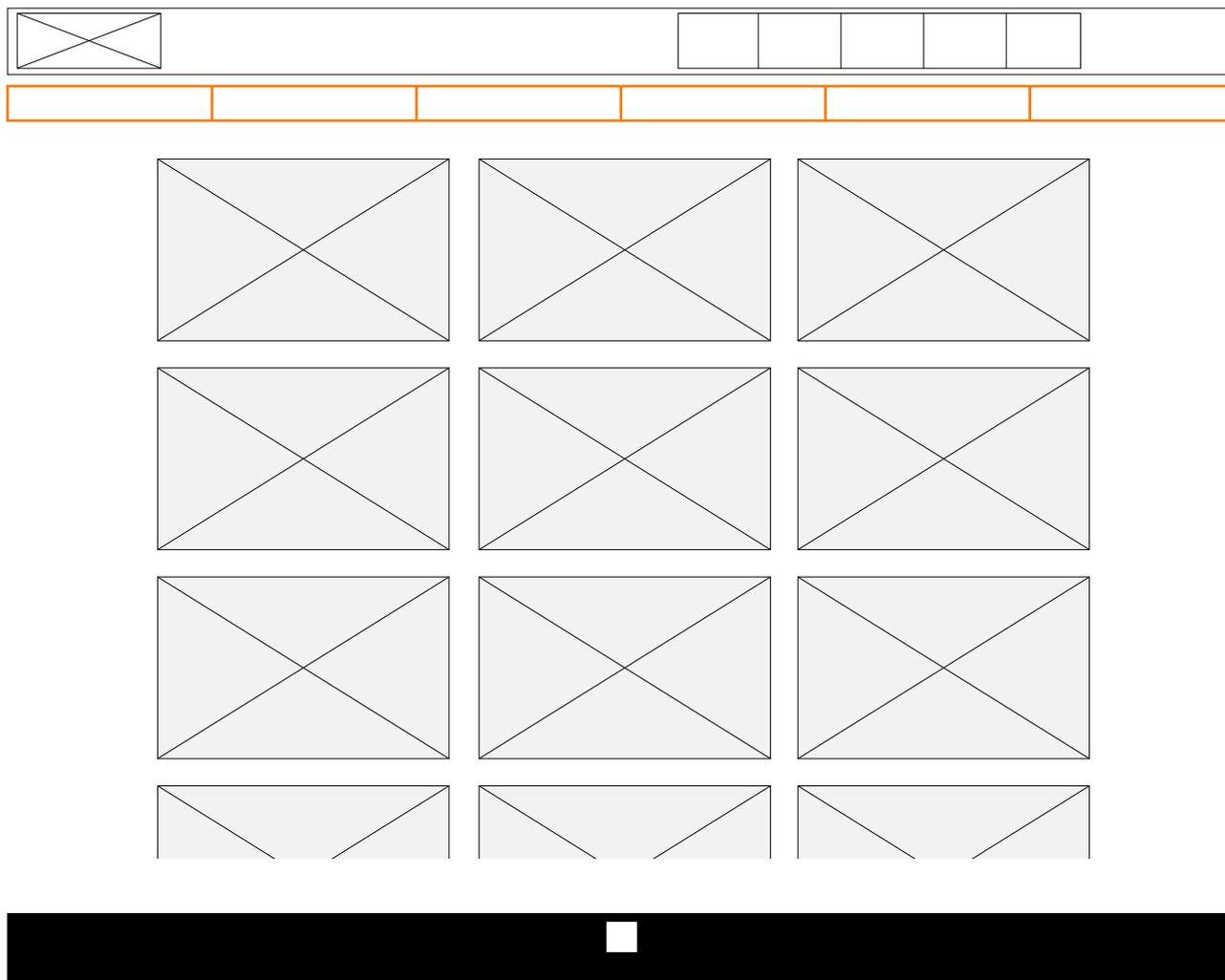
(2)



(5)

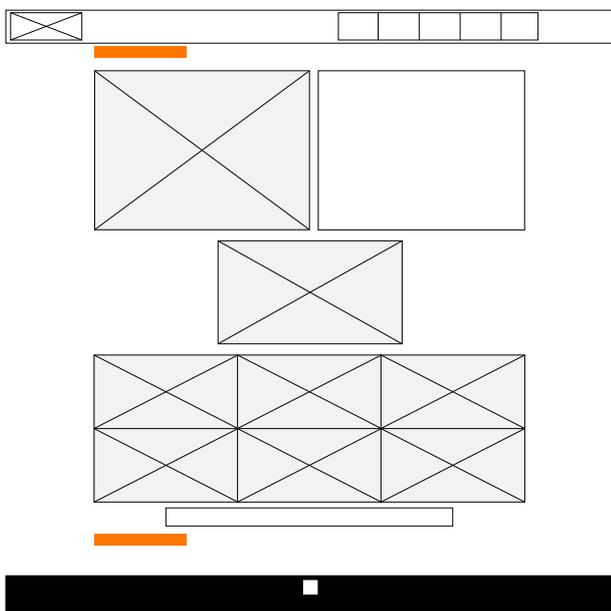


Du côté du portfolio, un affichage en damier présentant les projets par ordre antéchronologique. Des options de filtrage par type de projets permettraient autant de faciliter la navigation dans les projets en fonction de l'intérêt des clients que de montrer un éventail des domaines d'activité de l'agence.



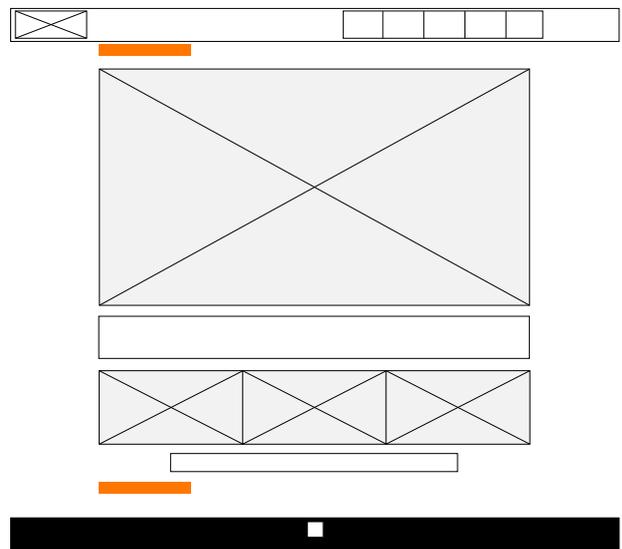
D'une manière générale, pouvoir enrichir en contenu les pages des projets m'a semblé être un point à développer. L'ajout d'une date de publication, d'un titre, d'un champ de texte explicatif, de plusieurs images d'illustration avec un navigateur et optionnellement d'une vidéo.

Les pages spécifiques aux projets peuvent avoir besoin d'une mise en page personnalisée en fonction du type de production. Par exemple, une vidéo peut être associée à un projet à majorité Print. Dans le cas d'Ars Magna, la sortie du nouvel album d'un chanteur dont la pochette et le livret ont été conçus à l'agence, mais également la vidéo lyrics associée à une chanson de l'album. Or la vidéo peut aussi être un projet autonome dans le cas des vidéos corporate de l'agence. Les sites internet conçus par l'agence peuvent avoir besoin d'une mise en page incluant un lien cliquable.



Exemple de mise en page générale avec le maximum de contenu disponible.

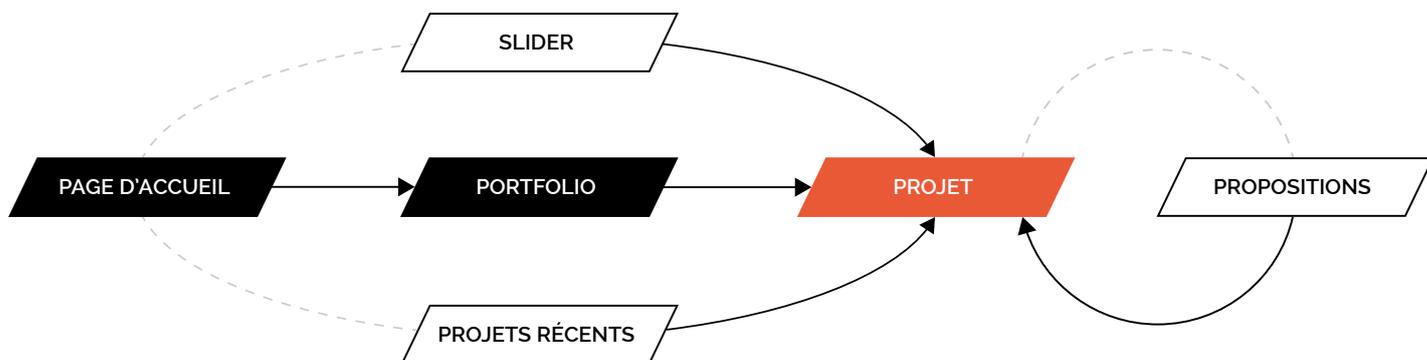
Mise en page pour les projets en motion design avec vidéo mise en avant.



Ce qui m'a semblé être important a été de fluidifier la navigation entre projets du portfolio sur la totalité du site. Faciliter la navigation de l'internaute pour lui apporter rapidement ses réponses, mais également l'inviter à papillonner sur d'autres projets, aptes à ouvrir son champ des possibles.

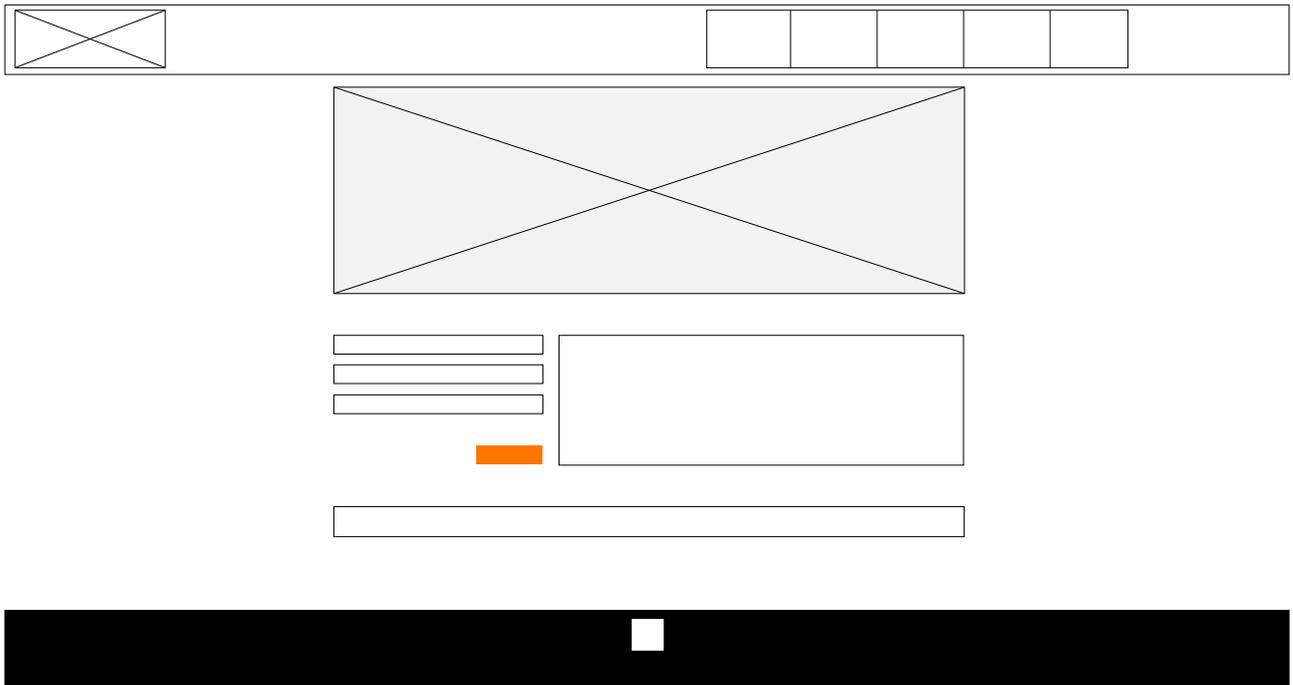
"Ah ils savent aussi faire ça ? Ça pourrait être utile, je n'y avais pas pensé."

Les liens en page d'accueil invitent évidemment à cela, mais l'inclusion d'un module de proposition ou découverte dans les pages de chaque projet permettrait de naviguer entre les projets sans retour sur la page générale du portfolio. Une économie de clic bienvenue.



Il existe un principe dans l'ergonomie du web qui veut que pour s'assurer du confort de navigation de l'internaute, chaque information soit accessible en 3 clics maximum depuis la page principale. Sur le schéma ci-dessus représentant le parcours vers un projet de l'agence, une flèche correspond à un clic. Ainsi, slider et projets récents offrent un accès direct à un projet.

La page de contact conserve le contenu de l'ancien site, mais se retrouve débarrassée de son apparence de page de blog. Le plan GoogleMap est ici plus facilement personnalisable que sur l'ancien site.



La page "Qui sommes-nous" est restée en stand-by. Ayant intégré l'agence depuis moins d'un an, il m'importait que l'équipe soit libre d'y exprimer son contenu indépendamment de ma personne. J'ai pensé à leur proposer de remplir cette rubrique à partir du moment où une vision d'ensemble du nouveau site leur serait possible.

Bien qu'il ne concerne pas directement l'internaute, le back-office est également un élément nécessitant réflexion en matière d'usage pour l'administrateur. En effet si je lui fais porter la responsabilité de la gestion du contenu du site, il me faut lui fournir une interface simple d'accès. J'ai besoin pour cela d'avoir une idée claire de ses attentes sur le plan de la gestion :

- Les actions principales de l'administrateur concernent l'ajout, la modification et la suppression de contenu, situé dans le slider, les publications en page d'accueil et bien sûr le portfolio.
- Le back-office doit être découpé en différentes rubriques pour chaque zone de contenu. Slider, publications, portfolio...
- Une zone de réglage générale concernant le back-office (changement du mot de passe, durée de la session) et la personnalisation de certains éléments de site public.
- Le back-office doit avoir une interface claire, légère et simple d'accès. Des boutons visibles et explicites et offrir une lisibilité optimale du contenu de chaque rubrique.

Depuis le début de ce mémoire, deux acronymes ont été répétés à plusieurs reprises et le seront encore par la suite : **UX** et **UI**.

Le premier sigle signifie User eXpérience et englobe une réflexion ergonomique extrêmement large. Trop pour être abordé de manière détaillée dans ce mémoire. On peut retenir qu'il place le ressenti de l'utilisateur au centre de l'attention au moment de la conception d'un projet. Apprendre à s'oublier pour se projeter à la place de notre usager est une part importante du travail d'un UX designer et ne s'improvise pas, nécessite de longues études et couvre toutes les étapes de production d'un projet. Bien que la démarche UX ne sera pas abordée ici, une réflexion liée à la facilité d'utilisation du site par l'internaute motivera ce projet en parallèle de la relation au code. Il s'agit d'ailleurs de l'un des intérêts liés à la création des différents wireframes présentés précédemment.

Le dernier sigle correspond au travail esthétique de l'interface, User Interface. Il est forcément encadré par un UX designer parce que l'esthétique d'un site ne sert pas simplement à le rendre joli, mais peut aussi aider à faciliter son usage, ou au contraire risquer de le rendre confus ! Cette étape ne sert donc pas à ajouter des fioritures, mais plus clairement à concrétiser le travail réalisé en amont dans la réflexion UX.

Avec le chemin de fer et les wireframes, une ébauche du site web se met en place et l'on peut entamer la phase suivante.

LA MAQUETTE

La maquette concerne uniquement le site public et du point de vue esthétique, il faut respecter l'actuelle charte graphique d'Ars Magna.

La police du logo est une typo bâton sur mesure. L'approche des caractères pose quelques soucis que l'on peut compenser dans des logiciels de traitement graphique. Mais depuis un navigateur, il peut être difficile d'anticiper les résultats :

Monsieur Jack, vous dactylographiez
bien mieux que votre ami Wolf.

De plus, étant assez marquée, cette typo peut risquer de devenir redondante sur des titres présents sur les pages web. Il faut lui laisser son caractère en limitant son usage.

D'où la raison de l'emploi d'une seconde typo, Raleway, également bâton et géométrique :

Voyez ce koala fou qui mange des journaux
et des photos dans un bungalow.

Les deux textes d'exemple précédents sont des pangrammes. Leur but est de tester toutes les lettres de l'alphabet en situation de rédaction. Respectivement Ars Magna et Raleway.

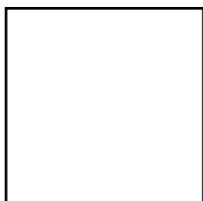
On retiendra les trois couleurs suivantes : le blanc est majoritaire, suivi d'un gris anthracite en contraste ponctué d'un orange en tonique. Il est tout à fait possible de modifier le rapport de force entre ces couleurs. Leur usage reste donc très flexible.



Couleur : Orange
RVB : 235, 90, 54
Hexadécimal : #eb5a36



Couleur : Gris anthracite
RVB : 35, 31, 32
Hexadécimal : #231f20



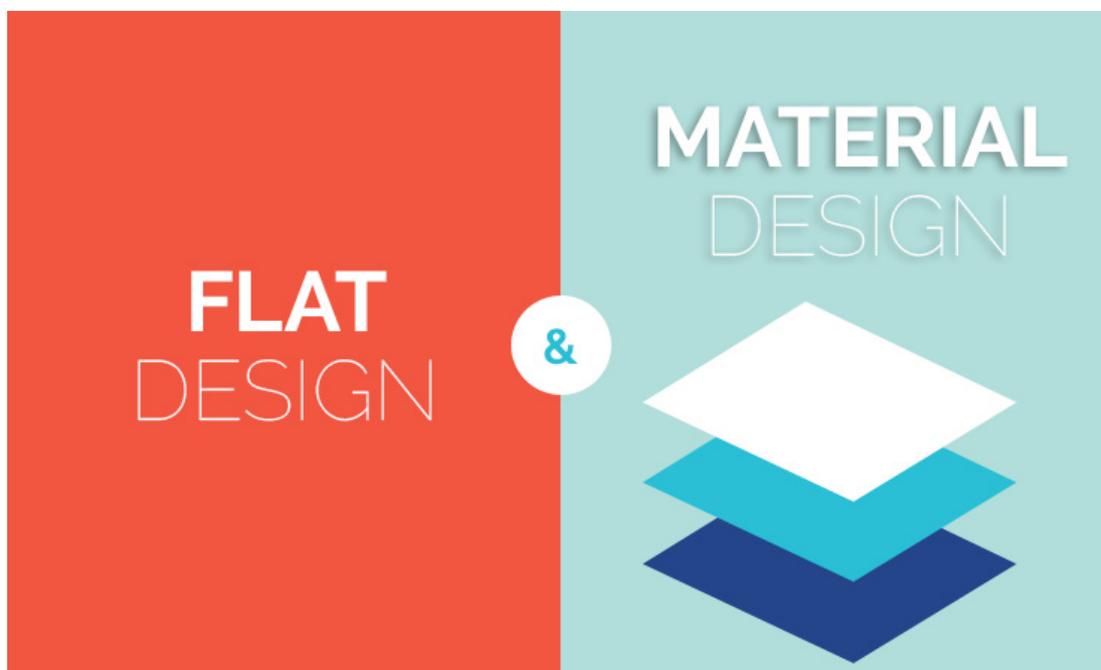
Couleur : Blanc
RVB : 255, 255, 255
Hexadécimal : #ffffff

Je n'ai volontairement pas précisé leur code couleur CMJN, car la quadrichromie correspond à l'impression papier qui n'est pas le sujet.

De manière générale, le style graphique fonctionne par des aplats de style Flat Design ⁽¹⁾ pour garder une bonne souplesse d'adaptation. Dans le respect de la charte, j'ai opté pour le blanc en couleur de fond principale, le noir pour le texte et l'orange en tonique pour texte et graphique.

Si l'orange est une dynamique qui fait ressortir certains éléments, je m'autorise également à écrire en blanc dans de petits encadrés sur fond noir ou orange pour apporter des alternatives, tant que cela ne brouille pas la cohérence générale. Je me suis questionné sur la possibilité d'employer parfois une teinte de gris. J'ai testé dans le header de la maquette et cette idée m'a amené à travailler certaines zones blanches et noires en transparence.

Ainsi, à partir d'un style graphique géométrique avec des aplats sur trois couleurs, on arrive à obtenir une palette de possibilités suffisamment large pour envisager visuellement un site web. Autant conclure que j'ai procédé à un glissement du Flat vers le Material Design ⁽²⁾ afin d'enrichir légèrement les possibilités graphiques. Un sentiment qui se renforcera par la suite avec l'ajout de légères ombres.



(1) Ou design plat. C'est un style graphique sobre et minimaliste usant de formes simples et d'aplats de couleurs. Sa capacité à répondre au besoin du responsive web design en ont rapidement fait un style épuré très prisé pour les interfaces de logiciel. Le succès du Flat Design a amené une évolution du concept pour un autre type de design d'interface.

(2) S'il est surtout un ensemble de règles de design proposées par Google s'appliquant à l'interface graphique des logiciels et applications, le Material Design propose en respect avec ses choix d'ergonomie une amélioration sensible du Flat Design avec par exemple l'usage des ombres portées.

Pour la conception de la maquette, Adobe Muse a été employé en lieu et place de Balsamiq. 😞
Pour des raisons de disponibilité, car l'agence m'a fourni un accès à la suite complète d'Adobe Creative Cloud. Par curiosité envers ce logiciel, que je n'imagine remplacer ni la conception de site web ni la conception de maquette. Il reste potentiellement utile pour mettre en place une maquette interactive fonctionnelle et tester la navigation de manière plus réaliste.

Adobe a depuis annoncé l'arrêt du développement de Muse. Ce logiciel n'a pas été la première tentative de la firme pour fournir une solution de conception WYSIWYG pour le web. Cet acronyme barbare signifie "What You See Is What You Get" qu'on peut traduire littéralement par "Ce que vous voyez est ce que vous obtenez".

La logique WYSIWYG est centrale dans le travail d'un graphiste bien qu'à une époque, on travaillait également en ligne de code pour concevoir des graphiques. Cette approche a évidemment révolutionné le travail de la PAO (Publication assistée par ordinateur).

Dans le cas du web, cette approche reste problématique, car si un logiciel comme Muse promet aux utilisateurs un rendu proche du résultat final, il fait malheureusement abstraction du casse-tête que représente la fidélité d'affichage sur différents navigateurs et de l'optimisation du code. D'autant plus que ce code généré par un programme n'est que très peu lisible par un développeur et donc difficile à personnaliser en dehors de Muse avec un logiciel de programmation plus traditionnel comme SublimeText ou Atom.

Il y a un peu plus de quinze ans, j'ai déjà eu l'occasion d'utiliser l'ancienne solution d'Adobe dont Muse est la continuité : Adobe GoLive. Au départ, l'interface WYSIWYG a été rassurante pour moi et m'a facilité la création de sites web alors que je n'avais que 15 ans.

Avec le recul, je me rends compte que cette apparente facilité a été un piège, car il ne m'a pas incité à m'intéresser directement au HTML. Et quand le CSS s'est démocratisé, je me suis senti techniquement désorienté et dépassé d'autant plus que GoLive a progressivement été abandonné. J'ai arrêté de m'intéresser au web par la même occasion.

C'est dans mon parcours personnel un autre exemple où la solution de facilité empêche d'approfondir ses connaissances. Une expérience, source de ma motivation pour le projet actuel.



Adobe Muse



balsamiq Wireframes

À ce stade de la maquette, seul le header, le slider et les filtres du portfolio sont fonctionnels pour la navigation.



RS 2M - Le nouveau catalogue de formations des dispositifs médicaux



Encore un soir le nouvel album de Céline Dion



Camille et Julie Berthouze, leur nouvel album en leur comédie



Auparavant vertical et sur le bord droit, la largeur du header dépendait du contenu texte, il est désormais horizontal. Je l'ai également fixé à l'écran de telle manière qu'on ne le perd plus avec le défilement de la page. Une très légère transparence permet d'alléger le bandeau tout en appuyant sa présence au premier plan.

J'ai tenu à alléger le découpage des rubriques du site pour ne plus se perdre dans des sous-rubriques et réduire le nombre de clics. J'ai ainsi la place de rajouter une icône vers la page Facebook.

Un petit changement de couleur accompagne la catégorie de la page en cours.

Le footer est resté très simple avec la présence d'un scroll-to-top permettant de remonter en haut de page.

J'ai proposé un grand slider sur la page d'accueil dont l'équipe avait envie sur son site WordPress, avec texte et barre de navigation. Les projets récents sont disposés en dessous.

La page de contact est conforme au wireframe et souligne un défaut : la place du numéro de téléphone, finalement trop discret et en retrait. Une solution sera apportée lors du développement front-end.



ACCUEIL OÙ SOMMES-NOUS PORTFOLIO CONTACT



ÉCRIVEZ-NOUS

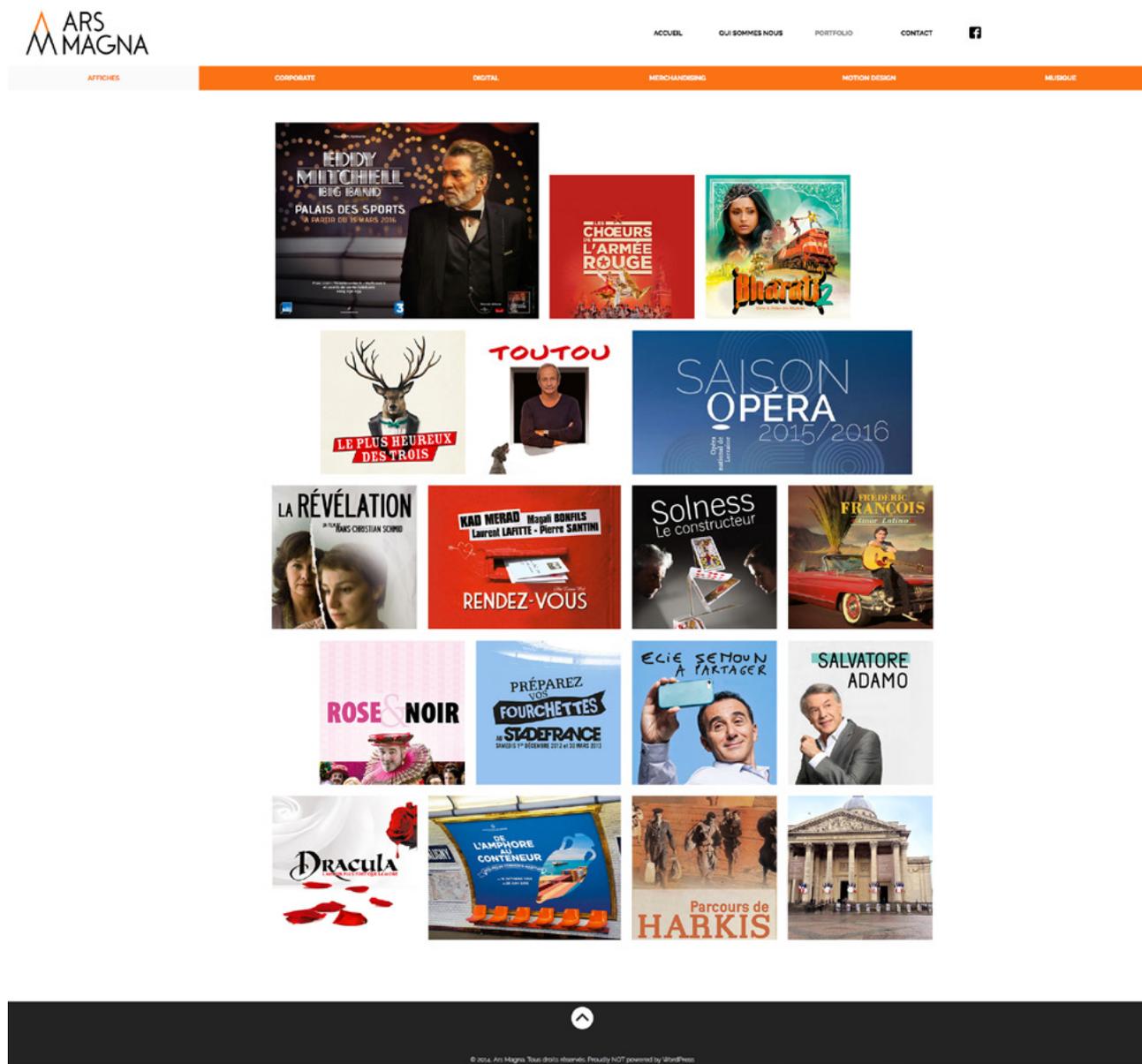
Form for contact: Name, Email, Subject, and a large text area for the message. Includes a 'Envoyer' button.

OU APPELEZ-NOUS AU 01 53 75 10 10

Écrivez-nous via le formulaire de contact ou appelez-nous au 01 53 75 10 10. Nous reviendrons vers vous dès que possible.



Sur le portfolio, un hover sur les vignettes de projets sera rajouté au moment du développement front-end car Muse ne permet pas les hover élaborés.



J'ai profité de l'usage de Muse pour expérimenter une disposition des vignettes moins tristes. Je ne suis pas certain d'avoir les compétences en code permettant de mettre en place une telle structure. Le mélange de formats de vignettes se doit d'être configuré par l'administrateur du site pour éviter des cadrages aléatoires ne mettant pas toujours en valeur toutes les images.

Est-ce une bonne idée de donner cette responsabilité à l'administrateur du site qui voudra quelque chose de simple ? L'équilibre entre choix esthétique, simplicité et connaissances techniques ne me semble pas stable, je me suis donc reporté sur une autre structure au moment du développement front et back-end. Moins régulière et froide, mais automatisée.

Les visuels des projets m'ont été fournis par l'agence. Ils ont été conçus dans d'autres perspectives que celle d'une représentation en damier, or tous les visuels ne s'y prêtent pas. Ça n'est pas un problème très grave, mais il faudra le prendre en compte pour le choix des visuels des prochains projets dans le cadre d'un usage réaliste.

La création de cette maquette m'a permis de me confronter à une première mise en image concrète et d'entamer une réflexion sur la manière d'engager le développement front-end. Mes compétences dans ce domaine étaient encore balbutiantes à cette période, j'ai donc produit un certain nombre d'erreurs qui m'ont été enrichissantes.

Les étapes de développement n'ont pas forcément été organisées de la manière la plus efficace, car je me suis laissé porté par ma motivation à apprendre et à découvrir. Ce qui m'a valu plusieurs phases de réécritures du code.

INTRODUCTION AU DÉVELOPPEMENT

Front-end et back-end, quelles différences ? Pour expliquer cela de manière simplifiée, un site web est **comme un spectacle d'opéra ou une pièce de théâtre** : il se passe des choses différentes sur scène ou en coulisse...

La scène représente le spectacle offert aux yeux du public tandis que dans les coulisses se préparent les acteurs et accessoiristes pour la suite du spectacle.

La scène, cela serait la page web affichée sur le navigateur de l'internaute, ce qu'on appelle le côté client. L'écriture du code implique l'usage des langages HTML5, CSS3 et JavaScript.



Le **HTML** est un langage à la base d'un site internet. On lui réserve le contenu brut d'un site comme le texte, les images, sans considérer la mise en page et l'aspect esthétique.

Ce travail revient au **CSS** qui permet de définir le style des éléments. La définition des styles peut aller loin, jusqu'à considérer des animations en boucle ou avec interaction de l'utilisateur.

Mais pour une interaction ou des actions plus poussées, le langage **JavaScript** est pour le coup un vrai langage de programmation permettant d'insuffler une gestion dynamique du contenu d'une page web.

Les **couisses**, c'est évidemment la partie back-end, côté serveur, où l'on prépare le contenu de la page à générer suivant la demande de l'internaute. Les langages employés peuvent varier, mais j'ai décidé d'explorer les plus populaires d'entre eux :



Il nécessite l'usage d'un langage de preprocessing permettant de générer des pages HTML à la demande, le plus connu étant celui que j'ai décidé d'utiliser : **PHP**.

Il peut requérir l'usage de données persistantes. Derrière ces mots barbares se cache une base de données permettant de stocker toutes sortes de données. Du texte (titre et article d'un blog), une date (de parution, de modification) ou même des nombres.

Si stocker un article de blog dans une base de données permet de la récupérer quand un utilisateur en demande l'accès, gérer une base de données nécessite l'apprentissage d'un langage de gestion de base de données. Il en existe un grand nombre, j'ai choisi **MySQL** pour sa popularité et son ouverture qui m'a semblé être une bonne première approche.

Si PHP est un langage exécuté côté serveur, il n'est pas le langage du serveur. Il en existe plusieurs et c'est **Apache** que j'ai choisi pour sa popularité et son ouverture. La documentation est également très facile à trouver et techniquement accessible.

Il existe également des frameworks reposant sur ces langages ayant pour but de simplifier le travail des développeurs. Avec un framework, quelques lignes de codes reviennent à remplacer toute une portion de code à écrire soi-même en répondant aux contraintes de compatibilité, sécurité et optimisation. Ils peuvent nécessiter un apprentissage plus ou moins long. Par exemple, jQuery est un framework JavaScript apportant un gain de temps précieux, car il permet en peu de lignes de code de réaliser des tâches qui seraient complexes en JavaScript. Il s'assure également de la compatibilité de ces différentes tâches en fonction des navigateurs. jQuery est un framework souvent employé par les webdesigners pour le gain de temps et de compatibilité évident. Il peut également être décrié, car oblige à charger la totalité de sa bibliothèque de fonctions quand bien même on n'en utiliserait qu'une seule.

Dans mon cas, j'ai appris à l'utiliser dans la foulée de JavaScript et m'en suis servi à deux reprises dans le projet actuel. Mais avec le recul, mon souhait est aujourd'hui de me passer le plus souvent de jQuery.

En effet, j'ai utilisé deux fonctions de jQuery que j'évoquerais plus loin dans le projet et cela m'oblige à faire charger la totalité de la bibliothèque de fonctions jQuery à l'utilisateur. C'est une lourdeur inutile que j'imposerais à l'utilisateur et cela ne me paraît pas pertinent dans le but d'offrir un site léger.

De plus, si jQuery repose sur JavaScript, cela veut dire que tout ce qui est faisable dans un framework l'est également dans son langage d'origine. Et dans ma situation d'apprentissage, utiliser jQuery, c'est risquer de m'empêcher d'apprendre à utiliser JavaScript en profondeur. J'ai donc pour objectif de progressivement retirer jQuery du projet.

Pour la partie back-end, je me suis posé la question d'un framework PHP tel Symfony. Le temps d'apprentissage s'avérait très long avant de pouvoir entreprendre la moindre application concrète. J'ai donc reporté ce contact à plus tard et décidé de me confronter à un développement from-scratch. En pensant que cela me sensibiliserait sans doute plus aux avantages d'un framework back-end.

Parce qu'il n'est pas concevable de réaliser un site web sans le tester, il me faut mettre en place un environnement en local, donc depuis un ordinateur capable d'être client et serveur en même temps. Il existe de multiples solutions, par exemple d'installer un serveur Apache et une base de données MySQL. Ces tâches sont grandement simplifiées sur Mac, car Apache est installé par défaut.

Mais j'ai décidé de me tourner vers MAMP, une solution logicielle prenant en charge la simulation d'un serveur. L'intérêt de la méthode concerne surtout la portabilité du projet en développement depuis différents ordinateurs. Nul besoin alors de réinstaller et configurer Apache ou MySQL sur chaque ordinateur, car MAMP s'en charge.

J'ai parfois utilisé son équivalent Windows : WAMP, pour faciliter les tests depuis différents navigateurs et système d'exploitation.



Pour parler plus en profondeur du développeur, je ne peux me résoudre à créer deux chapitres distincts séparant front et back-end. Ils seront donc évoqués de concert dans deux parties distinctes : site public et back-office. Certains éléments comme le slider ou les projets font appel à une base de données qui naturellement fait référence au back-end.

Et je considérerais une troisième partie plus orientée back-end soulignant des étapes invisibles pour l'utilisateur, mais qui touchent à des notions d'organisation, d'optimisation et de sécurité.

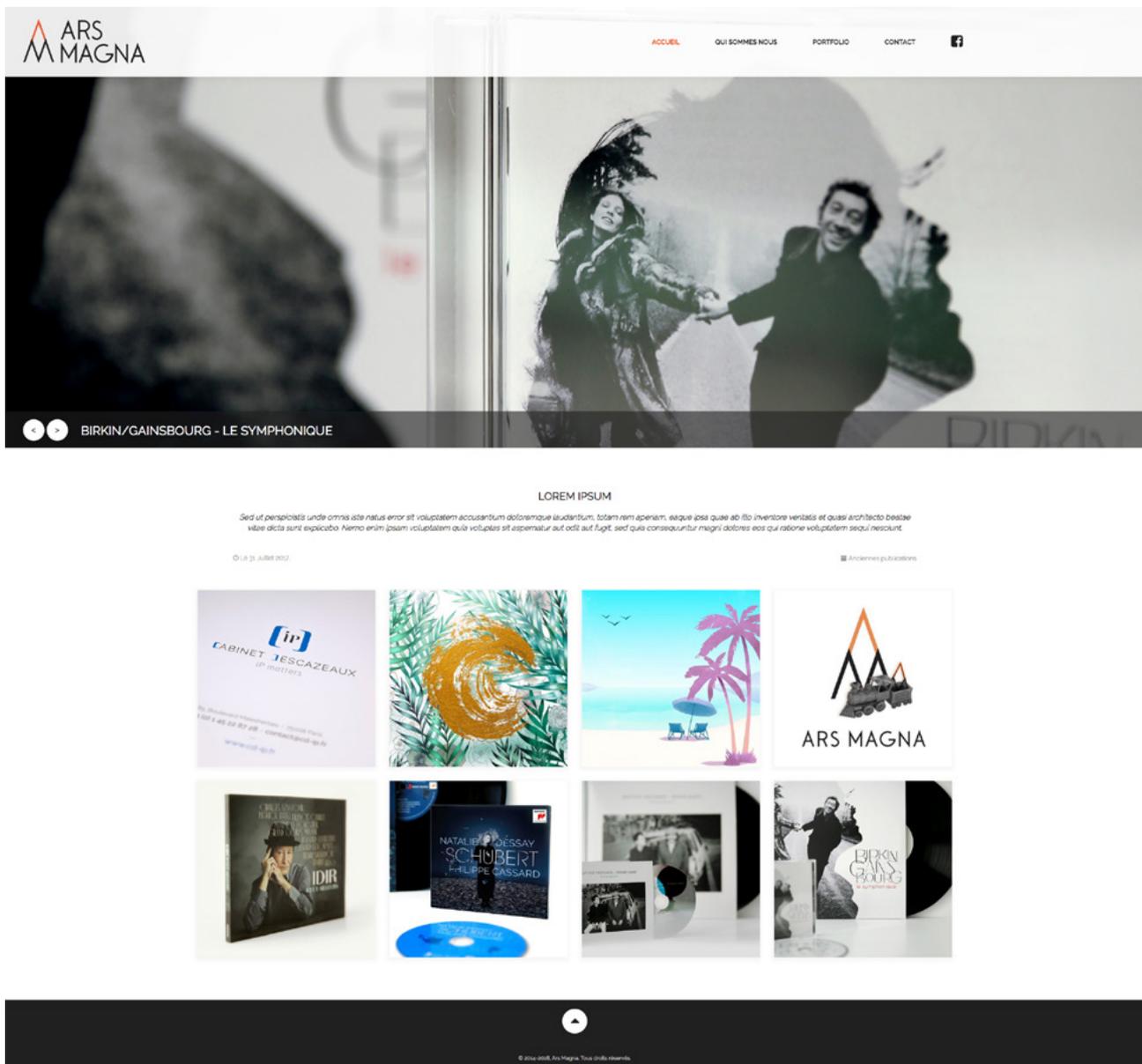
Je ne pourrais pas non plus faire un rapport exhaustif sur tout le cheminement qui a été le mien. Parce qu'il sera plus pertinent de mettre en lumière certains points méritant d'être développés, mais aussi parce que malgré la préparation dans l'organisation de mon travail, beaucoup de découvertes ont ponctué mon avancement de nouvelles idées qui ont rallongé et parfois redéfini mon organisation générale.

RÉALISATION DU SITE PUBLIC

1) HEADER & FOOTER

Dans un premier temps, j'ai commencé à concevoir la page d'accueil avec header et footer. En se référant à la maquette, le header est conçu assez simplement, mais j'ai décidé de lui apporter des améliorations d'usage pour enrichir sa simplicité.

D'abord en augmentant le padding de chaque rubrique pour améliorer la zone cliquable, puis en colorant la rubrique du menu correspondant à la page. Le logo Ars Magna du header est également un raccourci vers la page d'accueil. C'est un doublon discret qui me semble utile et intuitif. La coloration de la rubrique concernant la page actuelle a dans un premier temps été initiée en jQuery. J'ai finalement réécrit cette partie en optant pour l'application d'une classe "currentPage" via la construction de la page par PHP. La construction du header étant incluse dans une fonction utilisant un paramètre pour définir la rubrique à laquelle appliquer la classe.



Le footer est classique et dispose d'un ScrollToTop, très pratique pour remonter en haut de page bien que inaccessible si l'utilisateur est en milieu de page. J'ai donc modifié le comportement des éléments du header pour que le bouton de la rubrique de la page actuelle ne servent plus de lien, mais de second ScrollToTop. Cet effet se répercute également sur le logo Ars Magna, lien doublon vers la page d'accueil.

Au départ inclus en jQuery, le ScrollToTop a été remplacé par une version JavaScript que j'avais réalisé pour un autre projet. Beaucoup de lignes de code viennent remplacer cette fonction, car j'ai été confronté en pratique aux diverses façons qu'ont les navigateurs d'interpréter JavaScript. Il m'a fallu employer des variables vérifiant le navigateur utilisé par l'internaute pour adapter le code du ScrollToTop.

D'ordinaire, je n'ai pas les connaissances me permettant d'écrire ses variables, mais internet et principalement stackoverflow.com sont des ressources particulièrement utiles. Il reste important de bien vérifier les dates de parution des solutions disponibles sur le web pour ne pas employer une technique obsolète.

L'animation du ScrollToTop n'est pas tout à fait identique à celui existant en jQuery, mais suffisante et personnalisable.

Sur le footer, j'ai pu inclure un petit élément de PHP facilitant la mise à jour de l'année dans le copyright. Il s'agit de ce genre de petit détail qui fait toute la magie dans l'utilisation du code. La capacité à automatiser la production de contenu n'est pas forcément une nouveauté pour un graphiste, mais c'est tout de même une découverte qui m'incite à réfléchir aux usages potentiels à injecter un peu partout sur mes pages.

J'y reviendrais plus en détail dans la partie back-end !

2) PAGE D'ACCUEIL

A) Slider

Compte tenu de mon niveau de connaissance et de toutes les découvertes que j'ai déjà dû assimiler jusque là, le slider m'a paru être une tâche lourde à réaliser. J'ai donc dans un premier temps cherché le code d'un autre développeur à intégrer dans ma page.

J'ai découvert "[ResponsiveSlides.js](#)" qui semblait le mieux répondre à mes attentes.

Je suis parvenu à adapter le CSS du slider pour le faire correspondre à ma maquette malgré un manque de documentation. J'ai constaté l'usage de jQuery dans ce code ainsi qu'un cycle d'animation des slides très instable, souvent pollué par les actions de l'utilisateur, ce qui est un comble. Impossible également de changer la méthode de transition qui est un simple fondu enchaîné.

Tous ces petits détails plutôt agaçants m'ont amené à réécrire mon propre slider responsive en JavaScript suite à un exercice en cours qui m'en a offert la capacité. Pouvoir reposer entièrement sur son propre code est une sensation agréable d'indépendance. Le revers de la médaille est qu'il m'incombe d'organiser mon code pour en faciliter son entretien. À ce stade du développement, je n'en avais qu'une très légère idée qui se renforcera par la suite. Je me suis inspiré de "[ResponsiveSlides.js](#)" dans l'idée du lancement d'une fonction avec des paramètres pouvant modifier le comportement du slider avec transition par glissement. On peut désormais définir la durée de lecture des slides ainsi que leur transition depuis le back-office.

Le code est donc facilement portable et j'ai déjà eu l'occasion de l'utiliser sur un projet concret pour un client d'Ars Magna. Cet emploi de mon code dans une vraie mission m'a convaincu de l'intérêt de continuer à approfondir ce projet personnel.

Les codes HTML et JavaScript de ce slider sont conçus pour autoriser des injections en PHP de données depuis une base. On peut ainsi récupérer les données de chaque slide comprenant l'image, le texte et l'URL, mais aussi des éléments de configuration du slider comme le temps de lecture d'un slide ou la durée de chaque transition.

Pour en revenir aux URLs, rappelons qu'il s'agit d'une adresse web et qu'elle peut avoir au moins deux formes : relative ou absolue. Cela va nous intéresser pour modifier le comportement du slider...

- Une URL relative n'est pas une URL complète. Elle déduit son adresse complète en fonction de l'URL où l'on l'active.

Par exemple : `"/portfolio"` depuis l'adresse `"http://www.agencearsmagna.com"` conduira vers l'adresse : `"http://www.agencearsmagna.com/portfolio"`

- Tandis qu'une URL absolue est une adresse complète.

Dans le cas du site, il serait intéressant de pouvoir adapter le comportement d'un lien du slider suivant l'adresse. Pour créer un nouvel onglet s'il s'agit d'un lien externe, par exemple vers un site créé par l'agence, ou naviguer dans la page actuelle quand le lien concerne un élément du portfolio.

Et c'est là qu'une distinction entre URL relative et absolue peut devenir très utile. Le code JavaScript du slider est capable d'analyser la structure de l'URL pour déterminer si celle-ci est relative ou absolue par la présence ou non d'un `"http://"` en début d'URL. Si c'est le cas, on ajoute l'attribut `"target=_blank"` à la balise `<a>` afin de créer un nouvel onglet.

L'analyse de l'adresse se fait par le biais d'une regex, qu'on appelle également une expression régulière. Il s'agit d'une chaîne de caractère suivant une syntaxe très précise décrivant un ensemble de chaînes de caractères possibles.

Dans le cas de notre slider, la regex employée est la suivante : `/https?:\\//`

Pour faciliter le travail de l'administrateur dans l'ajout de contenu depuis le back-office, je me suis servi de la fonction `preg_match()` en PHP pour déterminer si l'URL ajoutée à un slide concerne un lien depuis le site de l'agence ou vers l'extérieur :

Si l'URL contient `"http://www.agencearsmagna.com/"`, celle-ci est modifiée avant ajout en base de données pour ne contenir que la partie relative de l'adresse.

Dans l'optique de simplifier la création des images pour les slides, j'ai fourni un modèle au format Photoshop PSD pour gérer simplement les dimensions et les parties visibles des images.



Cette capacité d'avoir une vue autant sur la partie graphique que sur le code m'octroie une vision large des besoins utilisateurs. Anticiper le besoin de créer facilement des images et de faciliter l'ajout d'URLs gérées automatiquement, tout cela est possible parce qu'en dehors de l'étude des besoins utilisateurs, je me projette comme étant usager du service. Et pouvoir répondre à cette projection avec des aptitudes croisées de graphistes et de développeurs m'autorise à couvrir un large champ de connaissances techniques dont le premier revers est le temps à y accorder en termes de concrétisation.

B) Publications

Toujours en page d'accueil et juste en dessous du slider, j'ai finalement décidé d'ajouter un petit encadré texte.

Un simple champ `<section>` contenant un titre `<h2>` et son paragraphe `<p>`. Ces deux derniers champs ne sont pas obligatoires et côté back-end, j'ai laissé la possibilité d'injecter du code depuis le back-office pour personnaliser le contenu de ces publications. Je m'attarderais sur les aspects techniques et sécuritaires que cela implique dans la prochaine partie. Il est également possible d'accéder aux précédentes publications depuis ce même lien.

LOREM IPSUM

Sed ut perspiciatis unde omnis iste natus error sit voluptatem accusantium doloremque laudantium, totam rem aperiam, eaque ipsa quae ab illo inventore veritatis et quasi architecto beatae vitae dicta sunt explicabo. Nemo enim ipsam voluptatem quia voluptas sit aspernatur aut odit aut fugit, sed quia consequuntur magni dolores eos qui ratione voluptatem sequi nesciunt.

© Le 31 Juillet 2017.

«Lorem Ipsum» est un faux texte inspiré du latin dont on sert pour générer du texte et effectuer des tests de mise en page. Une petite ombre très légère sert à encadrer et séparer les différentes publications depuis la page d'archive.

C) Projets récents

Les projets récents représentent un affichage automatique des derniers projets de l'agence. Une image cliquable renvoyant vers la page du projet concerné. Leur mise en page est simple et prévue pour être responsive.

Côté back-office, des options autorisent l'administrateur à personnaliser les paramètres en sélectionnant le nombre de projets récents à afficher, entre 1 et 12, ainsi que la catégorie des projets, par défaut sur "Tous".

AFFICHAGE AUTOMATIQUE DE PROJETS :

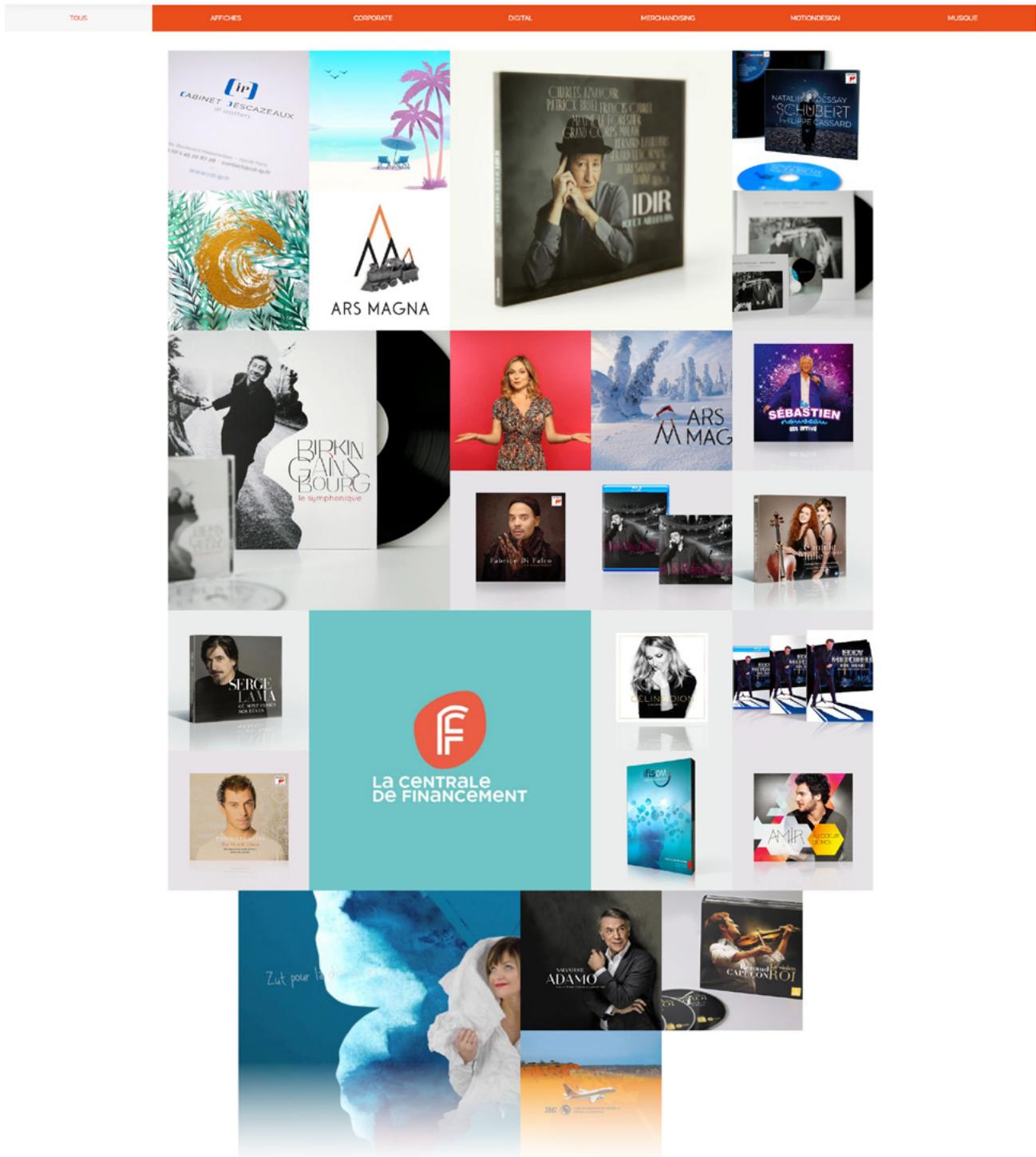
Nombre de projets : 8 

Type de projets : Tous 

3) PORTFOLIO

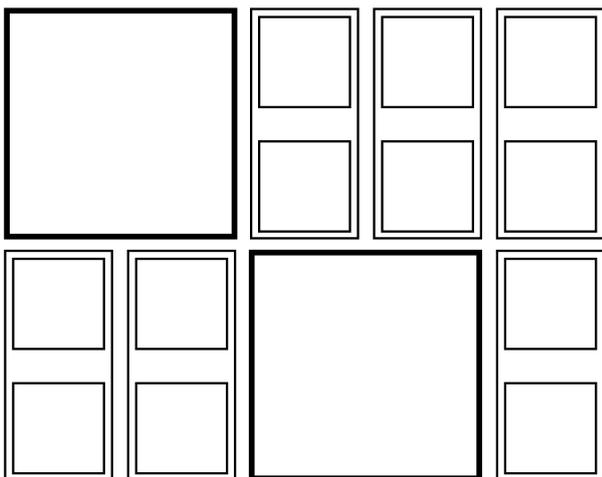
A) Page générale

La page générale du portfolio présente un grand damier avec tous les projets d'Ars Magna. Des filtres sont disponibles pour choisir une catégorie précise de projets à présenter. Un hover affiche le titre du projet et un clic nous envoie vers le projet sélectionné.

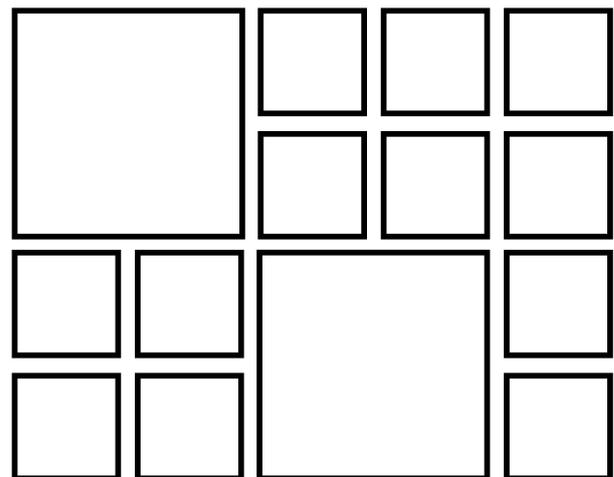


En pleine découverte des possibilités de PHP et MySQL, j'ai potentiellement commis une importante erreur de jugement quand j'ai conçu le damier de projet sur la page générale du Portfolio. Je ne voulais pas d'un damier trop classique et très vite, l'idée d'alterner de grandes et petites vignettes m'a demandé de réfléchir à une autre mise en page. J'ai alors pensé créer une structure HTML générée en PHP pour reproduire la mise en page recherchée. Je n'ai pas pensé que cette mise en page pouvait être conçue plus simplement en CSS avec Flex car j'ai eu envie d'expérimenter le PHP.

Bien que ce projet ait pour but d'apprendre et de découvrir, il aurait été bon que sa conception finale soit la plus proche possible des standards d'aujourd'hui. Or le HTML doit rester le plus neutre possible là où la mise en page doit au maximum être déléguée au CSS. D'autant plus que la structure que j'ai conçue ne m'avantage pas pour le responsive. Cette erreur souligne une limite évidente : il me faut clairement faire très attention à ne pas trop m'emballer au risque de m'engager dans des choix techniques manquant de pertinence et uniquement motivés par la découverte.



Erreur commise dans la structure HTML en regroupant deux images dans une <div> pour obtenir une colonne.



Une solution simple avec une mise en page Flex.

B) Page projet - générique

J'ai créé un modèle de mise en page générique pour tous les projets ainsi que deux autres modèles plus adaptés aux projets motion design et web. Pour l'administrateur, la sélection du modèle de mise en page est transparente et automatisée en fonction des sélections de catégories d'un projet. Un projet sera affiché en motion si la seule catégorie choisie est "Motion design" et il en va de même pour les projets digitaux.

J'ai choisi un fonctionnement automatique, mais la question d'un choix manuel laissé à la discrétion de l'administrateur est toujours en suspens.

The screenshot shows a website gallery page for the project 'AMIR AU CŒUR DE MOI'. The page features a navigation bar with the logo 'ARS MAGNA' and menu items: ACCUEIL, QUI SOMMES NOUS, PORTFOLIO, CONTACT, and a Facebook icon. A red button labeled '< RETOUR A LA GALERIE' is positioned above the main content area.

The main content area displays a large image of the album cover for 'AMIR AU CŒUR DE MOI'. To the right of the image, the text reads: « AU CŒUR DE MOI » L'ÉDITION COLLECTOR EST DISPONIBLE !. Below this, a short paragraph describes the project: Découvert lors de « The Voice » 2014 où il a atteint la finale, Amir a été choisi pour représenter la France lors du prochain concours Eurovision. Ars Magna est fier d'avoir été choisi pour concevoir et réaliser les visuels de l'album original et de cette édition collector.

Below the main image, there is a video player showing a scene with graffiti that says 'NON. NON. NON.' and 'JE N'SAIS PAS POURQUOI JE S'EFFRAIS SANS TOI'. Below the video player, there is a collage of images showing various merchandise items, including a grey beanie, a CD/DVD case, a t-shirt, and a book.

At the bottom of the page, there is a red button labeled '< RETOUR A LA GALERIE'. Below the button, there is a small text block: Dans la catégorie "Motion Design", découvrez aussi: « Où sont passés nos rêves » Le nouvel album de Serge Lama // Pendaison de crémaillère - Juin 2017 // Diane Tell - Voyage au Paradis.

The footer of the page contains a play button icon and the copyright notice: © 2016-2018, Ars Magna. Tous droits réservés.

< RETOUR À LA GALERIE



« AU CŒUR DE MOI » L'ÉDITION COLLECTOR EST DISPONIBLE !

Découvert lors de « The Voice » 2014 où il a atteint la finale, Amir a été choisi pour représenter la France lors du prochain concours Eurovision.

Ars Magna est fière d'avoir été choisie pour concevoir et réaliser les visuels de l'album original et de cette édition collector.

© Le 29 Avril 2015



☞ Dans la catégorie "Musique", découvrez aussi
Le Violon Roi - Renaud Capuçon // « Les sauvages » Fabrice Di Falco // Chimichurri, l'album à quatre mains

< RETOUR À LA GALERIE

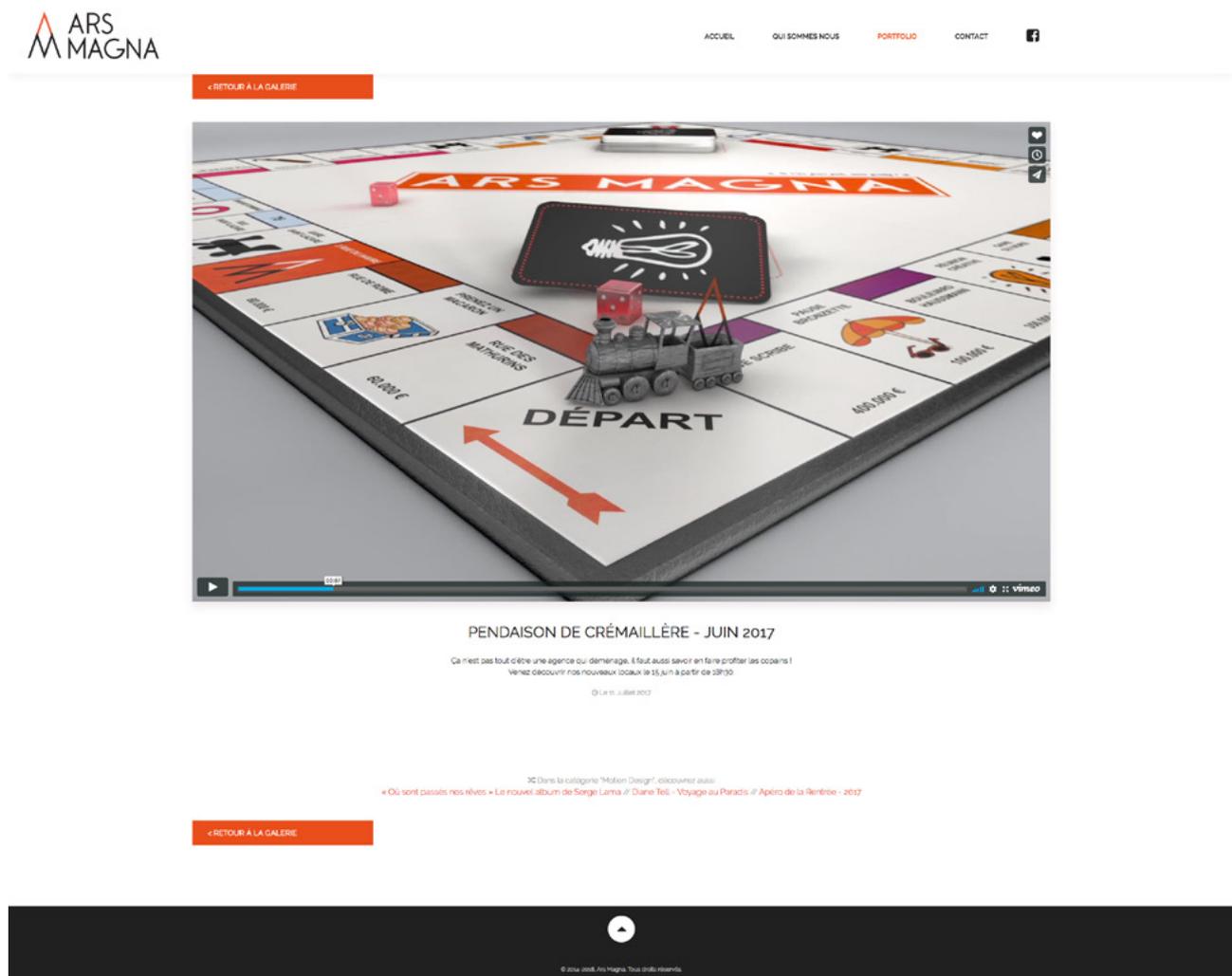


Dans l'exemple précédent, il faut souligner la présence d'une vidéo qui n'est pas un contenu obligatoire. Grâce au PHP, l'ajout de la balise la vidéo s'opère sous condition d'avoir récupéré du contenu depuis la base de données. De même pour les images d'illustration ou le texte de description.

Une version responsive pour mobile est disponible, mais non finalisée.

C) Page projet - motion design

Si la vidéo constitue le centre du projet et non un élément accessoire, il est important de la mettre en avant. Il est important qu'à l'ouverture de la page, elle soit la cible principale du regard de l'internaute.



La vidéo profite ainsi de l'espace de la page et passe devant tout le reste du contenu, y compris le titre situé en dessous et invitant à lire une éventuelle description et quelques illustrations. Le titre est de toute façon lisible en hover depuis la page générale du portfolio avant de cliquer, on assume que l'internaute a déjà eu l'opportunité d'en prendre connaissance. Y compris si ce dernier est arrivé sur la page depuis le slider ou le damier de projets récents.

Les vidéos sont stockées sur Vimeo ou YouTube. Grâce à des paramètres spécifiques à l'URL de ces dernières, on peut en forcer la lecture automatique. Il s'agit d'une pratique de moins en moins conseillée, car considérée comme intrusive pour l'utilisateur. Je suis d'avis que la lecture automatique est abusivement employée pour les publicités et certains contenus vidéo plutôt secondaires dans certaines pages. Son emploi ne serait pas pertinent dans le cas des projets du portfolio n'étant pas uniquement "motion design".

J'ai considéré que dans le cas des projets centrés sur la vidéo, il restait pertinent de l'employer bien que j'ai conscience que les navigateurs web embarquent de plus en plus un réglage par défaut interdisant la lecture automatique d'une vidéo.

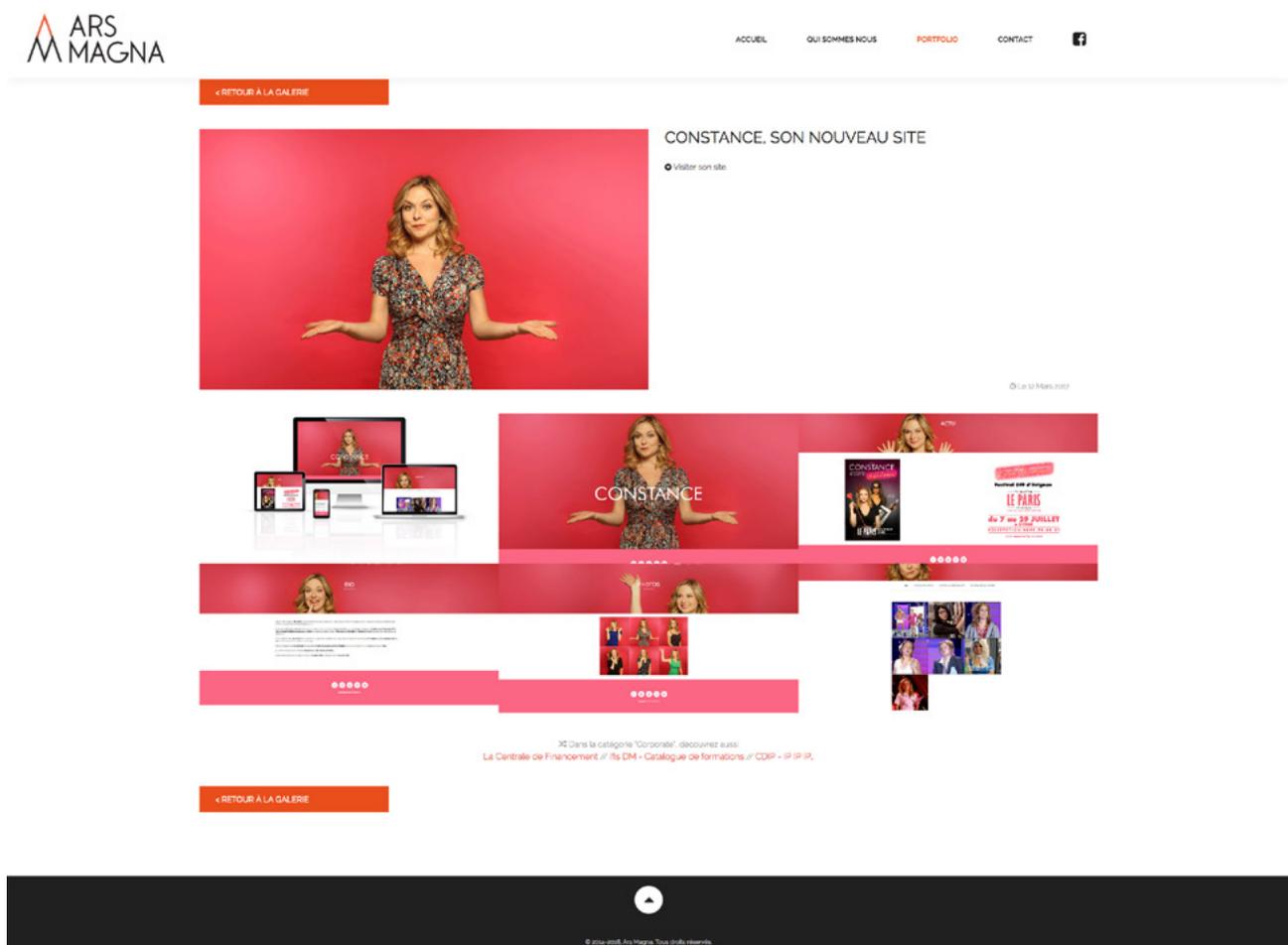
Si l'on attribue souvent à un graphiste des préoccupations liées à l'UX, on voit ici émerger une réflexion similaire en partant d'un questionnement lié au code. Rien de très étonnant quand on sait qu'un UX designer se doit d'être pluridisciplinaire pour couvrir le plus possible dans les détails la relation qu'un utilisateur peut avoir avec une interface ou un objet.

Je suis loin de disposer des compétences et des réflexes nécessaires pour être UX designer, mais c'est tout de même un troisième poids qui vient s'ajouter dans la balance pour dynamiser mes choix de graphiste ou de développeur. Ce poids agit quelque peu comme un médiateur dans mes réflexions graphiste/codeur pour désigner les choix les plus pertinents. Même si, comme je l'avais souligné auparavant, j'ai déjà commis des erreurs de jugement en me laissant porter par la curiosité technique !

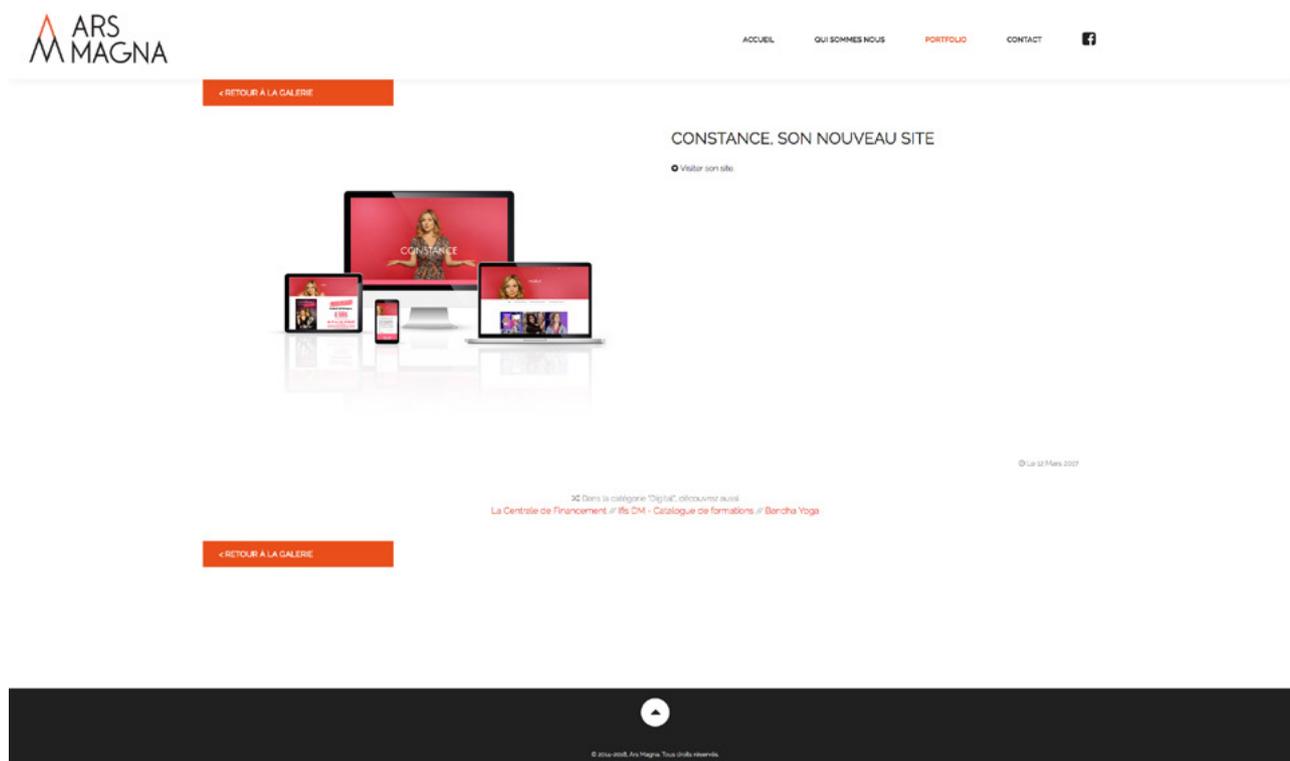
D) Page projet - digital

Cela n'était pas prévu au départ, mais je me suis décidé à créer un modèle de page projet spécifique au digital. Plus particulièrement pour mettre en avant les sites web créés par l'agence. Un choix principalement justifié par la nature des visuels alimentant ces projets : un mockup et, surtout, des captures d'écran du site.

En effet, l'usage de la mise en page générale est loin d'être avantageux pour ce cas de figure comme le démontre la capture suivante :



J'ai opté pour une solution simple consistant à ne montrer qu'une seule image, le mockup. Ce visuel est cliquable comme pour les autres pages et permet de lancer le visualiseur d'images pour naviguer dans les captures d'écran. On obtient ainsi une page plus agréable à l'œil.



Il aurait également été possible de conserver des visuels, mais en ne montrant qu'une petite partie des captures dans lesquels on aurait effectué un zoom. Je n'ai pas testé cette solution par manque de temps, mais également parce que je ne peux m'assurer que cette solution soit profitable en toute circonstance.

E) Page projet - divers détails

Dans un premier temps, j'ai opté pour un visualiseur avec jQuery. Puis j'ai profité de mon expérience dans la réalisation du slider pour concevoir un module JavaScript offrant une navigation dans des visuels. Il fonctionne comme un module en pop-up devant le site web, effacé derrière un fond blanc semi-transparent.



Pour naviguer d'une image à l'autre. Un rembobinage automatique ramène à la première image.



Aller de la première à la dernière image.

N'importe quelle autre touche du clavier aura pour effet de fermer le visualiseur.

Un clic montre l'image sélectionnée en grand, des commandes facilitent la navigation d'une image à l'autre bien que le clavier le permette également.

Depuis le back-office, l'administrateur peut configurer les durées d'apparition en fondu et d'animation entre chaque image.

Lors d'un précédent projet sous forme de jeu web, j'avais découvert comment appeler un évènement JavaScript depuis une touche du clavier. J'ai donc réemployé mes connaissances.

Les petits projets sont bénéfiques, car leur taille modeste offre l'avantage de pouvoir creuser en profondeur les détails de la fabrication et d'en retirer une expérience favorisant la conception des projets d'envergure.

Un petit module en fin de page propose de découvrir un maximum de trois projets dans la même catégorie que le projet visionné. Si le projet dispose de plusieurs catégories, PHP en choisit préalablement une au hasard.

Techniquement, j'ai eu un travail d'optimisation à fournir sur la sélection aléatoire de projets dont je parlerais plus loin.

✂ Dans la catégorie "Motion Design", découvrez aussi

« Où sont passés nos rêves » [Le nouvel album de Serge Lama](#) // [Pendaison de crémaillère - Juin 2017](#) // [Diane Tell - Voyage au Paradis](#)

Pour revenir sur la page générale du portfolio, deux boutons "Retour" sont accessibles en haut et bas de pages. Pour le coup, mon choix dans le comportement des rubriques du header m'a quelque peu déstabilisé. Cliquer sur la rubrique portfolio pour revenir à la page générale aurait pu être pertinent dans ce contexte, mais ici, la rubrique portfolio est désactivée et sert de scrollToTop comme prévu à la base. Je me suis rendu compte de ce problème à l'usage.

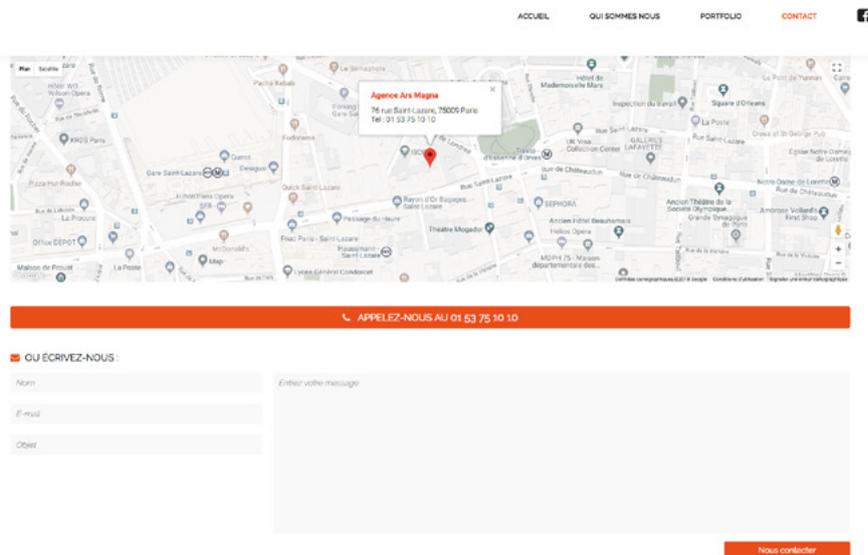
Pourquoi ce problème n'a-t-il pas été anticipé lors des wireframes ou de la maquette ?
Simplement parce que c'est au moment concret de coder que ma curiosité m'a amené à considérer ce comportement du header.

J'avais auparavant prévu un bouton "Retour" en bas de page ne s'affichant que si le contenu du projet était conséquent (plus de 3 images d'illustration ou présence d'une vidéo).
Mais j'ai finalement enlevé cette condition pour conserver un comportement régulier.

4) CONTACT

Sur la page Contact, il me fallait intégrer un Plan Google. Il existe une intégration simplifiée, mais également une API spécialement conçue à l'intention des développeurs front. En dehors de la curiosité, l'intérêt d'utiliser l'API est de pouvoir personnaliser l'affichage et le contenu du plan.

Le repère sur le plan est ainsi surmonté d'un panneau indiquant les coordonnées de l'agence. Cliquer dessus ouvre un onglet vers Google Map avec l'adresse de l'agence pour profiter d'informations complémentaires et de l'itinéraire.



C'était ici la première fois que je découvrais une API riche à intégrer sur un site. La documentation m'est apparue claire et rassurante. La richesse des possibilités et l'organisation des utilisateurs de l'API peuvent être particulièrement chronophages. Je n'ai pas pu m'attarder trop longtemps sur tous les détails !

J'ai dû faire face à un problème en CSS lié à l'usage de Flex qui ne m'offrait pas le résultat escompté sur Chrome malgré un affichage normal sur les autres navigateurs. Après quelques recherches, le problème était apparemment connu et la seule recommandation que j'ai trouvée préconisait de contourner le problème en changeant ses exigences de mise en page.

L'encart pour les coordonnées téléphoniques a finalement été repositionné entre le plan et le formulaire, avec une mise en forme le rendant plus attractif. J'ai également découvert la possibilité d'en faire un lien cliquable autorisant un appel automatique :

```
<a href="tel:+33153751010">Appelez-nous au 01 53 75 10 10</a>
```

Pratique si le site est consulté depuis un mobile, mais possible également depuis des ordinateurs, selon les logiciels installés.

Un Macintosh proposera de lancer un appel depuis FaceTime.

Le formulaire situé sous l'encart téléphonique ne fonctionne pas, car il nécessite que le site soit installé sur un vrai serveur capable de gérer l'envoi de mail. Il semble possible de configurer MAMP pour cela, mais la solution m'a semblé peu sûre et chronophage. J'ai donc fait le choix de ne pas trop me préoccuper du bon fonctionnement du formulaire de contact malgré l'ajout du code PHP censé l'utiliser. Cette partie est restée au stade théorique jusqu'à la réalisation d'un projet concret pour l'agence concernant le développement d'un site internet pour un conseiller en brevets d'invention ⁽¹⁾. J'ai ainsi récupéré mes données théoriques et réussi à faire fonctionner un formulaire de mail en y intégrant un captcha.

(1) Chez Ars Magna, j'ai eu pour mission le développement front-end du site d'un conseiller en brevets d'invention <http://www.cabinet-marconnet.fr/accueil.php>

5) QUI NOUS SOMMES ?

Le contenu de cette page est resté vierge pour plusieurs raisons. Au moment de la maquette, je m'étais contenté de récupérer le contenu de l'ancien site d'Ars Magna pour en revisiter la mise en page. L'avis de l'équipe a été d'en profiter pour remanier le contenu. Aussi j'ai considéré de compléter cette rubrique en dernier, après que l'équipe ait pu voir le fruit de mon travail sur la page d'accueil, de contact, sur le portfolio et le back-office. Dans l'optique de les aider à avoir une vision plus précise du contenu qui leur semblerait judicieux et en harmonie avec les autres rubriques et l'ambiance générale du site.

Les mois ont passé et un nouveau site WordPress de l'agence a vu le jour. Bien qu'il ait été envisagé que cette section pourrait être simplement supprimée, l'ancien contenu est toujours d'actualité, accompagné d'exemples de projets récents.

Cette rubrique restera donc vierge sur mon projet malgré quelques idées pour y apporter de la nouveauté, dont principalement l'intégration d'une photographie en 360°.

Cette idée a émergé de l'association de deux pensées. D'une part, Ars Magna a toujours considéré ces locaux comme devant être accueillant pour que ses clients s'y sentent bien. Et le déménagement en 2017 nous a offert de nouveaux locaux lumineux et deux terrasses avec vue sur la tour Eiffel qu'on nous envie souvent. D'autre part, j'ai eu de la curiosité pour des applications mobiles offrant la possibilité de créer des photographies en 360°. La plus simple et gratuite étant à ce moment-là "Street View" de Google.

Un bon nombre de modules basés sur HTML5, CSS3 et JS proposent de gérer des photos 360°. Citons par exemple Embed360 ⁽¹⁾, ou encore Pannellum ⁽²⁾ et aussi Photo Sphere Viewer ⁽³⁾.

(1) <https://leemark.github.io/embed360/>

(2) <https://pannellum.org>

(3) <https://photo-sphere-viewer.js.org>

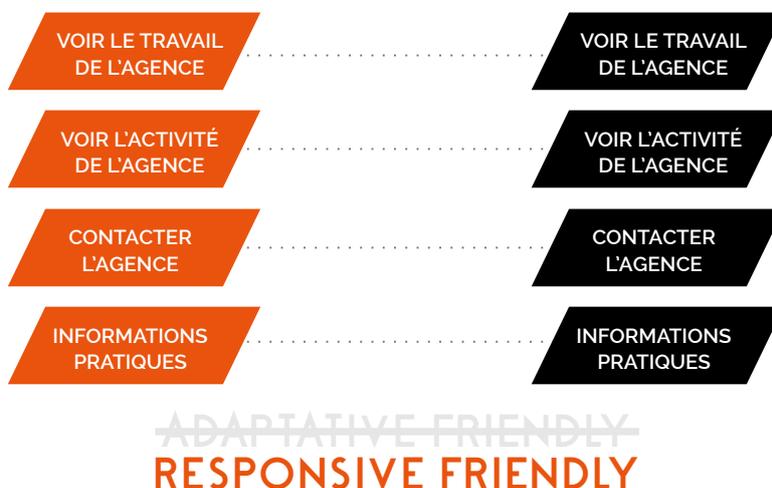
6) RESPONSIVE OU ADAPTATIF ?

Un site internet ne se résume plus à un affichage depuis un ordinateur classique et quand on réfléchit à sa version mobile, il existe plusieurs possibilités, dont le choix d'un site responsive ou adaptatif.

Un site web responsive est un site avec une ou des feuilles de style dont les propriétés d'affichage dépendront de certains paramètres tels que la taille de la fenêtre de l'utilisateur ou la plateforme employée (mobile, tablette ou desktop), etc. Un seul site s'adaptant à toutes les résolutions possibles est avantageux en termes de maintenance et de coût de production, mais demande un important travail en amont sur la maquette, car toutes les mises en page ne se prêtent pas efficacement au design responsive. De même qu'en fonction des plateformes, on peut considérer vouloir mettre en avant un contenu différent pour adapter sa stratégie selon les plateformes et cette souplesse peut requérir une structure HTML adaptée, ce que ne permet pas le responsive web design.

Et c'est pour cette raison que malgré la prédominance du responsive web design, la solution d'un site en adaptatif peut parfois s'avérer plus pertinente. En adaptatif, un serveur peut décider de renvoyer l'internaute vers la version desktop ou mobile du site web en fonction du contexte de l'internaute. Cela implique un HTML spécifique à chaque plateforme. Seuls le CSS et la base de données peuvent rester uniques et alimenter les deux versions du site. Bien entendu, chacun de ces sites peut également disposer d'un CSS responsive et cumuler les deux techniques.

Dans le cas du site d'Ars Magna, il ne m'a pas semblé avoir besoin de profonds changements d'usage entre les versions mobile et desktop. J'ai tout de même tenu à tester une méthode permettant de s'assurer de la stratégie à adopter, il s'agit du Gradient Chart ⁽¹⁾.



(1) <https://www.cennydd.com/writing/the-gradient-chart>

Après avoir listé les fonctions principales d'un site web, on crée deux colonnes (desktop puis mobile) dans lesquelles on copie ces fonctions en les réordonnant par ordre de priorité. En reliant par une ligne les mêmes fonctions d'une colonne à l'autre, on peut constater le besoin d'un site adaptatif si des lignes se croisent souvent. Si au contraire il y a peu ou pas de croisement de lignes, le responsive sera la solution idéale.

Le responsive a donc été choisi pour adapter le site sur mobile. J'ai repoussé l'écriture de la partie responsive du CSS en raison du travail important qui m'attendait côté back-end. Je n'ai finalement pas eu le temps de fournir une version mobile complète du site et elle restera donc à un état inachevé.

De plus, si ce travail devait être complété, il nécessiterait aujourd'hui une réécriture complète basée sur le SCSS. Aussi connu sous le nom de Sass ⁽¹⁾, il s'agit d'un préprocesseur CSS, un outil permettant de générer dynamiquement des fichiers CSS.

Il facilite grandement l'entretien des feuilles de style en y intégrant une logique d'écriture héritée des langages de script. Scripted Cascading Style Sheets ?

Apprendre le SCSS, c'est apprendre à redécouvrir et aimer le CSS. Je ne le quitte plus depuis que j'en ai appris les bases. La future version CSS4 tend à se rapprocher des pratiques initiées par Sass et les autres préprocesseurs CSS en introduisant par exemple l'usage des variables. C'est une bonne nouvelle bien qu'elle risquera de poser quelques problèmes aux utilisateurs de Sass le temps de la transition technologique, car l'usage de ces nouvelles variables CSS4 ⁽²⁾ n'est pas optimal pour tous les navigateurs.

(1) <https://www.alsacreations.com/article/lire/1717-les-preprocesseurs-css-c-est-sensass.html>

(2) <https://codepen.io/jakealbaugh/post/css4-variables-with-fallbacks-using-sass>

RÉALISATION DU BACK-OFFICE

1) INTRODUCTION



BACK-OFFICE

SLIDER

PUBLICATIONS

PORTFOLIO

RÉGLAGES

PHP MYADMIN
(nouvel onglet)

RETOUR SITE
(déconnexion)

D'usage privé, un back-office est une interface depuis laquelle administrer le contenu d'un site et en configurer l'affichage. Il s'agit de l'outil incontournable des CMS tel que WordPress. C'est la raison pour laquelle offrir une alternative à WordPress dans le cadre de mon projet ne peut se faire raisonnablement sans passer par la conception d'un back-office.

Actuellement pensé pour être hébergé dans un sous-dossier du projet global, on y accède via l'URL suivante :

agencearsmagna.com/bo-admin

Quelques notions de sécurité en lien avec l'accès à ce back-office seront abordées un peu plus loin.

Comme il m'importait de conserver une interface claire, l'esthétique générale est restée minimaliste. Une version mobile du back-office n'est pas à l'ordre du jour, car l'importation de fichiers images et l'ajout d'URL vidéo s'avèrent bien plus aisés depuis un desktop.

J'ai opté pour un header fixe et vertical sur la gauche, afin de trancher avec le site public. On y retrouve les éléments du site vitrine susceptibles de requérir une gestion de contenu.

- Le slider.
- Les publications.
- Le portfolio.

Ils sont suivis d'un onglet "Réglages".

Le header est clos par deux liens :

- Le premier vers PHPMyAdmin, l'interface graphique permettant une gestion fine de la base de données.
- Et "Retour site" déconnectant la session de l'administrateur pour le renvoyer vers le site public d'Ars Magna.

Les trois rubriques citées précédemment sont réparties dans un fichier PHP comprenant la vue générale et les formulaires d'ajout, de modification et de suppression de contenu.

Bien entendu, ces fichiers PHP ne présentent pas la totalité de ces actions sur la même page. L'affichage dépend de la méthode GET ⁽¹⁾ orientant le résultat vers l'une ou l'autre de ces actions.

GET est une méthode courante pour transférer une variable depuis une URL. Dans notre cas, elle prend la forme suivante :

agencearsmagna.com/bo-admin/slider.php?action=ajout

Ici, la variable "action" ajoutée à l'URL contient la valeur "ajout". Et au moment de charger le fichier "slider.php", PHP récupère la variable "action" qui par le biais d'une condition orientera le contenu affiché suivant sa valeur.

```
if ($action == "ajout") {
    /* J'affiche le contenu de la page d'ajout */
}
else if ($action == "edit") {
    /* J'affiche le contenu de la page de modification (et j'ai bien reçu l'identifiant de l'élément cible) */
}
else if ($action == "supp") {
    /* J'affiche le contenu de la page de suppression (et j'ai bien reçu l'identifiant de l'élément cible) */
}
else {
    /* Rien ne correspond alors j'affiche simplement la vue générale */
}
```

Dans les faits, rien n'empêche d'écrire directement l'URL à la main avec la valeur GET pour forcer un affichage. Il existe une autre méthode, POST, que j'emploie pour des notions de sécurité et nous reviendrons sur cet aspect bientôt.

(1) Une méthode est une commande liée au protocole HTTP spécifiant un type de requête, c'est-à-dire qu'elle demande au serveur d'effectuer une action.
https://fr.wikipedia.org/wiki/Hypertext_Transfer_Protocol#M%C3%Agthodes

2) DESCRIPTION DES RUBRIQUES

A) Slider

Comme pour les publications et le portfolio, la page du slider s'ouvre sur une page offrant une vue générale résumant son contenu.

Slider

Ajouter...

	CDIP - IP MATTERS 🔗 portfolio-25	Modifier	Supprimer
	BIRKIN/GAINSBURG - LE SYMPHONIQUE 🔗 portfolio-1	Modifier	Supprimer
	DÉCOUVREZ LE NOUVEAU SITE DE CONSTANCE 🔗 portfolio-23	Modifier	Supprimer
	BANDHA YOGA - UN SITE, PLUSIEURS YOGAS 🔗 http://bandhayoga.paris	Modifier	Supprimer
	« OÙ SONT PASSÉS NOS RÊVES » LE NOUVEL ALBUM DE SERGE LAMA 🔗 portfolio-9	Modifier	Supprimer

Conseils :

Minimum recommandé de slides : 3

Maximum de slides : 5

Tous les champs sont à remplir obligatoirement. On y trouve les données suivantes :

- Date, préremplie par défaut à la date du jour
- Titre, limité à 70 caractères pour simplifier les problèmes d'affichage
- Image, contenant le nom de l'image associé.
- Lien, dont le comportement a déjà été expliqué dans "Développement Front-end / Page d'accueil / Slider"

Ajouter un slide

Une date future publiera l'élément à la date indiquée.

 Date : 2018-06-01 14:57:59

Titre :

Obligatoire

Lien :

Obligatoire - Les URLs extérieures génèrent un nouvel onglet

*Privilégier un JPEG au format 1920*695px, optimisé pour le web.*

 : Aucun fichier sélectionné.

AJOUTER

Annuler

N'oubliez pas de lutter contre les torticolis !



Néanmoins, la rubrique slider dispose d'un fonctionnement un peu particulier. Mais surtout, j'ai accordé de l'importance à la façon de cadrer l'administrateur pour lui faciliter la réflexion sur le contenu. En effet, contrairement au portfolio ou aux publications qui assument de contenir beaucoup d'informations, un slider ne peut pas contenir une vingtaine de slides sans risquer de perturber l'utilisateur et alourdir sa navigation. À l'inverse, un ou deux slides ne sont pas suffisants pour justifier le contenu dynamique d'un slider.

C'est pourquoi j'ai inscrit un minimum de 3 slides dans les conseils en bas de page. Un maximum de slides bloque l'ajout de nouveaux éléments. Par défaut configuré à 5 slides, ce paramètre peut être modifié dans la rubrique "Réglages". Ce faisant, des emplacements disponibles facilitent visuellement la gestion de contenu du slider.

MAXIMUM DE SLIDES :

5  Modifier

MAXIMUM DE SLIDES :

7  Modifier

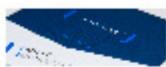
Modification effectuée :)

Slider

Ajouter...

<< Emplacement disponible >>

<< Emplacement disponible >>



CDIP - IP MATTERS
 portfolio-25



BIRKIN/GAINSBURG - LE SYMPHONIQUE
 portfolio-1



DÉCOUVREZ LE NOUVEAU SITE DE CONSTANCE
 portfolio-23



BANDHA YOGA - UN SITE, PLUSIEURS YOGAS
 <http://bandhayoga.paris>



« OÙ SONT PASSÉS NOS RÊVES » LE NOUVEL ALBUM DE SERGE LAMA
 portfolio-9

Conseils :

Minimum recommandé de slides : 3

Maximum de slides : 7

Il aurait pu être utile de rendre l'interface plus intuitive en admettant que cliquer sur un emplacement disponible renvoie sur le formulaire d'ajout de slide, quitte à créer un doublon du bouton "Ajouter..." déjà présent.

Dans les petits détails qui ont nécessité du temps de développement, le bouton "Ajouter..." est inutilisable grâce à l'attribut "disabled" seulement si le total de slides en base de données est égal — ou supérieur, on ne sait jamais ! — à la limite autorisée.

Si malgré la limite de slides atteinte on tente d'accéder au formulaire d'ajout en forçant via la méthode GET, le formulaire est encapsulé dans une condition vérifiant une nouvelle fois la limitation et invitera l'utilisateur à faire machine arrière :

 Ajouter un slide

Vous avez atteint le maximum de 5 slides !

Si la configuration de cette limitation se fait via la rubrique "Réglages", il y est impossible de baisser la valeur en dessous du nombre de slides déjà présent dans la base de données. En dehors du maximum de slides autorisés, les "Réglages" permettent de modifier l'animation du slider comme précisé auparavant.

B) Publications

Le fonctionnement est très simple et classique. Il accepte un titre et/ou un texte. Si aucun des deux champs n'est renseigné, le bouton d'envoi du formulaire est bloqué.

Le champ de texte est laissé totalement libre à l'administrateur, c'est-à-dire qu'aucun blocage ne lui empêche de renseigner du code HTML ou JavaScript. Une façon d'autoriser du contenu varié avec mise en page personnalisée.

Ajouter une publication

Une date future publiera l'élément à la date indiquée.

 Date : 2018-06-01 14:55:46

Titre :

Texte :

```
<p style="text-align:center;"></p>
```

AJOUTER

Annuler

Ce type de liberté peut être source de problèmes de sécurité dans un formulaire classique, mais je reviendrais sur cet aspect ultérieurement.

Pour encourager l'emploi du code HTML, une balise `<p>` renseignée avec un alignement centré est incluse par défaut dans le champ de texte. Il a été prévu côté JavaScript que ce texte seul ne puisse compter comme un contenu valide pour activer le bouton d'envoi.

Ajouter

< 1 - 2 - 3 >



CABINET MARCONNET - CONSEIL EN BREVETS D'INVENTION - Publié le 27 Février 2018
Le cabinet Marconnet fait peau neuve. Merci au Cabinet Marconnet d'avoir fait appel à Ars Magna pour réaliser la refonte complète de son site internet.

Modifier Supprimer



VOEUX POUR 2018 - Publié le 18 Janvier 2018
Toute l'équipe d'Ars Magna vous souhaite une année Heureuse, Créative, Radieuse, Florissante, Amusante, Graphique, Resplendissante, Faste, Triomphante.
Une vidéo est disponible.

Modifier Supprimer



CDIP - IP IP IP - Publié le 8 Janvier 2018
- hourra ! pour CDIP (« Cabinet Descazeaux - Industrial Property »), tout jeune cabinet de conseil en propriété industrielle qui a choisi Ars Magna po

Modifier Supprimer



BANDHA YOGA - Publié le 21 Septembre 2017
Un site, plusieurs yogas. «a href="https://bandhavogaparis.fr/accueil/" class="fa fa-arrow-circle-right" aria-hidden="tr

Modifier Supprimer



APÉRO DE LA RENTRÉE - 2017 - Publié le 1er Septembre 2017
Sans texte.
Une vidéo est disponible.

Modifier Supprimer



PENAISSON DE CRÉMALLÈRE - JUIN 2017 - Publié le 11 Juillet 2017
Ca n'est pas tout d'être une agence qui déménage. Il faut aussi savoir en faire profiter les copains ! Venez découvrir nos nouveaux locaux le 15 juin.
Une vidéo est disponible.

Modifier Supprimer



« ICI ET ALLEURS » IDIR - Publié le 9 Mai 2017
Avec Ici et Ailleurs, Idir nous offre un album de duo accompagné par nos plus grands artistes français. Un hommage aux classiques de la chanson française.

Modifier Supprimer



NATALIE DESSAY CHANTE SCHUBERT - Publié le 9 Mai 2017
Natalie Dessay nous plonge dans l'univers romantique du compositeur allemand accompagnée au piano par Philippe Cassard.

Modifier Supprimer



CHIMICHURRI, TALEBUM À QUATRE MAINS - Publié le 9 Mai 2017
Rencontre entre le jazzman français Baptiste Tridon et la percussionniste argentine Minino Garay. La tradition afro-américaine et celle de l'Amérique.

Modifier Supprimer



BIRKIN/GAINSBOURG - LE SYMPHONIQUE - Publié le 9 Mai 2017
Un album où Jane Birkin s'aventure et interprète les classiques de Serge Gainsbourg, accompagnée par un orchestre symphonique. Un album délicat en tout.

Modifier Supprimer

< 1 - 2 - 3 >

N'oubliez pas de lutter contre les torticolis !



C) Portfolio

Parce qu'il concerne le contenu le plus important du site, le portfolio est la portion la plus riche et complexe des trois. Le contenu obligatoire implique un titre et une image principale (nommée image0) en base de données. Tandis que le contenu optionnel propose un champ de texte avec les mêmes libertés que pour les publications.

Ajouter un projet ?

Une date future publiera l'élément à la date indiquée.

 Date : 2018-06-01 14:50:31

Titre :

Obligatoire

Texte :

HTML / Javascript autorisés

Lien Video :

Ajoutez l'URL de la page web (Youtube ou Vimeo)

*Privilégier des JPEG au format 1600*1200px, optimisés pour le web.
Image principale obligatoire.*

 : Aucun fichier sélectionné

 n°1 : Aucun fichier sélectionné

Un tag obligatoire.

Une sélection unique "Motion Design" ou "Digital" force une mise en page spécifique.

Ajoutez un ou plusieurs tags :

- Affiches
- Corporate
- Digital
- Merchandising
- Motion Design
- Musique

AJOUTER

Annuler

i) Vidéo

Un champ optionnel pour la vidéo coche ou décoche automatiquement le tag "motion design" suivant s'il est rempli ou vidé. Il implique une URL qui est réinjectée dans une balise `<iframe>` générée au moment de l'affichage sur la page du projet. Comme les vidéos peuvent provenir autant de Vimeo que de YouTube, il a fallu conditionner la génération de l'`<iframe>` avec des options spécifiques à l'un ou l'autre de ces services de streaming vidéo. On vérifie si l'URL de la vidéo contient "vimeo.com" pour lui attribuer des paramètres spécifiques, sans quoi on oriente vers les paramètres YouTube.

```
<iframe src="<?php
    echo $portfolio['video'];
    if ( preg_match('#vimeo.com#', $portfolio['video'])) { echo '?autoplay=1&title=0&byline=0&portrait=0';}
    else { echo '?autoplay=1'; }
?>" frameborder="0" webkitallowfullscreen mozallowfullscreen allowfullscreen></iframe>
```

Mais en amont de l'affichage, au moment de valider le formulaire en back-office, j'ai décidé de faciliter le travail de l'administrateur en retravaillant l'URL des vidéos.

En effet, le plus naturel pour un utilisateur est de copier-coller le lien direct de la page YouTube ou Vimeo. Or le lien présent dans le champ d'adresse du navigateur n'est pas un lien valide pour `<iframe>`. Selon les services de streaming, il faut se renseigner dans les fonctions de partage pour trouver la balise `<iframe>` correspondante où l'on trouve une version adaptée de l'URL. Je me suis donc renseigné sur le modèle d'URL de Vimeo et YouTube pour transformer l'URL de la page classique pour l'adapter à la balise `<iframe>` juste avant de l'inclure en base de données.

```
// Vidéo
if ($_POST['video'] != "" AND preg_match('#youtube.com#', $_POST['video'])) { // Youtube
    $video = preg_replace('#https://www.youtube.com/watch?v=(.*)#', 'https://www.youtube.com/embed/$1', $_POST['video']);
    if (preg_match('#(.*)&#', $video)) {
        $video = preg_replace('#(.*)&(.*)#', '$1', $video);
    }
}
else if ($_POST['video'] != "" AND preg_match('#vimeo.com#', $_POST['video'])) { // Vimeo
    $video = preg_replace('#https://vimeo.com/(.*)#', 'https://player.vimeo.com/video/$1', $_POST['video']);
}
}
```

Cela m'a pris beaucoup de temps pour rendre cette action aussi transparente pour l'utilisateur. Cette simplification exige un travail important de ma part en front et back-end pour alléger le travail de l'administrateur.

Même si je m'en doutais auparavant, je réalise par la pratique que beaucoup d'idées permettant de faciliter la vie requièrent un travail important dont les utilisateurs ne se rendent pas compte. Il s'avère très gratifiant que des apports considérables pour faciliter la vie des usagers passent aussi inaperçus.

En UX, on considère qu'un utilisateur qui ne se plaint pas est un utilisateur satisfait. Et cela demande beaucoup de travail. Cette logique se retrouve jusque dans le développement qui n'implique ici qu'une simple automatisation de tâche. Je trouve plutôt logique et rassurant que l'UX relie autant le travail de l'interface que celui du code, invisible pour l'utilisateur.

Plus j'approfondis et plus je constate qu'il y a bel et bien à travers la méthodologie UX la possibilité de gérer un équilibre entre mes compétences de graphiste et de codeur.

ii) Images optionnelles

Après l'image obligatoire, on propose à l'administrateur l'ajout d'une première image secondaire. Si l'on renseigne ce champ, un nouveau `<input>` apparaîtra proposant de charger une image supplémentaire. L'action se répétera pour un maximum de 6 images secondaires. Les 6 balises `<input>` servant à charger ces fichiers images sont en réalité présentes sur la page depuis le départ, mais dissimulé par une propriété `display:none` en CSS.

Ce choix n'a sans doute pas été le plus pertinent, car quitte à employer le JavaScript pour faire apparaître les `<input>`, autant les ajouter progressivement à l'arborescence HTML. Ceci est tout à fait réalisable en JavaScript et aurait été bien plus propre, mais aussi l'opportunité d'exploiter les notions de nœuds HTML pour intervenir sur le DOM ⁽¹⁾.

(1) Le DOM est une interface de programmation normalisée par le W3C, qui permet à des scripts d'examiner et de modifier le contenu d'un fichier HTML.
https://fr.wikipedia.org/wiki/Document_Object_Model

Une fois les champs images remplis, l'envoi en base de données range chaque image dans une colonne précise. Là encore, une idée qui fonctionne, mais mes lacunes en programmation m'ont amené à inutilement complexifier la programmation...

Quand des suppressions ont lieu, cela risque de créer des colonnes vides au milieu de la liste des images, posant problème dans l'affichage des balises `<input>` du formulaire tel que je l'ai décrit juste avant.



Pour régler le problème, j'ai dû écrire un bout de code qui assure le rangement des images sans aucun blanc entre elles. Ainsi l'affichage des `<input>` ne risque plus d'avoir de souci.



Avec le recul et l'expérience accumulée depuis, je réalise qu'une seule colonne "images" aurait pu suffire à stocker toutes ces images en base de données. Ces noms sous forme de chaînes de caractères auraient été concaténer avec une séparation tel un symbole "point virgule".

Lors de la récupération, une fonction PHP explode() aurait permis de redécouper les différents noms d'images en se servant du "point virgule" comme repère pour récupérer un tableau contenant chaque image bien séparée. Économie de rangement en base de données, mais aussi dans le code PHP.

iii) Catégories

Ce formulaire est un tronc commun pour les trois modèles de mise en page de projet, influencés selon le choix des catégories — ou tags — sélectionnées par l'administrateur. Les catégories se présentent sous forme de cases à cocher. Au minimum, il en faut obligatoirement une.

Un tag obligatoire.

Une sélection unique "Motion Design" ou "Digital" force une mise en page spécifique.

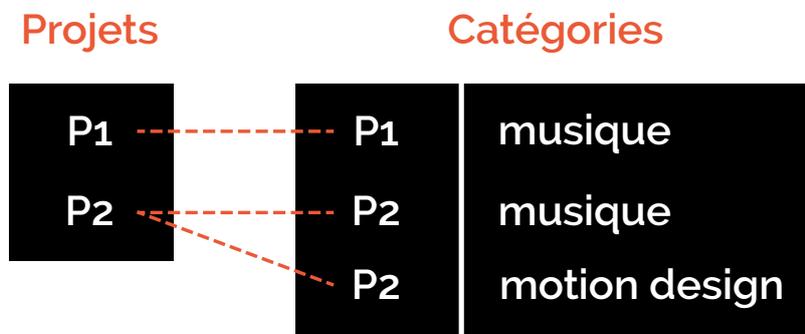
Ajoutez un ou plusieurs tags :

- Affiches
- Corporate
- Digital
- Merchandising
- Motion Design
- Musique

En base de données, la gestion des catégories implique un niveau de maîtrise que j'étais loin de pouvoir revendiquer. Cela a été l'occasion de faire de nouvelles erreurs, que j'aurais pourtant pu éviter, dont j'espère retirer des connaissances.

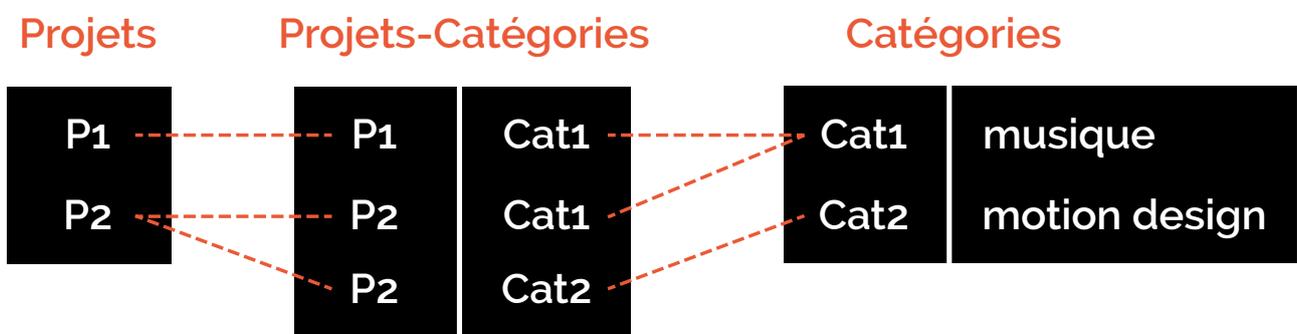
La solution employée :

Une table "portfolio_cat" qui relie l'identifiant d'un projet de la table "portfolio" à une chaîne de caractère étant le nom de la catégorie. Cela permet d'employer une table en moins comparée à la solution idéale. Le problème, c'est que le changement de nom d'une catégorie implique de modifier toutes les itérations du nom de cette catégorie. La méthode fonctionne, mais n'est pas pérenne sur le long terme.



La solution idéale :

Une table "catégories" ne répertorie qu'une seule fois le nom d'une catégorie. Et une nouvelle table servant à faire la liaison entre l'identifiant d'un projet et celui d'une catégorie.



Je connaissais la solution idéale avant de créer la base et me suis malgré tout porté vers ma première solution, source de problème. Quand j'ai suivi le tutoriel sur MySQL, j'ai suivi à la lettre la solution idéale tout en pensant qu'on pouvait faire plus simple.

Lors de ma mise en pratique personnelle, j'ai tenu à tenter ma solution et c'est pourquoi je constate aujourd'hui l'intérêt de la solution idéale présentée dans le tuto. Je n'ai donc pas de regret d'avoir eu l'occasion d'éprouver mon idée pour en constater ses limites.

Autre point important concernant les modèles de mises en page : Automatiser une tâche sur laquelle l'utilisateur peut avoir envie d'un certain contrôle risque d'être source de frustration. Un risque qui s'accroît si on augmente le nombre de catégories et de mises en page dédiées. Aussi la motivation initiale de rendre la tâche transparente pour simplifier le travail de l'administrateur n'était peut-être pas aussi pertinente.

Il se peut que cette idée d'automatisation ait été inspirée par la découverte technique de PHP, au détriment d'une attention apportée à l'utilisateur. Encore une fois, la démarche UX prouve son importance pour cadrer les priorités dans le travail.

3) DES DÉTAILS SIMPLIFIANT LA VIE

Dans le même esprit que la simplification d'usage des URLs de vidéo, un certain nombre de détails ont demandé d'investir du temps de programmation. En voici quelques-unes.

A) Date, affichage et utilisation

Du point de vue de la base de données, une date complète s'affiche suivant ce format :

"2018-02-27 15:47:12"

Or si je veux afficher la date sur le site public, celle-ci n'est ni présentable ni agréable à lire. C'est pourquoi j'ai écrit une fonction PHP dont le but est de convertir la date à un format plus naturel qui donne le résultat suivant :

"Le 27 février 2018"

Mon code n'utilise que la première partie de la date correspondant à l'année, au mois et au jour. Le mois est un remplacement simple par son équivalent en toutes lettres. Le jour est plus complexe, car il peut nécessiter d'enlever un "0" comme dans "Le 06 janvier 2016". Et dans le cas de "01", nous obtiendrons "Le 1er janvier 2016".

En dehors de son affichage, la date est également employée pour vérifier la validité de l'élément. J'ai conçu la requête SQL de telle manière qu'elle sélectionne tous les éléments sauf ceux ayant une date ultérieure. L'intérêt de cette méthode est de pouvoir programmer à l'avance l'affichage de nouveau contenu, à la seconde près.

B) Système de pagination

Dans le cas des publications et du portfolio, un contenu conséquent est à prévoir. Aussi, la vue générale côté back-end risque de s'allonger considérablement au fur et à mesure du temps, affichant un contenu dans le back-office avec une verticalité importante. Pour répondre à ce problème, un système de pagination a pour objectif de découper le contenu d'une rubrique en plusieurs pages :

< 1 - 2 - 3 >

En considérant 25 projets dans le portfolio, on peut décider de n'afficher que 10 projets par page. On aura ainsi 3 pages avec seulement 5 éléments sur la dernière. Le nombre d'éléments par page n'est pas statique, il dispose d'un paramètre dans la rubrique "Réglages". Il est accompagné d'un second paramètre dénommé "Page par bloc".

OPTIONS DE PAGINATION :

10 élément(s) par page.

9 page(s) par bloc.

Ce dernier paramètre est une anticipation à un problème à venir quand plus de contenu sera à prévoir. En admettant 280 projets, il faudrait actuellement un découpage en 28 pages. Notre problème vertical devient alors un problème horizontal !

1 - 2 - 3 - 4 - 5 - 6 - 7 - 8 - 9 - 10 - 11 - 12 - 13 - 14 - 15 - 16 - 17 - 18 - 19 - 20 - 21 - 22 - 23 - 24 - 25 - 26 - 27 - 28

Le bloc de pages permet d'insérer une ellipse pour ne pas montrer la totalité des pages en une seule fois comme le présentent les exemples ci-dessous :

< 1 - 2 - 3 - 4 - 5 - 6 - 7 - 8 - 9 ... 28 >

< 1 ... 20 - 21 - 22 - 23 - 24 - 25 - 26 - 27 - 28 >

< 1 ... 12 - 13 - 14 - 15 - 16 - 17 - 18 - 19 - 20 ... 28 >

Son réglage de base admet 9 pages par blocs. Un nombre obligatoirement impair, car la page actuelle est au centre du bloc. Ce réglage est également dynamique, voici 3 pages par blocs :

< 1 ... 13 - 14 - 15 ... 28 >

C) Découpage du code

Grâce à la fonction "**include**", PHP peut aller chercher du code PHP provenant d'autres fichiers PHP. L'intérêt de cette fonction, c'est de pouvoir alléger ses fichiers de travail pour s'y retrouver plus facilement et, surtout, éviter d'avoir à répéter inutilement du code.

Une chose que j'ai bien retenue durant mon autoapprentissage, c'est qu'un bon codeur est un codeur feignant ! Je me suis souvent demandé si des connexions étaient possibles avec le travail d'un graphiste.

Il se trouve qu'en observant les améliorations d'un logiciel comme Photoshop, un grand nombre de fonctions ont pour nouveauté de se substituer à des actions déjà possibles, mais bien plus chronophages. Plus que la possibilité de faire de nouvelles choses, elles permettent d'économiser du temps de production qui favorise, au choix :

- De profiter du temps économisé pour produire plus de choses.
- De restreindre les budgets.

La seconde option s'avère choisie bien plus souvent. Elle offre à de petites entreprises l'accès à des possibilités jusque là réservées aux grosses entreprises capables d'investir dans un budget conséquent. Il s'agit donc d'un mal pour un bien qui a des conséquences évidentes sur la façon dont évolue le secteur professionnel.

La diversité des compétences qu'on exige de plus en plus de la part d'un employé. En particulier les graphistes auparavant spécialisés et aujourd'hui sensible au 360°, c'est à dire disposant de compétences en print, mais aussi en motion et en web.

"**include**" permet donc de découper des fonctions à usage fréquent par exemple la connexion à la base de données, la conversion des dates et le système de pagination. Mais il faut également inclure le découpage logique d'un site internet, comme les head, header et footer. Tout cela suppose de gérer une multitude de fichiers. Nous y reviendrons bientôt dans la prochaine partie.

Le découpage offert par "include" me permet une comparaison imagée appuyant mon intérêt grandissant pour le code :

Créer des fonctions de manière individuelle tout en s'assurant de faciliter leur implémentation, c'est comme être un mécano virtuel qui conçoit des rouages pouvant s'emboîter de façon très diverse, créant une bien plus grande machine.

À travers ce jeu d'échelle de représentation, on comprend mieux l'intérêt d'organiser son code de manière rigoureuse.

DÉTAILS DU BACK-END

S'il y a un fossé entre un graphiste et un développeur, il y en a aussi entre un développeur front-end et un back-end. Jusqu'à présent, j'ai pu détailler mon activité en mixant ces deux aspects. Certains traits spécifiques au back-end subsistent et nécessitent une section dédiée.

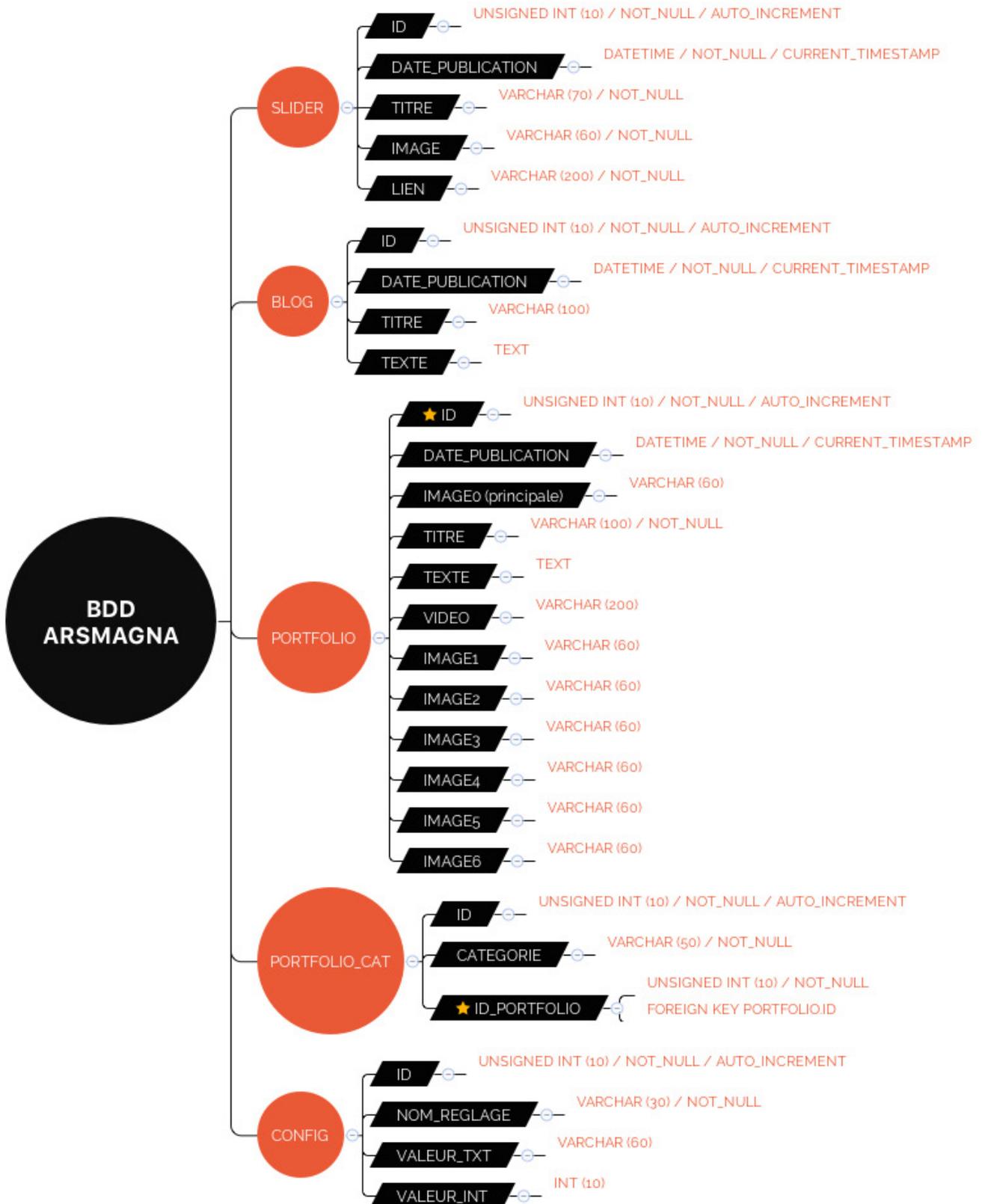
Du point de vue de l'exigence professionnelle d'un vrai développeur back-end, cette partie se devrait d'être bien plus développée qu'elle ne le sera dans mon cas. Plus que de simplement rapporter point par point mon activité, il s'agit de souligner les éléments dont j'ai eu conscience et de ce que j'ai pu apporter à mon niveau, dans les limites qu'impliquent ces multiples casquettes sur la tête d'un graphiste.

1) BASE DE DONNÉES

Elle contient les informations affichées dynamiquement comme les éléments du portfolio, du slider et du blog, mais aussi la configuration du site public et du back-office.

Comme son nom l'indique, le site public est le site visité par les internautes. Le back-office est quant à lui un site privé avec accès sur mot de passe, souvent réservé à un administrateur. C'est depuis ce lieu qu'on va pouvoir gérer le contenu du site à partir d'une interface graphique. Dans le cadre du projet, la gestion du contenu du site et l'interface graphique du back-office sont également sous ma responsabilité.

Une réflexion en amont m'a donc amené à définir une structure présentée sur la page suivante.



Une base de données est constituée d'une ou plusieurs tables dans lesquelles on classe des informations qui peuvent avoir des relations avec d'autres tables.

Si l'on prend la table liée au slider, on retrouve des colonnes avec types d'éléments correspondant à la date de création d'un slide, le titre, l'URL et l'image du slide. Chaque ligne de la table étant un slide précis. Il s'accompagne d'une colonne très spécifique aux bases de données que l'on nomme "id". Il s'agit d'un numéro unique au sein de la table qui est attribuée à un élément, une façon simple de sélectionner ou identifier un élément en base et que je serais souvent amené à employer.

Non, je ne suis pas un numéro ! Mais si c'est mon numéro qui me rend unique, alors je ne sais plus quoi penser...

La table du portfolio est bien plus complexe, car en plus d'intégrer un plus grand nombre d'informations, elle est en relation avec une autre table spécifique aux catégories des projets.

Après une session d'apprentissage, cette base de données est ma première mise en application de mes connaissances. Et bien entendu, plusieurs erreurs ont été commises et déjà soulignées dans les parties précédentes.

2) SÉCURITÉ

A) Get & Post

Parce qu'au départ, je n'avais pas saisi les avantages de la méthode POST, j'ai conçu mes premiers retours de formulaires du back-office avec la méthode GET. Tout fonctionnait bien jusqu'à ce que je réalise que sans même accéder au back-office via le mot de passe et donc, sans session utilisateur, je pouvais demander à la base de données de supprimer des éléments juste en renseignant correctement l'URL depuis le navigateur !

Une sensation étrange mêlée de fierté pour avoir testé une idée aussi tordue, mais également de frayeur si j'avais mis un tel site en ligne ! Cette situation m'a permis de comprendre l'intérêt de la méthode POST et de réécrire mon code en conséquence...

B) Les injections malveillantes

La base étant de considérer que l'utilisateur est malveillant et qu'il faut le cadrer. Il existe au moins deux types d'injections de code que j'ai anticipé.

Tout d'abord, les injections SQL qui sont une méthode d'attaque très répandue se concentrant sur une manière d'envoyer une requête SQL via PHP qui ne sécuriserait pas les données écrites par l'utilisateur.

Il est vivement conseillé de faire la requête en deux étapes. Cela forcera l'interprétation du contenu de la variable comme un contenu non exécutable et évitera certaines tentatives de piratages.

```
// Requête SQL non sécurisée
$req = $bdd->query("SELECT * FROM utilisateurs WHERE login='$login' AND password='$password'");

// Requête SQL sécurisée
$req = $bdd->prepare("SELECT * FROM utilisateurs WHERE login= ? AND password= ?");
$req->execute(array($login, $password));
```

Un principe assez similaire a lieu non pas au moment de la requête vers la base de données, mais au moment de la construction de la page par PHP.

La fonction PHP `echo()` permet d'injecter des informations dans le HTML. Mais si `echo()` injecte du code JavaScript, ce dernier sera interprété nos risques et périls. Là encore, nous pouvons encapsuler un contenu inconnu pour que le code JavaScript ne soit interprété que comme une simple chaîne de caractères.

```
<!-- Affichage non sécurisé -->
<h1><?php echo '<script type="text/javascript">alert("Oups !");</script>'; ?></h1>

<!-- Affichage sécurisé -->
<?php $variableSecurise = htmlspecialchars('<script type="text/javascript">alert("Oups !");</script>'); ?>
<h1><?php echo $variableSecurise; ?></h1>
```

Certains frameworks PHP intègre un langage dénommé Twig, prenant automatiquement en charge ces aspects sécuritaire. L'affichage devient alors d'une légèreté déconcertante !

```
{{ variable }}
```

C) Session et mot de passe

L'accès au back-office se doit d'être restreint à des utilisateurs autorisés. Le contenu PHP de chaque page du back-office est encapsulé dans une condition vérifiant si une session valide est active. Et pour se faire, le bon mot de passe doit être inscrit sur la page de connexion.

Accès back-office

Mot de passe : Valider

Mot de passe invalide :f

Une session est logée dans un cookie de session, qui disparaît si l'utilisateur ferme son navigateur. Un cookie peut être défini comme un témoin de connexion stockant des informations pouvant être requises de façon fréquente par un serveur.

En l'occurrence dans notre exemple : le mot de passe d'accès.

Il est possible pour renforcer la sécurité de supprimer la session sous condition par exemple, un long temps d'inactivité. Paramétré sur 15 minutes, il est possible de le changer dans la rubrique "Réglages" du back-office. Tout comme on peut y changer le mot de passe lui-même.

Il existe énormément de types de cryptage, le sujet est lui-même bien plus dense que tous les langages du web réunis si l'on considère que l'histoire de la cryptographie remonte à l'époque de l'Empire romain !

Il s'agit ici du sujet en face duquel je me suis senti le plus dépassé.

J'ai donc entrepris quelques recherches pour me renseigner sur une méthode de cryptage proposant le meilleur équilibre entre sécurité acceptable et accessibilité technique. J'ai donc opté pour un cryptage SHA-1 auquel j'ai ajouté un salage pour renforcer le mot de passe.

Le salage consiste à ajouter une chaîne de caractères, le plus souvent aléatoire, au cryptage. C'est le grain de sel qui vient compliquer les tentatives d'attaque.

D) HTTPS

Si HTTP est ce qu'on appelle un protocole de transfert hypertexte, la base de toute adresse d'un site internet, le "S" ajoute une notion de sécurité qui a pour but de confirmer au client distant que nous sommes bien le serveur authentique et non un serveur pirate déguisé.

Mettre en place cette sécurité a du sens depuis un vrai serveur. Je me suis renseigné sur sa configuration depuis MAMP mais je n'ai pas jugé bon de développer cette partie dans ma situation. Je note son existence, en espérant avoir l'occasion de l'employer un jour.

E) Un back-office en local ?

Si un mot de passe et une session permettent de limiter l'accès à l'interface d'administration du site, il n'y a pas plus sécurisé que de limiter ses différents chemins d'accès. On peut donc envisager un site dans une session MAMP depuis le poste local de l'administrateur qui commandera à distance la base de données du site public.

J'ai imaginé cette possibilité au fur et à mesure du développement du back-office. Le code actuel nécessiterait d'être réadapté.

F) Cacher l'extension .php

Pour empêcher les tentatives de piratage, il faut s'assurer de donner le moins d'informations techniques aux attaquants potentiels. Pour cela, il peut être utile de cacher l'extension de fin de fichier dans l'URL ou s'assurer que tous les tests d'extension — `.php .html .htm .shtml .asp` — renvoient vers un même résultat neutre. Cela se configure par des redirections serveur...

G) Pour aller plus loin

OpenClassRooms, site d'autoapprentissage en ligne qui m'a beaucoup servi pour acquérir toutes ces connaissances propose un cours sur les bases de la sécurité d'un site web disponible gratuitement à cette adresse :

<https://openclassrooms.com/courses/protégez-vous-efficacement-contre-les-failles-web>

3) CONFIGURATION DU SERVEUR APACHE

La configuration nécessaire sur Apache concerne donc en grande partie les redirections. Le principe consiste à écrire des liens sur les pages de notre site renvoyant par exemple vers `/portfolio-1` pour le premier projet de l'agence, plutôt que `/portfolio.php?id=1`

Sans configuration Apache, `/portfolio-1` ne renvoie vers rien d'existant et génère une erreur 404, code d'erreur signifiant que la ressource n'existe pas. Mais si on configure Apache pour capter la requête `/portfolio-1` afin qu'il la traduise en `/portfolio.php?id=1`, il peut nous trouver une réponse et l'enverra à l'internaute sous l'URL :

`"https://www.agencearsmagna.com/portfolio-1"`

Nous utilisons pour cela une syntaxe propre à Apache — et impliquant les regex décidément très utiles — ressemblant à ceci :

```
RewriteRule ^portfolio-([0-9]+)$ portfolio.php?id=$1
```

4) ERREUR 404 :(



Dans le cas d'une erreur 404, un message générique est affiché à la discrétion de chaque navigateur, mais nous avons le loisir de la personnaliser depuis Apache :

ErrorDocument 404 <https://www.agencearsmagna.com/404>

J'ai pris le temps et la liberté de rendre cette page un peu plus riche qu'un simple écran personnalisé. Une image et une citation aléatoires sont affichées comme le montrent les captures suivantes :



« Le temps n'est que la courte préface de notre existence ; l'éternité est le volume, qui ne sera jamais complet. »
Edward Young ; Les nuits (1742)



« On sait jamais, sur un malentendu ça peut marcher ! »
Jean-Claude Dusse ; Les Bronzés font du ski (1979)



« Un malentendu suffit souvent pour empoisonner l'existence. »
Sosthène de La Rochefoucauld ; Le livre des pensées et maximes (1861)



« La page demandée n'existe pas. N'a t'elle d'ailleurs jamais existé ? »

Le nom de l'auteur est un lien vers sa page Wikipédia. Un petit jeu est caché en blanc sur fond blanc, un jeu de trois dés 6 lancés aléatoirement pour tenter d'obtenir 421. On peut l'afficher en tentant hasardeusement de sélectionner le texte.



*« Un malentendu suffit souvent pour empoisonner l'existence. »
Sosthène de La Rochefoucauld ; Le livre des pensées et maximes (1861)*

421, bien joué !

Ce contenu n'est pas pertinent directement avec l'agence, mais communiquer à l'internaute de l'amusement par le code amène un peu de chaleur dans les relations qu'on entretient avec lui. Certains contenus de ce type, bien cachés, sont nommés des easter-egg. Il est l'occasion de faire preuve d'un peu de légèreté sans pour autant rompre avec la pratique du code et la page 404 se prête souvent à ce type d'occasion.

Dans les années 90, le navigateur NetPositive sur système d'exploitation BeOS était nommé ainsi parce qu'il affichait de petits haïkus lors des erreurs 404, très fréquentes durant cette époque d'internet.

*Cables have been cut
Southwest of Northeast somewhere
We are not amused.*

Bien que peu connu, un projet open source de réécriture du système BeOS a été baptisé HaïkuOS en hommage à NetPositive. Cet élément historique prouve l'importance que peut avoir la relation qu'on entretient au code. Et comme pour toute relation, il faut du temps pour la développer. Il faut accepter certaines maladresses et se forger un cadre équilibré. J'ai déjà eu l'occasion de souligner certaines maladresses que j'ai pu commettre sur ce projet.

5) OPTIMISATION DES PERFORMANCES

A) L'ordre des conditions

Quand on cherche à optimiser des instructions à exécuter en fonction de certaines conditions, leur ordre de vérification peut aider à accélérer le processus. Prenons comme exemple concret celui de la fonction servant à convertir la date en affichage naturel. Une portion du code concernant l'affichage du jour repose trois tests :

- Vérifier si le jour contient un "0" avant le chiffre pour le supprimer.
- Sauf s'il s'agit de "01" qu'on traduit par "1er".
- Dans tous les autres cas, on ne fait rien.

Du point de vue de l'optimisation de ces tests, on remarque très vite que le troisième point ne devrait pas arriver en dernier, car sur une base théorique de 30 jours, il y a 20 probabilités de ne pas avoir de "0 + n" que l'inverse ! Le cas le plus rare étant "01" : une probabilité sur 30.

Il peut donc être judicieux de structurer ses tests de façon à accélérer le comportement du code suivant les plus fortes probabilités. Nous obtenons donc :

- Vérifier s'il n'y a pas de "0 + n",
- Sinon vérifier qu'il ne s'agit pas de "01" pour enlever "0".
- Sinon remplacer simplement "01" par "1er".

Il s'agit ici d'une optimisation extrêmement mineure mais dont il m'a semblé utile d'avoir conscience pour adopter rapidement de bonnes pratiques.

B) Les requêtes SQL

Pour mettre en place le module de proposition de découverte de projets basés sur la même catégorie, j'ai dans un premier temps écrit une requête récupérant par ordre aléatoire et limité à 3 les identifiants de projets de même catégorie.

```
SELECT portfolio.id, portfolio.titre FROM portfolio
INNER JOIN portfolio_cat ON portfolio_cat.id_portfolio = portfolio.id
WHERE portfolio_cat.categorie = [categorie] AND portfolio_cat.id_portfolio != [identifiant du projet actuel]
ORDER BY RAND() LIMIT 3;
```

En l'état, cette solution fonctionne, mais peut montrer une importante lacune de performance sur une base de données conséquente. Pour cela, il faut comprendre le fonctionnement de la fonction RAND() :

Dans l'exemple de notre table de projet, cette fonction crée temporairement une nouvelle colonne contenant un nombre aléatoire unique pour chaque donnée de la table. Les données sont ensuite réagencées dans l'ordre croissant selon le nombre aléatoire attribué. Enfin, on ne conserve que trois de ces éléments suivant l'instruction "LIMIT 3". Pour une table contenant seulement 28 projets, les différentes étapes de cette fonction sont sans impact pour l'utilisateur. Sur le long terme, la lourdeur de ces instructions peut ralentir le temps d'exécution de la requête.

Dans l'idée que le portfolio puisse être lourd en projet et conserver une certaine rapidité, j'ai écrit une requête SQL simplifiée ne contenant que les id des projets de même catégorie en excluant le projet actuel. Ces identifiants sont rangés dans une variable tableau en PHP dont je tire aléatoirement une valeur en prenant grand soin de supprimer la ligne sélectionnée après récupération. Je réitère l'opération deux fois si la variable tableau est constituée de suffisamment de valeurs pour obtenir mes trois sélections de projets à proposer.

L'optimisation de cette requête n'est assurément pas la meilleure et a été longue pour moi à concevoir. Ce temps de réécriture a tout de même été très utile pour avoir l'occasion de découvrir des fonctions PHP, dont `explode()` qui m'est apparu comme une solution intéressante de gestion des images du portfolio. Je l'ai découvert malheureusement trop tard, mais on constate ici l'intérêt de prendre le temps pour chercher des solutions.

Investir un tel temps en programmation durant mon alternance a été une aubaine. De même que de déjà avoir un bagage de graphiste, car on peut difficilement imaginer acquérir des compétences croisées de graphistes et de codeur durant un même temps d'apprentissage.

C) Des fichiers images miniatures

Alors que je dépassais la vingtaine de projets dans le portfolio, j'ai commencé à remarquer un blocage de quelques secondes avant l'affichage de la page générale des projets sur le site public. Mon premier réflexe a été de penser que quelque chose dans le code ralentissait l'affichage, avant de réaliser qu'il s'agissait simplement des images qui n'étaient pas encore optimisées pour le web. Le damier de projet était rempli d'images haute définition !

Après avoir mis en place des recommandations pour les images sources, j'ai conçu un bout de code pour créer un double miniature et alléger chaque image au moment de leur import. Ainsi tout est rentré dans l'ordre. Le dossier d'images miniatures pèse 14,2 Mo contre 44,3 pour le dossier source.

La prochaine étape serait sans doute de ne pas afficher la totalité des projets sur la page générale, mais d'en cacher une partie et d'ajouter dynamiquement les images côté client quand l'utilisateur arrive en bas de page. Cette solution demanderait l'intervention du JavaScript.

6) ORGANISATION GLOBALE DU PROJET

A) Le code

Si le code PHP est découpé pour ne pas être répété, il faut prévoir des dossiers pour ranger tous les fichiers qui en résultent. Les fonctions récurrentes sont ainsi isolées du code pour être importées au besoin.

Il existe en programmation le modèle MVC, signifiant modèle-vue-contrôleur. Un principe d'organisation du code découpant les grandes tâches suivant une certaine logique pour en faciliter le développement et la maintenance. À ce stade de mes connaissances, j'avais connaissance du principe sans avoir eu l'occasion de le pratiquer. J'ai reporté mon contact avec le modèle MVC à beaucoup plus tard, lors de mon apprentissage de Symfony.

B) Les fichiers

Je ne vais pas décrire en détail l'arborescence des fichiers du site. Un plan en fin partie la présentera en image. Le back-office est isolé du reste du site dans un dossier spécifique pour les notions de sécurité et de souplesse déjà évoquées.

Les images chargées pour le portfolio et le slider ont un dossier spécifique, comprenant également un sous-dossier "m" pour "mini" contenant les versions miniatures des images.

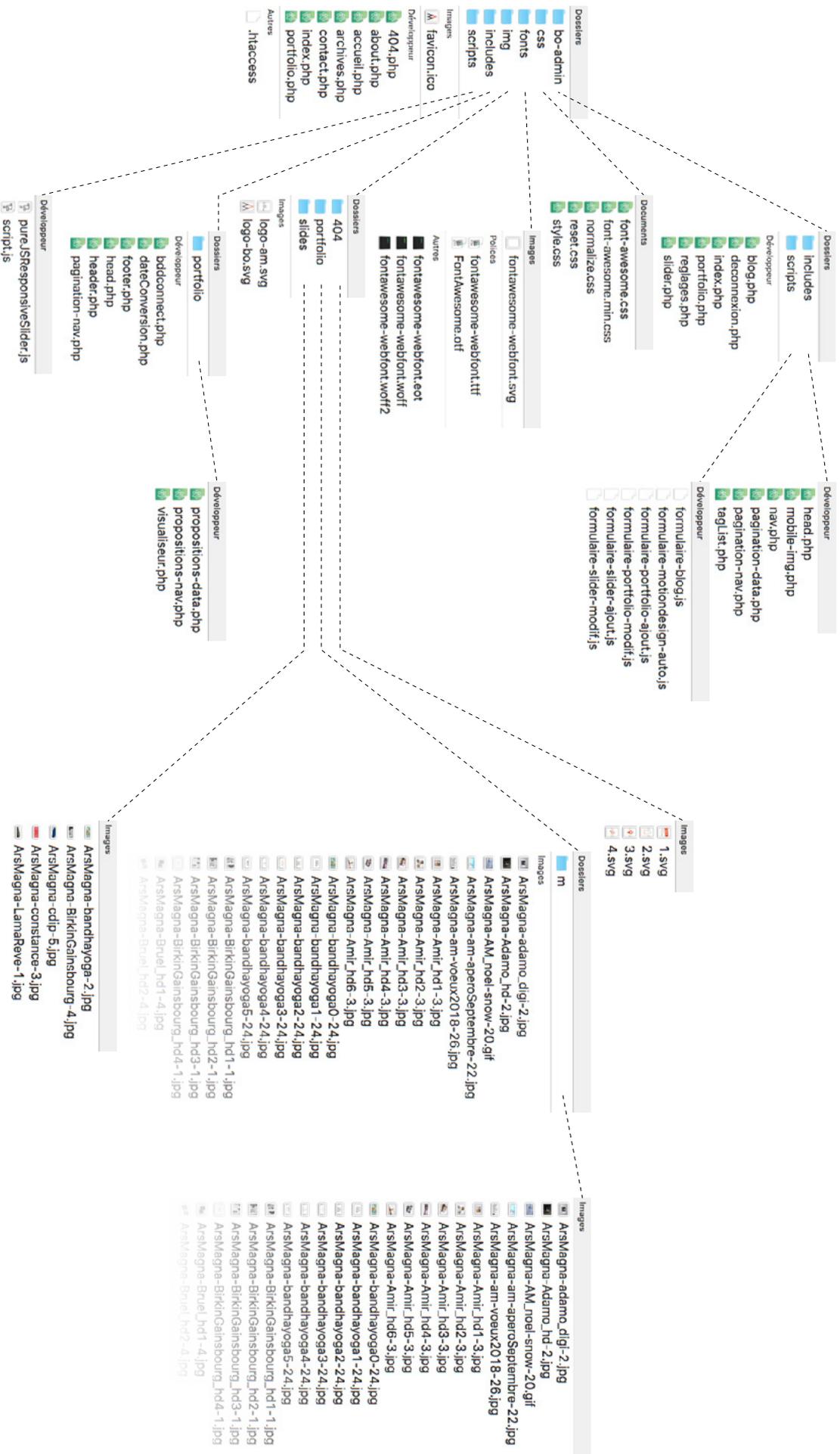
J'ai envisagé un système de nommage des images pour éviter d'écraser d'anciens fichiers portant le même nom. La structure de renommage est la suivante :

ArsMagna-[nom original]-[numéro unique].[extension de fichier]

- Le nom de l'agence est réinjecté dans le nom du fichier pour faciliter son référencement dans les moteurs de recherche d'images.
- Le nom original est conservé.
- Un numéro unique est attribué. C'est cet élément qui assure l'identité unique du fichier pour éviter les écrasements involontaires.

Ma première méthode a consisté à me servir d'un timestamp comme numéro unique. Mais la solution la plus simple était de reporter l'id de l'élément du portfolio dans cette partie du nom du fichier.

- L'extension du fichier, qui peut être .jpg ou .gif



N'oubliez pas de lutter contre les torticolis !



ET LE SEO ?

Acronyme de **Search Engine Optimization**, il concerne des règles à suivre dans la conception d'un site web pour assurer son bon référencement naturel. Ce type de référencement est gratuit, il prend du temps, mais peut offrir une bonne présence dans les résultats de recherche d'autant plus que l'utilisation de ces moteurs est devenue la principale méthode de navigation sur le web. Selon Alexa ⁽¹⁾ en 2017, Google est toujours premier dans le top 10 des sites internet les plus visités au monde et le reste du top concerne des sites et services web facilitant la navigation par recherche.

Parmi les règles à suivre afin de ne pas être pénalisé par les moteurs, il y a déjà le bon usage des balises HTML5. Pour s'assurer le respect de la démarche, on peut employer un outil en ligne nommé Markup Validator Service ⁽²⁾ qui est un outil officiel du W3C, consortium qui fait office d'autorité en la matière.

Le W3C est un organisme de standardisation présent depuis 1994, avec pour credo "Un seul web partout et pour tous". Cette volonté d'unifier le web dans ces méthodes de conception tend à simplifier le travail des concepteurs de tout bord, développeurs comme designers, avec au final un impact considérablement positif pour les internautes.

C'est avec cet objectif en tête que les moteurs de recherche tendent à pénaliser les sites web ne faisant pas l'effort de se plier aux règles de bonnes pratiques. Le SEO est donc une préoccupation qu'on ne peut ignorer.

S'assurer de répondre aux besoins d'accès pour les personnes en situation de handicap a également un impact positif sur le référencement naturel. Je me suis assuré de remplir tous les attributs "alt" des balises images, offrant une alternative textuelle aux images en cas de problème d'affichage ou pour l'usage de commandes vocales.

Une navigation souris et clavier a été partiellement implanté dans le site, en particulier pour le visualiseur d'images du portfolio.

Bien d'autres recommandations sont disponibles sur le site Web Accessibility Initiative :

<https://www.w3.org/WAI/>

L'emploi du responsive pour assurer une adaptation au mobile devient de plus en plus une obligation. Nous savons d'or et déjà qu'il sera bientôt nécessaire pour être recensé par Google.

(1) Alexa est un outil dédié au SEO lancé par Amazon <https://www.alexa.com/topsites>

(2) Disponible à cette adresse : <https://validator.w3.org>

Je n'ai malheureusement pas eu le temps de finaliser le responsive de mon projet. Un choix assumé, car j'ai préféré prioriser la découverte de la partie back-end.

En parallèle de ce projet, j'ai activement employé le SCSS sur des missions de l'agence, dont un site entièrement responsive, jusqu'à une adaptation du code JavaScript. Ces connaissances auraient pu influencer le développement de ce projet personnel, mais la contrainte de temps en aura décidé autrement.

Les robots scrutant le web à la recherche de sites à indexer s'intéressent également à la structure des URLs, à leur simplicité de lecture. C'est ainsi qu'en dehors d'une problématique de sécurité, la réécriture des URLs depuis la configuration du serveur Apache offre une réponse simple à deux problèmes différents.

Il existe des solutions alternatives, dont la possibilité d'effectuer un routage des URLs, depuis PHP mais cela nécessite de mettre en place un système comprenant un noyau interrogeant des routeurs, travaillant de concert avec des contrôleurs. Comme c'est le cas dans la majorité des frameworks PHP dont Symfony.

Une telle organisation du code justifie l'emploi rigoureux du modèle MVC évoqué plus tôt.

Le head, partie invisible d'un site web depuis sa page, se doit de contenir des éléments explicites et différenciant pour renseigner les robots.

L'ajout fréquent de contenu est une condition pour obtenir et maintenir un bon positionnement dans les résultats de recherche. Dans le cas de l'agence, de nouveaux projets tombent tous les mois. Sans compter l'usage des publications textes ou du rafraîchissement du slider.

Lors d'une semaine de cours, nous avons eu un contact très intéressant avec le SEO qui nous a fait réaliser l'importance d'un tout autre domaine que le graphisme et le code : le marketing. Nous atterrissons ici dans une logique avec ses mécaniques propres qui ne peuvent être improvisées. Il faudrait même sans doute envisager un équivalent à ce mémoire de la part d'une personne avec un bagage de graphiste codeur en quête d'un nouvel équilibre dans un univers complètement différent ! La question de la place de l'UX comme cadre de travail s'avère malgré tout toujours pertinente.

TEST UX

Maintenant qu'une bonne partie du site public et du back-office sont fonctionnels, il va être intéressant d'effectuer un test utilisateur. Il soulignera les points à améliorer pour le confort des usagers. Sachant qu'ici, il y a deux tests qui se croisent : celui des internautes et celui de l'administrateur.

J'ai prévu un ensemble de petites missions à réaliser, par un cobaye du nom de Clara Nigen. Elle est à l'aise avec l'informatique, familière avec l'UX et utilise déjà WordPress. Je lui ai fourni un fichier texte avec des exercices à effectuer dont voici la liste :

- Se connecter au back-office.
- Créer un nouveau slide. Modifier la limite de slider si elle est déjà atteinte. Intégrer un lien vers le projet déjà existant dans le portfolio.
- Modifier la dernière publication pour remplacer le "lorem ipsum" par un vrai texte et effacer toutes les anciennes entrées.
- Supprimer le projet le plus ancien du portfolio.
- Changer la troisième image d'un projet en particulier.
- Créer un nouveau projet (images fournies), ajouter un titre, un texte et un lien vers une vidéo YouTube. Le projet doit apparaître sur le site d'ici une semaine.
- Modifier les réglages du slider pour que son contenu ne s'affiche que durant une seconde et sans animation.
- Les projets récents en page d'accueil ne doivent afficher que 7 projets en catégorie "Musique".
- Changer la durée de la session administrateur ainsi que le mot de passe.

Je suis resté à ses côtés pour observer la scène. J'ai également filmé le test pour ajouter un peu de stress et faire ressortir les défauts du site. En UX, il est habituel de partir du principe que le cerveau note principalement l'existence de ce qu'il ressent comme négatif. Si le positif demande un effort de concentration, la démarche UX s'appuie sur le ressenti de ce qui trouble l'utilisateur. Donc si les points qui seront soulevés montrent des améliorations possibles, tout ce qui ne sera pas évoqué suggérera plutôt une réussite.

Sur le site public, le seul détail fâcheux concerne le comportement du header où la rubrique de la page actuelle sert de scrollToTop. Il est efficace partout, sauf dans la rubrique portfolio, car cette dernière dispose de deux niveaux d'arborescence. Et dans le second niveau correspondant aux pages projets, on aimerait pouvoir remonter à la page générale. Or ça n'est pas ce qui se passe. La question se pose alors. Faut-il concevoir une exception dans le comportement du header pour cette rubrique ? Et si une exception pour la rubrique principale du site est

un non-sens, faut-il revoir le fonctionnement global du header ?

Le test en back-office a permis de faire remonter plusieurs écueils. Le plus récurrent a été la liste des éléments contenus dans les parties slider, publications et portfolio. Par défaut, cliquer sur un élément envoi vers sa visualisation en contexte, sur le site public. Or Clara a cliqué avec l'envie instinctive d'accéder à la page de modification. Un changement s'impose donc !

La date est peut-être mal placée ou pas assez visible. L'ajout d'un calendrier pour sélectionner la date autrement que par son format DATETIME de la base de données rendrait son usage plus agréable.

Alors que j'avais conçu un code facilitant l'ajout de lien vidéo en modifiant l'URL de la page YouTube ou Vimeo, Clara est allée récupérer l'URL depuis un nouvel onglet apparu récemment sur YouTube. Ce dernier n'a pas été validé dans l'envoi du formulaire vers le serveur.

L'image principale servant à illustrer le projet dans le damier devrait être positionnée autrement pour se démarquer des images secondaires. On pourrait aussi la rebaptiser "vignette".

La rubrique "Réglages" n'est peut-être pas l'idéal. Retrouver des réglages concernant la page d'accueil, le visualiseur de projets ou le slider sur une même page est troublant. Il serait bon de retrouver les réglages propres à chaque partie sur leur propre page, pourquoi pas dans un onglet pour basculer entre contenu et réglage.

À la vue de ces retours, on peut constater que tous les points soulevés concernent des zones qui n'étaient pas couvertes par une réflexion UX. Le comportement du header est arrivé durant le développement front-end et la masse de travail a occupé mon attention au point de focaliser mon attention sur l'objectif sans remettre en cause la pertinence en matière d'ergonomie. Je ne regrette pas mon choix, car il m'a permis d'approfondir des points techniques importants autant en PHP qu'en JavaScript.

Bien que j'ai entrepris une recherche ergonomique sur le site public, j'ai omis cette étape pour le back-office qui s'est résumé à des wireframes à la main sur une feuille de papier. Le principe a été le même, mais bien moins approfondi.

Le résultat général m'a semblé positif et cohérent avec les lacunes présentes depuis le début du déroulement de mon projet. Les modifications peuvent impliquer des défis intéressants et qui seront sans doute rappelés dans de futurs projets !

BILAN

Depuis la réalisation de ce projet de site, Ars Magna a renouvelé son thème WordPress. La différence notable concerne la simplicité de l'affichage. Car la page d'accueil s'avère être directement une page de portfolio. Pas de slider, ni de publication.

Sans renier les choix effectués dans mon projet, je reconnais la sobriété et l'efficacité d'un tel modèle et j'aimerais m'en inspirer pour un potentiel book personnel, entièrement fait main. J'y ajouterais la possibilité de contrôler les projets présents au début du damier pour retrouver la stratégie de mise en avant manuel du slider, mélangé au damier actuel de projets récents.

La masse de travail que représentait un tel site dans une perspective de découverte était considérable. Je n'ai pas été en mesure de boucler complètement le projet, mais je reste satisfait d'avoir eu l'occasion d'approfondir les points essentiels que je voulais explorer.

Je retiendrais principalement les lacunes suivantes :

- Le responsive n'est pas complet et le CSS aurait mérité d'être réécrit en SCSS. Avec l'aide de COMPASS, un framework SCSS assurant la compatibilité des propriétés CSS entre navigateur.
- Considérant la masse d'éléments à gérer pour l'administrateur, une navigation par moteur de recherche aurait été une autre expérience intéressante à envisager.
- La page "Qui sommes-nous" est restée vide. Nous n'avons pas pu trouver un moment pour en discuter. Il reste dommage que je n'ai pas profité de cette page pour expérimenter un module JS pour intégrer des photos des locaux de l'agence à 360°.
- Le formulaire de contact par mail ne fonctionne toujours pas, mais ayant eu l'occasion d'en concevoir un pour un site client avec intégration d'un captcha, cette partie n'a plus été d'un grand intérêt.
- Une version multilingue du site aurait été intéressante à expérimenter, mais il me semble que toute la structure du site aurait nécessité d'être repensée en conséquence.
- D'où l'intérêt d'un respect rigoureux du modèle MVC dans l'organisation du code.
- Et bien sûr, l'encadrement UX qui mériterait de ne pas être oublié lors des découvertes techniques liées au code.

Ce projet m'aura permis d'engranger une somme de connaissance technique importante :

- Un de mes premiers développements front-end d'importance dont les maladroites m'ont aiguillé vers une méthodologie efficace.
- Un premier contact avec la notion de back-end, particulièrement floue.
- L'introduction à un langage de preprocessing HTML tel PHP pour commencer à penser dynamiquement ses pages web.
- Les problématiques d'organisation du code et des fichiers.
- Les considérations liées à l'optimisation et la sécurité.
- La démarche SEO

Bien qu'en manque de profondeur, ce regard large sur la technique a pour effet bénéfique de m'aider à mieux cerner le rôle de chaque expert en son domaine. Car en plus de m'offrir une autonomie certaine sur des projets d'envergure moyenne, je peux surtout m'assurer de mon efficacité dans un rôle en particulier en coordination avec le reste de l'équipe. Qu'il s'agisse d'être efficace sur ma mission ou d'apporter un soutien technique.

Les nouvelles connexions que ce contraste de compétences m'offre sont une source supplémentaire d'inspiration. Qu'elle soit graphique ou technique. Une profusion d'idées bienvenue qui demande un minimum de gestion pour ne pas risquer d'empiéter sur les besoins utilisateur. D'où l'importance de la démarche UX pour conserver un certain cadre.

Mais si l'UX peut apporter un équilibre dans le temps à investir entre production visuelle et code, quel équilibre trouver entre l'apport de ces expériences personnelles et celles d'une vraie situation professionnelle ? Car ma principale inquiétude est de pouvoir trouver une place professionnelle. Or un large champ de compétences n'est pas toujours une solution idéale. On ne trouve pas tous les jours des annonces d'emploi mêlant graphisme et code. Il ne me semble peu crédible de prétendre à un poste de développeur. Axer mes recherches sur le graphisme en apportant un supplément de code est actuellement à mes yeux la solution la plus viable. Est-ce une preuve de lucidité ou plutôt d'appréhension dans la qualité de mes compétences ?

D'autant plus que malgré la somme de connaissances assimilées, il me manque encore des technologies en passe de devenir incontournables comme AngularJS ou Symfony...

Peut-être aussi que plutôt de focaliser sur l'emploi, il me faille orienter mes recherches vers un certain type de structure. Durant mon BTS Audiovisuel Montage, nos professeurs nous avertissaient déjà des fusions de postes à venir dans le secteur traditionnel de la vidéo, considérées comme une hérésie. Il est vrai que si envisager un motion-designer-réalisateur-cadreur ne choque plus beaucoup aujourd'hui, une telle concentration de compétences n'offre pas le même niveau d'expertise qu'une équipe dédiée. Mais ce manque d'expertise tente très souvent d'être compensé par des outils informatiques de plus en plus sophistiqués.

La conception de nouveaux outils influence sans ménagement l'évolution du marché du travail et impose continuellement de s'adapter pour survivre. Si des métiers disparaissent, de nouveaux viennent les remplacer. Sans être foncièrement supérieur ou inférieur, simplement plus adapté au contexte.

Les nouvelles technologies de la vidéo permettent d'amoinrir drastiquement les coûts de production. Pour une qualité certes en deçà des solutions traditionnelles, mais tout à fait adaptées à internet, un média qui n'est pas totalement innocent dans cette mutation qui ne se limite pas qu'au secteur professionnel. Ainsi, malgré l'aspect négatif mis en avant par mes formateurs de mon BTS, je conçois que cette évolution offre à de petites et moyennes entreprises l'opportunité d'oser se confronter à de grosses structures difficilement concurrençables sur un terrain plus traditionnel. Le secteur du web offre ainsi une chance de relancer le secteur professionnel audiovisuel, actuellement sclérosé, dans une forme de renaissance numérique.

Sur un autre plan, sera-t-il possible que l'évolution de mon ancien secteur trouve un écho favorable dans celui du développement web ? On parle effectivement de plus en plus des développeurs full-stack à qui l'on donne volontiers la charge des développements front et back-end, allant jusqu'au développement mobile ! Il est intéressant de noter qu'un développeur full-stack dispose tout de même d'une spécialisation et que certains domaines lui resteront limités ⁽¹⁾. Le front-end évolue très vite et le Big Data implique des problématiques face auxquelles on ne peut s'improviser expert.

(1) Un article intéressant à l'égard du développeur full-stack : <http://blog.ippon.fr/2017/08/07/developpeur-full-stack-oui/>

J'ai parcouru des discussions sur internet qui m'ont justement rappelé les débats du secteur audiovisuel... Et malgré ce que peut inspirer mon cheminement sur le projet personnel décrit dans ce mémoire, je ne me sens pas de prendre position dans ce débat.

Je pense qu'il est bon de trouver son équilibre entre plaisir du travail et capacité d'emploi. La sécurité de l'emploi est une préoccupation devenue importante. Nous nous adapterons comme nous devons le faire tous les jours à tout un tas d'échelles. Parce que l'évolution se fera naturellement et ne peut être forcée dans une direction sans conséquences fâcheuses. Et je ne peux anticiper une évolution similaire en développement web, secteur dont l'évolution est d'une rapidité fulgurante comparée à l'audiovisuel.

Dans une logique assez différente, mais connectable vient le cas des slasheurs. En prenant conscience de la raréfaction des emplois stables, il y a cette volonté des slasheurs de cumuler différents métiers. Autant pour répondre à un besoin financier, par envie de ne pas s'encroûter, mais également en recherche d'une vocation lors d'une période de transition.

De là à envisager un emploi de fleuriste en parallèle d'un poste en motion design avec du développement web, je dispose encore d'une certaine marge.

À la sortie de cette formation de deux ans, ma principale inquiétude, c'est de pouvoir trouver du travail, surtout un poste fixe. Ce champ large de compétences saura-t-il être un poids qui penchera du bon côté de la balance ?

GLOSSAIRE

Administrateur : C'est la personne en charge d'un site internet. Elle est responsable de son contenu et doit avoir accès à des fonctions d'ajout, modification et suppression de contenu.

AngularJS : Un framework JavaScript libre et open source conçu par Google. Il permet de développer des pages web. (Wikipédia : <https://fr.wikipedia.org/wiki/AngularJS>)

API : Application Programming Interface. C'est une interface de programmation applicative agissant comme un ensemble normalisé de classes, de méthodes ou de fonctions qui sert de façade par laquelle un logiciel offre des services à d'autres logiciels. (Wikipédia : https://fr.wikipedia.org/wiki/Interface_de_programmation)

Authoring DVD : Processus de création d'un DVD-Vidéo à fins d'utilisation dans une platine DVD de salon. La production complète d'un DVD-Vidéo comprend plusieurs étapes. Compressions de la vidéo, graphismes et animation des interfaces de navigation, intégration et programmation des éléments, inscription sur DVD. (Wikipédia : https://fr.wikipedia.org/wiki/Auteuring_DVD)

Balise : Les balises HTML sont les éléments de base du codage HTML d'une page web. Invisibles pour l'internaute (sauf en activant l'affichage du code source), elles sont interprétées par le navigateur pour la composition de la page. (Définitions Marketing : <https://www.definitions-marketing.com/definition/balise-html/>)

Bannière : Bandeau publicitaire dans une page Web, de forme originellement rectangulaire. Les bannières sont généralement actives. En cliquant, l'internaute est dirigé vers un autre site, souvent marchand. (Mercator-Publicitor : <http://www.mercator-publicitor.fr/lexique-publi-cite-definition-banniere>)

Back-end : C'est un terme désignant un étage de sortie d'un logiciel devant produire un résultat. On l'oppose au front-end (aussi appelé un frontal) qui lui est la partie visible de l'iceberg. (Wikipédia : <https://fr.wikipedia.org/wiki/Backend>)

Back-office : C'est l'ensemble des activités de supports, de contrôle, d'administration d'une entreprise. Ce service collabore avec le front-office pour offrir l'affichage des données. (Wikipédia : https://fr.wikipedia.org/wiki/Back_office)

Classe : On peut utiliser l'attribut class pour appliquer des styles CSS aux éléments d'une page. Une classe peut caractériser plusieurs objets (identiques ou non). (AlsaCréations : <https://www.alsacreations.com/article/lire/535-quelle-est-la-difference-entre-une-classe-class-et-un-id.html>)

Cinema 4D : Un logiciel de création 3D développé par la société allemande Maxon. C'est un outil permettant la modélisation, le texturage, l'animation et le rendu d'objets 3D. (Wikipédia : https://fr.wikipedia.org/wiki/Cinema_4D)

CMS : Un système de gestion de contenu ou SGC (content management system ou CMS en anglais) est une famille de logiciels destinés à la conception et à la mise à jour dynamique de sites Web ou d'applications multimédias. (Wikipédia : https://fr.wikipedia.org/wiki/Syst%C3%A8me_de_gestion_de_contenu)

Desktop : Format d'usage d'un site ou application web depuis un ordinateur traditionnel, généralement posé sur un bureau.

Div : Balise HTML gérée sous forme de bloc et permettant de regrouper des éléments pour mise en page.

eCard : Une carte virtuelle (appelée aussi e-carte, e-card ou cybercarte) est à la carte postale ce que le courrier électronique est au courrier postal. (Wikipédia : https://fr.wikipedia.org/wiki/Carte_virtuelle)

eMailing : L'e-mailing, publipostage en ligne¹ ou publipostage électronique² est une méthode de marketing direct qui utilise le courrier électronique comme moyen de communication commerciale de masse pour envoyer des messages à un auditoire. (Wikipédia : <https://fr.wikipedia.org/wiki/E-mailing>)

Erreur 503 : Code HTTP d'erreur signifiant «Service temporairement indisponible ou en maintenance». (Wikipédia : https://fr.wikipedia.org/wiki/Liste_des_codes_HTTP)

Flex : Les Flex Box CSS sont un standard CSS³ de disposition des éléments dans une page web. Ce standard permet d'avoir un design adaptatif à l'écran. Les éléments peuvent être réagencés selon que l'on est sur un téléphone portable ou un écran d'ordinateur. (Wikipédia : https://fr.wikipedia.org/wiki/Flex_Box_CSS)

Fonction : C'est une entité informatique qui encapsule une portion de code (une séquence d'instructions) effectuant un traitement spécifique bien identifié (asservissement, tâche, calcul, etc.) relativement indépendant du reste du programme, et qui peut être réutilisée dans le même programme, ou dans un autre. (Wikipédia [https://fr.wikipedia.org/wiki/Routine_\(informatique\)](https://fr.wikipedia.org/wiki/Routine_(informatique)))

Footer : Un footer ou pied de page désigne la partie basse d'une page web qu'on retrouve en général sur toutes les pages d'un site web. Le footer est donc inclus dans le gabarit de page de l'outil de CMS pour permettre sa multiplication sur toutes les pages. (Définitions Marketing <https://www.definitions-marketing.com/definition/footer/>)

Framework : Désigne un ensemble cohérent de composants logiciels structurels, qui sert à créer les fondations ainsi que les grandes lignes de tout ou d'une partie d'un logiciel (architecture). (Wikipédia : <https://fr.wikipedia.org/wiki/Framework>)

Front-end : Partie visible d'un programme nécessitant d'être développée.

GIF : Format d'image numérique couramment utilisé sur le web. Il accepte la transparence et surtout les animations. (Wikipédia : https://fr.wikipedia.org/wiki/Graphics_Interchange_Format)

Head : Balise HTML fournissant des informations générales (métadonnées) sur le document, incluant son titre et des liens ou des définitions vers des scripts et feuilles de style. (MDN : <https://developer.mozilla.org/fr/docs/Web/HTML/Element/head>)

Header : Partie supérieure d'un site qui comprend généralement les principaux éléments de navigation et qui reste visible quelle que soit la page du site sur laquelle on se trouve. Il a pour principales vocations d'orienter le visiteur, d'indiquer les différents contenus et services du site et éventuellement de rassurer. (Définitions Marketing : <https://www.definitions-marketing.com/definition/header-site-web/>)

Hover : Une pseudo classe permettant de spécifier l'apparence d'un élément au moment où l'utilisateur le survole avec le pointeur, sans nécessairement l'activer. (MDN : <https://developer.mozilla.org/fr/docs/Web/CSS/:hover>)

Illustrator : Logiciel de dessin vectoriel conçu par Adobe. Très employé pour la création de logo, l'illustration et parfois le maquetage.

Indesign : Logiciel d'Adobe pour la mise en page de document comme un flyer, un catalogue... ou ce mémoire.

Maquette : Une maquette de site web est une ébauche d'un site Internet qui est présentée sous forme statique à l'annonceur commanditaire. (Définitions Marketing : <https://www.definitions-marketing.com/definition/maquette-de-site-web/>)

Motion design : Un domaine du graphisme spécialisé dans l'animation. Il peut couvrir les effets spéciaux et l'habillage d'une séquence vidéo.

Photoshop : Logiciel de retouche d'image conçu par Adobe. Contrairement au vectoriel, il traite donc des images constituées de pixels.

Print : Le print recouvre l'ensemble des domaines en rapport avec l'impression de document. Un écran d'ordinateur ou de téléphone n'est donc pas la plateforme finale du print.

Regex : Les expressions régulières sont une chaîne de caractères permettant de décrire un ensemble variable par l'utilisation d'une syntaxe précise, qui se retrouvent dans une foule de langages et d'outils. (EduTech : http://edutechwiki.unige.ch/fr/Expression_r%C3%Agguli%C3%A8re)

Responsive : Le responsive web design englobe les techniques de conception de contenus Internet qui permettent de proposer des contenus autoadaptables en fonction des interfaces de consultation utilisées par le visiteur. (Définitions Marketing : <https://www.definitions-marketing.com/definition/responsive-web-design/>)

ScrollToTop : Fonction d'une page web permettant de lancer une animation de défilement afin de revenir en haut de page.

Slide : Élément contenu dans un slider.

Slider : Élément d'une page web contenant plusieurs slides défilant automatiquement à un rythme régulier ou uniquement sur contrôle de l'utilisateur.

SQL : Langage informatique normalisé servant à exploiter des bases de données relationnelles. (Wikipédia : https://fr.wikipedia.org/wiki/Structured_Query_Language)

Symfony : C'est un ensemble de composants PHP ainsi qu'un framework MVC libre écrit en PHP. Il fournit des fonctionnalités modulables et adaptables qui permettent de faciliter et d'accélérer le développement d'un site web. (Wikipédia : <https://fr.wikipedia.org/wiki/Symfony>)

UI : Dispositif de dialogue homme-machine, dans lequel les objets à manipuler sont dessinés sous forme de pictogrammes à l'écran, de sorte que l'utilisateur peut utiliser en imitant la manipulation physique de ces objets. (Wikipédia : https://fr.wikipedia.org/wiki/Interface_graphique)

URL : Adresse désignant une chaîne de caractères utilisée pour identifier les ressources du World Wide Web. (Wikipédia : https://fr.wikipedia.org/wiki/Uniform_Resource_Locator)

UX : La notion d'expérience utilisateur a vocation à compléter celle d'utilisabilité. L'expérience utilisateur est moins strictement fonctionnelle puisqu'elle met l'accent sur l'impact émotionnel cumulé à un bénéfice rationnel. Il est entendu que la démarche est bien de créer une expérience agréable. (Wikipédia : https://fr.wikipedia.org/wiki/Exp%C3%Agrience_utilisateur)

Wireframe : Schéma utilisé lors de la conception d'une interface pour définir les zones et composants qu'elle doit contenir. (Wikipédia : [https://fr.wikipedia.org/wiki/Wireframe_\(design\)](https://fr.wikipedia.org/wiki/Wireframe_(design)))

WordPress : Système de gestion de contenu libre et open source. Les fonctionnalités de WordPress lui permettent de créer et gérer différents types de sites Web : blog, site e-commerce, site vitrine ou encore portfolio. (Wikipédia : <https://fr.wikipedia.org/wiki/WordPress>)

WYSIWYG : Désignent en informatique une interface utilisateur qui permet de composer visuellement le résultat voulu, typiquement pour un logiciel de mise en page, un traitement de texte ou d'image. C'est une interface intuitive l'utilisateur voit directement à l'écran à quoi ressemblera le résultat final. (Wikipédia : https://fr.wikipedia.org/wiki/What_you_see_is_what_you_get)

