

Table des matières

INTRODUCTION	3
CHAPITRE I . OBJECTIF DU SITE	6
CHAPITRE II: ANALYSE DE L'EXISTANT	10
CHAPITRE III. CRÉATION DU PROJET	24
CHAPITRE IV. OUTILS UTILISES	32
CHAPITRE V. ÉTAPES DE CRÉATION	41
CHAPITRE VI. DESIGN	49
CHAPITRE VII. L'INTERFACE	57
CHAPITRE VIII. GESTION DES LANGUES	72
CHAPITRE IX. GESTION DES LISTINGS	80
CHAPITRE X. INSCRIPTIONS	83
CHAPITRE XI: COMPTES UTILISATEURS	88
CHAPITRE XII. COMMUNICATION	92
CHAPITRE XIII. ADMINISTRATION	95
CHAPITRE XIV. GESTION DES ARTICLES	97
CHAPITRE XV. PERFORMANCE	99
CHAPITRE XVI. API	102

INTRODUCTION

Ces dernières années, les besoins en terme de locations de logements sont de plus en plus fréquents et exigeants surtout de la part des jeunes. En effet, contrairement aux générations précédentes, les étudiants sont dorénavant prêts à « s'expatrier » afin de suivre un enseignement dans l'école la plus réputée, à perfectionner une langue étrangère dans le cadre d'un Erasmus et vivre une expérience humaine dans un autre pays.

Ces besoins de la part des jeunes et de leurs parents nécessitent toujours la recherche d'un logement et cette partie peut parfois se révéler compliquée. Les sites de locations présentent parfois des lacunes pouvant rebuter l'internaute : nombreux clics indispensables afin d'obtenir une information, inscription laborieuse et nécessitant parfois trop de champs obligatoires et personnels, manque d'informations quant aux divers outils,... Ces différents points ont pour conséquence de frustrer l'utilisateur voire de le perdre...

Site de location de logements pour étudiants, Akoter (akoter.julien-roland.be) évoque à la fois le type de logement ainsi que la proximité de ces logements par rapport aux différentes écoles. Ayant été moi-même locataire de kots durant mes années d'études, il m'a également fallu traverser les difficultés liées à la recherche d'un logement. C'est pourquoi mon choix de projet de fin d'études s'est porté sur la création d'un site de recherche de locations en Belgique, optimisé et utilisable par tous indépendamment des connaissances informatiques de chacun.

Différents éléments permettent de me démarquer des autres sites web de location de logements mais, selon moi, les plus importants sont les suivants : la carte permettant une recherche pertinente et précise d'un logement, la possibilité d'une interaction entre propriétaire et locataire automatisée incluant la disponibilité du logement et, enfin, l'intégration des agences immobilières.

D'un côté plus « professionnel », ce projet m'attirait inexorablement de par sa diversité : il me donnait la possibilité de développer mes connaissances générales et de m'améliorer dans plusieurs domaines augmentant ainsi ma polyvalence et me permettant de mieux cibler mes points faibles de manière à y remédier.

D'autre part, il me semblait important de m'assurer que j'avais ou non les capacités pour réaliser un projet d'une telle envergure seul et du début à la fin. Une fois dans le monde du travail, c'est avec fierté que je pourrai présenter ce projet personnel à différents employeurs potentiels. Celui-ci consti-

tuera ma carte de visite une fois mis sur le marché.

Afin de mener à bien ce projet, deux étapes essentielles ont été réalisées avant de passer à la pratique. Dans un premier temps l'analyse des principaux sites web de location de logements afin de déterminer leurs points forts et leurs points faibles. Dans un second temps, le listing des éléments importants par ordre de priorité et définir les différents outils à employer.

Une fois ces deux phases accomplies, nous allons à présent passer à la pratique : étapes de la création de mon site web et mise en place de divers outils liés à mon site.

Je vous laisse à présent, Mesdames, Messieurs découvrir mon site web et vous en souhaite une bonne visite et une bonne lecture de mon travail.

CHAPITRE I.

OBJECTIF DU SITE

Nous pouvons constater que de plus en plus d'étudiants sont scolarisés «loin» de leur domicile et sont dans le besoin d'un logement proche de leur école. C'est ainsi qu'ils se retrouvent en quête du logement étudiant idéal dans une région précise.

Afin de présenter des qualités supérieures aux sites déjà présents sur le marché belge, les objectifs principaux de ce site sont la recherche, la communication et l'interface.

En ce qui concerne le locataire, les personnes ciblées sont naturellement les étudiants ou leurs parents, à la recherche d'un logement pour leur(s) enfant(s). De plus, disponible en anglais (également en Néerlandais pour le nord du pays), le site est également ouvert aux personnes ne résidant pas en Belgique, facilitant ainsi la recherche de logements dans le cadre d'Erasmus.

Par conséquent les candidats «locataires» ont besoin de pouvoir filtrer les logements en fonction de divers paramètres : situation géographique, prix du logement,...

Une fois les filtres réalisés, le candidat «locataire» doit être en mesure de trier les résultats : par prix, par date de mise en ligne de l'annonce,...

Ensuite, il doit pouvoir scanner visuellement les différentes annonces afin de trouver le plus facilement celle potentiellement intéressante.

Finalement, il doit être capable d'examiner les détails de l'annonce pour décider si le logement correspond ou non à ses besoins. De plus, il doit éventuellement pouvoir prendre contact pour avoir davantage d'informations, l'ajouter aux favoris ou se proposer pour le logement. Cependant, la plupart de ces manipulations nécessitent une création de compte d'utilisateur.

D'autre part, notons que le site propose également la mise en ligne de locations pour les propriétaires. Il octroie des avantages et des services supplémentaires pour les agences immobilières.

Le site doit également être facile d'utilisation étant donné qu'il est susceptible d'être consulté par un public plutôt large. En effet, les étudiants peuvent rechercher eux-mêmes le logement de leurs rêves mais les parents prennent souvent l'initiative de rechercher le logement pour leur(s) enfant(s).

D'autre part, les propriétaires sont aussi susceptibles de naviguer sur le site pour visualiser d'autres logements concurrents avant de s'inscrire.

Par conséquent, les propriétaires seront peut-être intéressés par la visualisation de logements concurrents dans leur région. Ils doivent aussi pouvoir filtrer et trier comme les locataires. Ceci pourrait devenir un futur outil pour les propriétaires, permettant de cibler certains points négatifs comparés à d'autres logements à proximité...

S'il veut ajouter son annonce sur le site, le propriétaire doit d'abord passer par l'étape d'inscription utilisateur, c'est-à-dire la création de son propre compte. Une fois cette démarche accomplie, les champs requis du compte doivent être complétés pour mettre en ligne des annonces (adresse, téléphone, ...).

Lorsqu'il remplit son profil, le propriétaire peut toutefois s'ajouter en tant que professionnel, cette distinction lui permettant ensuite d'afficher son agence.

Après avoir répondu à ces diverses formalités, il doit pouvoir visualiser son annonce, vérifier sa mise en ligne, aller sur son compte et utiliser les outils disponibles pour les propriétaires.

Après un certain temps, il pourrait souhaiter consulter le nombre de vues de ses annonces, répondre aux messages et demandes d'éventuels locataires et gérer la location (modification, voir ses locataires, ...).

Notons que toute personne inscrite sur le site peut désirer à un moment ou à un autre d'apporter des modifications à son compte, voire le supprimer,...

Ainsi, il est primordial que le site soit facile à utiliser. En effet, le public cible fait partie d'une tranche d'âge assez large.

Cependant l'objectif n'est pas de simplifier outre mesure le site et ses outils. Il faut parvenir à offrir la meilleure communication possible afin de faciliter son usage et sa compréhension, sans toutefois ralentir les utilisateurs aguerris.

Il est illusoire de faire comprendre tous les fonctionnements du site à un nombre important de personnes ayant des niveaux de connaissances différents. Le meilleur moyen d'y arriver peut se trouver parmi les tests utilisateurs.

CHAPITRE II. ANALYSE DE L'EXISTANT

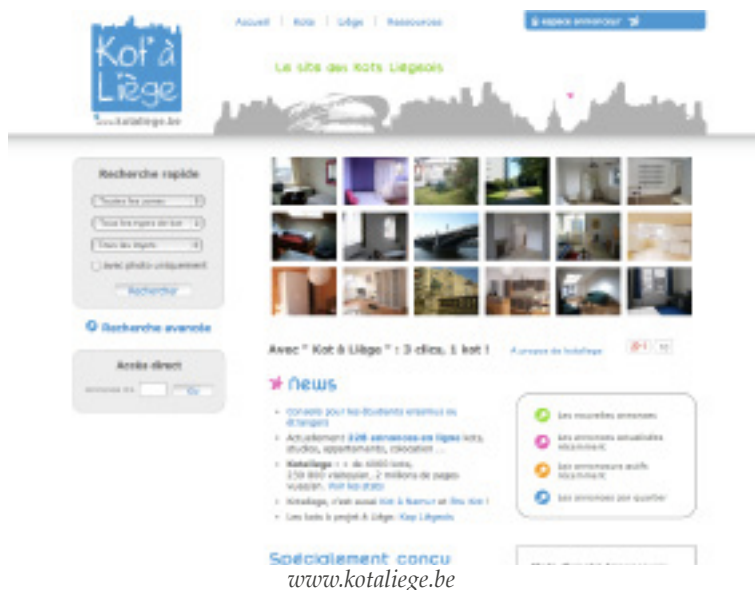
Pour réaliser mon propre site web, il m'a semblé utile d'analyser, dans un premier temps les différents sites de location en Belgique, principalement en Wallonie.

Dans un deuxième temps, je me suis inspiré des sites populaires dans la location en règle générale tel que Abritel.

De manière plus secondaire, pour des parties moins essentielles, je me suis également inspiré de Redfin pour sa carte, de Typeform pour ses formulaires, de Bootstrap pour le style de certains éléments et de Facebook pour la structure de certaines parties du compte utilisateurs.

Dans cette analyse seront abordés les sites de Kotaliege, kots.be, Appartager. Ensuite des sites qui m'ont inspiré tels que Abritel, Chambre à louer, Redfine et Typeform.

KOT' À LIÈGE



Page d'Accueil :

Design : une fois sur la page d'accueil, aucun élément important n'est mis en valeur. L'œil est attiré par le logo (à cause de sa forte masse de couleur contrastant avec le fond blanc) et la galerie d'images (une image attire par défaut le regard mais également à cause du nombre d'images, du bloc que ce nombre crée et de l'animation dont elles disposent).

Ces deux éléments sont-ils ceux à voir en premier ? Sont-ils les plus utiles à la page d'accueil ?

Une fois sur le site, le logo ne devrait pas constituer un élément visuellement dérangeant à la visite. Certes, il est évident

que l'utilisateur doit s'en souvenir mais d'autres moments sont plus opportuns pour mettre en valeur le logo (page de connexion).

Dans ce cas-ci, les images ne servent à priori à rien puisqu'elles ne sont pas cliquables, trop petites et par conséquent difficilement discernables pour des personnes avec une vue légèrement réduite.

Quels sont les éléments à mettre en valeur ?

La recherche des kots, la connexion ou l'inscription si l'internaute n'est pas déjà enregistré.

En premier lieu, la recherche. Les utilisateurs vont vouloir visualiser les kots et s'ils en trouvent un d'intéressant, ils souhaiteront s'enregistrer pour avoir un logement.

Dans d'autres cas de figure, un propriétaire peut désirer se créer un compte pour pouvoir louer son bien. Dès lors, il sera à la recherche d'un champ d'inscription ou si la personne a déjà son compte, elle peut souhaiter vérifier s'il n'y a pas de demande en attente et donc chercher à se connecter dès son arrivée sur le site.

Et qu'en est-il des autres éléments ?

Ils doivent avoir la même valeur d'accroche afin de ne pas déranger le système de recherche et la connexion / inscription.

D'autre part, les personnes trouveront facilement l'information après un peu de recherche sur la page d'accueil.

Et qu'en est-il du contenu ?

Lister les derniers kots mis en ligne me semble être une idée intéressante. Lorsqu'une personne est à la recherche de «LA» bonne affaire, elle est susceptible de consulter le site régulièrement voire tous les jours dans l'espoir qu'un nouveau kot à sa convenance soit disponible.

Ainsi, afficher les derniers kots mis en ligne sur la page d'accueil permet d'éviter de devoir naviguer dans le site.

Ce dernier point pourrait être notre troisième élément d'accroche.

Que penser du système de recherche utilisé ?

Dans le cas de certaines recherches, aucun résultat n'est disponible, c'est-à-dire aucune location répondant à nos critères et nous n'avons pas la possibilité d'en être informé avant d'avoir cliqué sur le bouton de recherche.

Dans le cas où aucun résultat n'est disponible, l'utilisateur doit revenir en arrière et éventuellement refaire une nouvelle recherche. Ce genre de comportement est assez pénible et présente les conditions pouvant amener à la perte d'utilisateurs.

D'autre part, avoir des résultats en temps réel permet à l'uti-

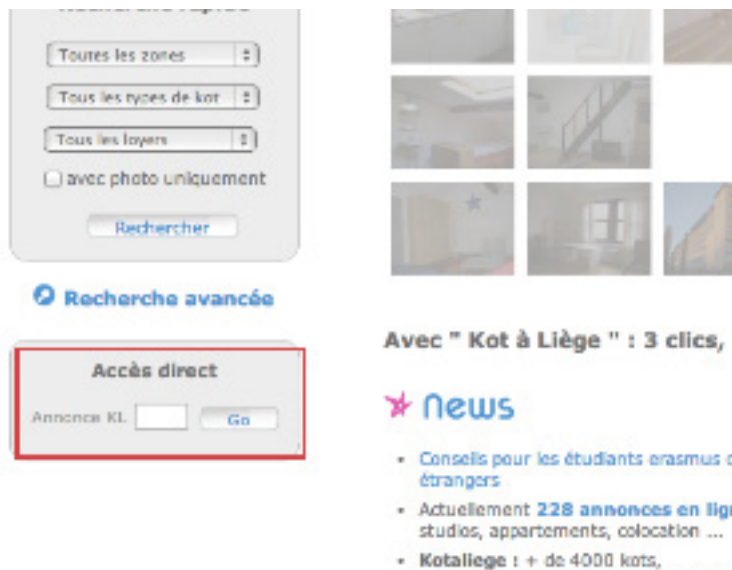
lisateur d'améliorer sa recherche ou de la corriger plus rapidement gagnant ainsi du temps.

En dehors de ces quelques inconvénients, le système propose malgré tout les champs les plus importants (zone, type de logement, prix,...), il est simple mais empêche de filtrer comme on le voudrait.

Qu'en est-il de l'interaction ?

Dans la majeure partie du site, il manque de l'interaction utilisateur qui aurait pu permettre de communiquer des informations utiles et de faciliter la navigation et la compréhension.

Par exemple, dans la page d'accueil (voir image ci-avant),



dans le champ «accès direct», nous sommes en principe supposés encoder le numéro du bien recherché et cliquer sur OK.

Cependant, si nous laissons le champ vide et que nous cliquons sur OK, le site nous redirige vers une page nous indiquant qu'il n'a pas trouvé celle en question.

Nous devrions être informés que le champ a été laissé vide et qu'il faut obligatoirement le compléter sans quoi il est normal que la recherche ne fonctionne pas.

Pas de photo ? Pas d'annonce !

Obliger les propriétaires à ajouter des photos directement pendant la création du bien peut les décourager mais en mettre une est un minimum pour attirer l'œil des locataires potentiels.

De plus, un site contenant de belles et grandes images est toujours plus attirant.

En effet, tout ce qui est réel attire le regard. Une image suscitera plus l'envie de cliquer sur l'annonce et communiquera plus d'informations qu'un long paragraphe.

Le choix des couleurs, du gris clair sur du blanc ?

Le gris utilisé dans ce cas-ci est trop clair. Assis normalement ce n'est pas trop dérangeant, mais dans le cas où nous sommes installés selon un mauvais angle de vision, il devient

difficile à lire malgré une bonne vue.

Faut-il mettre des éléments en valeur ?

Certains éléments auraient pu être mis en valeur tels que le prix de la location ou la date de disponibilité du logement.

Cela faciliterait le «scan» de l'annonce et fatiguerait moins vite l'utilisateur qui pourrait dès lors augmenter son temps de passage sur le site.

Toute l'annonce doit-elle être cliquable ?

Oui et je l'ai appris à mes dépens lors de précédents projets.

En faisant des tests sur les clics des utilisateurs sur une page web, on remarque qu'ils ne sont pas très précis, donc augmenter la taille des zones cliquables permet de leur faciliter la vie.

Page d'une annonce (http://www.kotaliege.be/annonce/show/page/2/quartier_0/2/quartier_1/3/quartier_2/4/quartier_3/5/type_kot_id/3/tranche_loyer/3/choix_dispo_key/99/order/loyer)

Dans ce cas-ci, il s'agit de l'URL de la page d'une annonce, guère parlante, trop longue et écrite en partie en anglais. Il me semble que pour le **Search Engine Optimize** (= pour Optimiser l'impact sur les Moteurs de Recherche), il est préférable d'avoir des URL plus parlantes : titre/région ou éventuellement donner un nom à l'annonce et utiliser un nom unique.

Se proposer pour un logement

Le seul moyen de contacter l'annonceur est via un message ou un numéro de téléphone.

Il en est ainsi dans la plupart des sites de locations, cependant mon objectif sera d'être plus organisé et de permettre une gestion interne. Cela demande plus de travail mais sera plus simple pour le propriétaire et fournira une liste précise des anciens locataires.



<http://www.kots.be/>

Page d'Accueil :

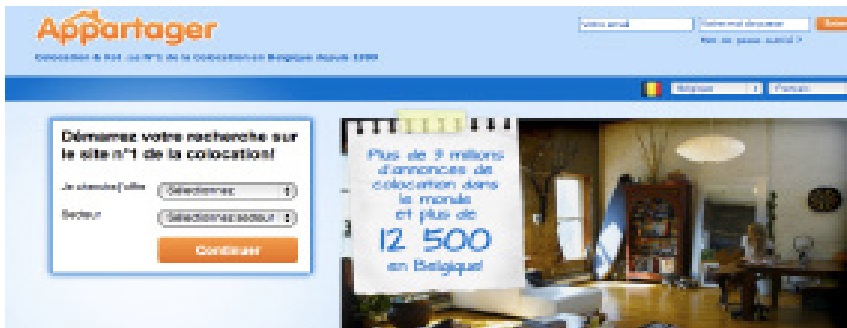
Design : les couleurs sont plutôt fades.

Sur cette page, nous devons tout d'abord opter pour la langue avant de pouvoir nous retrouver sur celle permettant les recherches.

C'est seulement après 3 à 4 clics que nous pouvons accéder aux annonces, mais étant donné l'absence d'annonce dans certaines catégories, on peut en conclure que le site n'est pas très fréquenté.

Personnellement je n'ai pas eu envie de poursuivre sur ce site et je pense qu'il en est de même pour les autres visiteurs ce qui n'est guère positif pour le site.

APPARTAGER.BE



Page d'Accueil :

Design : le design est correct, les couleurs sont vives et plutôt gaies.

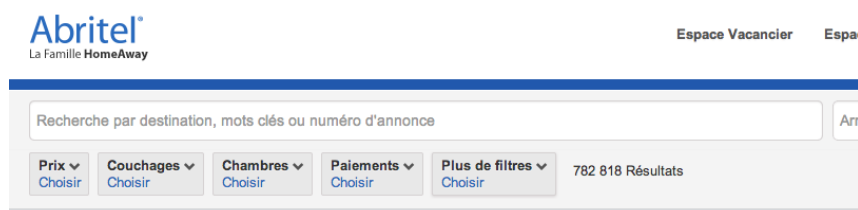
Par contre, le formulaire de recherches paraissant simple est en réalité beaucoup plus compliqué et plus long qu'il ne semble. De nombreux champs sont obligatoires afin de pouvoir obtenir une liste d'annonces. Ceci ne donne vraiment pas l'envie de faire une recherche. En principe, le formulaire de recherches est censé simplifier la vie de l'internaute mais étant donné le nombre de champs obligatoires, il la rend dans ce cas précis plus difficile, assurant l'échec de l'outil.

Isolé dans un coin de la page, le formulaire de connexion n'est pas très visible alors qu'il représente pour moi une des trois grandes priorités de la page d'accueil.

Par contre, point positif : les grandes étapes pour trouver une colocation sont mentionnées mettant au courant l'utilisateur par rapport à ce qui l'attend.

Annonces : Les annonces sont constituées majoritairement d'un bloc texte, d'une petite image permettant de voir une carte, mais cette image n'est pas très bien disposée et on passe facilement à côté de l'information.

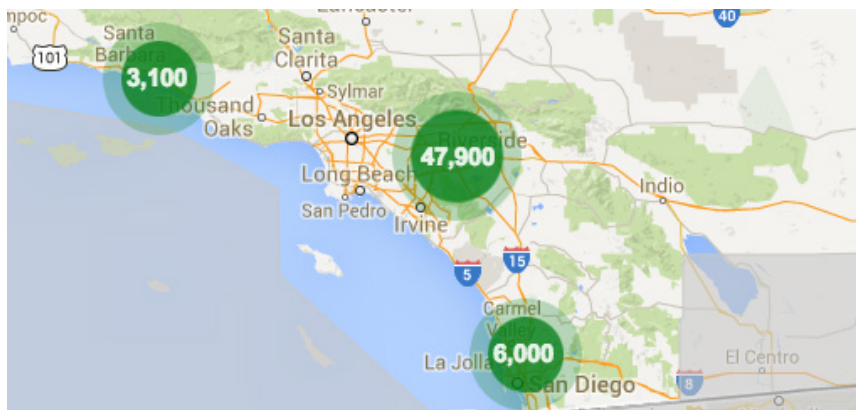
ABRITEL



Je me suis inspiré du design plutôt épuré disposant de beaucoup d'espace, de diverses fonctionnalités telles que leur système de filtre, de certaines parties du compte utilisateur et de leur expérience générale.

Etant donné mon manque de connaissance en matière de location, ce site a été ma référence n°1.

REDFIN

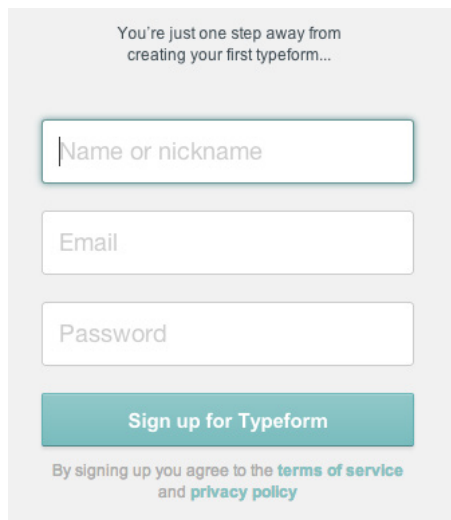


Je me suis inspiré de leur Google map et de la disposition des éléments situés sur et à côté de la carte.

Redfin a donc été ma référence en matière de Google Map.

TYPEFORM

Je me suis basé sur le style de leurs formulaires que ce soit par rapport aux espaces, à la disposition,... notamment celui d'inscription en faisant confiance à leur expérience.

A screenshot of the Typeform sign-up interface. At the top, a message reads: "You're just one step away from creating your first typeform...". Below this are three input fields: "Name or nickname", "Email", and "Password". Each field has a light blue border and a small blue icon on the left. Below the fields is a teal button labeled "Sign up for Typeform". At the bottom, a line of text states: "By signing up you agree to the [terms of service](#) and [privacy policy](#)".

CONCLUSION DE MES RECHERCHES

En conclusion, notons que la zone de recherche est l'élément primordial. Elle doit être simple ou précise ce qui complique son utilisation.

En règle générale, le design est souvent trop peu communicatif provoquant un manque d'envie de la part de l'utilisateur à visiter le site.

Le Javascript représente une partie essentielle à la création

de ce côté communicatif, attractif. L'interaction avec l'utilisateur doit être réussie. En ce qui concerne le propriétaire, il doit se sentir accompagné tout au long de l'inscription et de la gestion de ses biens.

CHAPITRE III. CRÉATION DU PROJET

CRÉATION DE LA STRUCTURE

Mon objectif principal est de positionner les éléments dans le but de déterminer des priorités dans le «scan» visuel de la page. La couleur constitue un enjeu important mais ce n'est pas pour autant la première étape.

Comme pour chaque projet, je commence par prendre mon cahier de croquis et mon crayon. Je liste les éléments importants de la page d'accueil par ordre de priorité : la carte contenant la recherche, l'inscription pour tout nouvel utilisateur, la connexion, les derniers kots ajoutés et toutes informations supplémentaires.

Pour mettre la carte en évidence, elle possède une hauteur de 80 % et une largeur de 100 % offrant une assez forte dominance visuelle.

Comme pour la plupart des sites, je crée une bannière com-

prenant les informations d'inscription / connexion, le logo, une phrase pour présenter le site et le choix de la langue.

En dessous de la carte un titre est présent dans la partie inférieure de la page et une partie du contenu visible nous indique que du contenu se trouve en-dessous.

Cette suite de contenu est disposée de manière verticale dans le but de rester dans ce style graphique et pour ne pas surcharger l'utilisateur de plusieurs informations non communes.

Sous cette carte se trouvent quatre types de contenus : la liste des derniers kots ajoutés, l'avis de personnes sur le site, des informations utiles par rapport au site (comment contribuer au développement du site, pourquoi choisir Akoter et pas un autre site ?, ...) et pour finir le footer avec des informations générales.

POURQUOI CE CHOIX DE STRUCTURE ?

Le but de la majorité des gens en visite sur le site sera la recherche d'un kot pour eux-mêmes ou pour un proche.

Il est donc primordial qu'ils ne tardent pas à la trouver. C'est pour cette raison que la recherche dispose d'une taille conséquente ne permettant pas de la manquer.

Ensuite l'inscription est mise en valeur, davantage que la connexion. En effet, une personne ayant déjà un compte sait mieux où se trouvent les éléments connexion / inscription qu'une personne n'ayant pas encore créé de compte.

La priorité à l'inscription sera seulement donnée par un fond de couleur unie.

Les derniers logements ajoutés seront mis à la suite. Certaines personnes pourraient venir visiter le site régulièrement dans le but de trouver la location parfaite. Il est donc utile de leur fournir cette information rapidement et sans devoir pratiquer une recherche.

Pour la suite, les avis des utilisateurs seront mis en avant plutôt que les informations complémentaires sur le site en général. Certaines personnes, méfiantes vis-à-vis d'Internet pourraient désirer savoir si le site est fiable. Cette information est donnée rapidement sans devoir chercher une page particulière.

Bien évidemment si quelqu'un souhaite lire les conditions et règles en vigueur, il faudra aller dans le footer (bas de page) car ces informations sont rarement consultées.

Après avoir analysé des sites de locations, il me semble important de mettre les éléments utiles en avant sans surcharger la page d'accueil comme le fait immoweb.be.

L'efficacité réside généralement dans l'organisation et la simplicité d'une page web.

Utiliser l'espace vertical permet facilement de structurer du contenu. Toutefois, il convient d'éviter de ne donner aucune indication visuelle par rapport à la présence de contenu en dessous.

La structure générale étant faite, passons ensuite à celle de chaque élément cité ci-dessus.

Pour la barre de navigation, j'ai voulu rester sobre, utiliser le strict minimum pour ne pas la surcharger. Les éléments qui la composent sont les suivants : le logo, le rôle du site, l'inscription, la connexion et le choix de la langue.

La langue est disposée en haut à droite et est l'élément le moins visible. Toutefois les visiteurs sont habitués de trouver le changement de langue à cet endroit. Ainsi il m'est possible de profiter de cette coutume pour ne pas mettre cet élément en évidence. D'autre part notons que mon site détecte automatiquement la langue à utiliser.

Le logo et introduction ainsi que l'inscription et connexion sont disposés sur le même niveau. Le logo et l'introduction occupant le côté gauche (priorité au sens de la lecture) et les autres éléments d'interaction se situant sur le côté droit, écartés du reste.

La structure de la navigation est plutôt simple et classique facilitant l'adaptation des visiteurs.

La structure de la Google Map est un peu plus complexe. Le but est de mettre en fond la carte et par-dessus les éléments d'interaction liés à cette carte.

Ces éléments sont disposés autour de son centre afin d'éviter qu'ils la masquent.

L'élément d'interaction le plus important est le système de recherche placé à gauche au vu de notre système de lecture. Cette structure comporte des éléments de formulaires qui peuvent paraître simples mais leurs différents positionnements, espaces, dimensions,... ont été longuement réfléchis et testés.

Toutefois vous trouverez davantage d'informations sur les formulaires dans le chapitre qui leur est consacré.

Le but est d'offrir des éléments d'indication d'état mais également de diriger l'utilisateur.

Ce système de recherche est déplaçable pour éviter de masquer une partie potentiellement importante de la carte.

Les autres éléments concernent des petits éléments d'interaction avec la carte. Par exemple, le bouton «voir tous les kots» sera uniquement utile après avoir modifié une distance. Si aucun kot n'a été trouvé dans la zone ciblée, ce bouton d'interaction permet à l'utilisateur d'afficher tous les kots et même ceux non compris dans la zone de sélection. Ce bouton fonctionne dans les deux sens : il affiche ou il masque.

Un mode permet de changer le style de la carte offrant un plus grand contraste, facilitant la lisibilité dans certains cas comme pour les personnes daltoniennes.

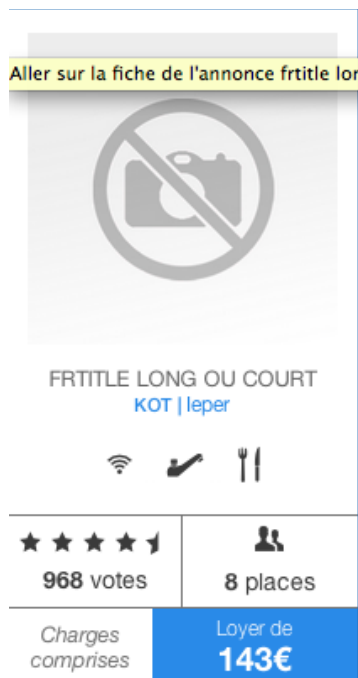
Un autre outil permet à l'internaute d'afficher la «street view» signifiant la vue de la rue. Cet outil peut être très utile

pour visualiser l'extérieur d'une location et de voir les environs. Cependant, il faut prendre en compte que les images ne sont pas toutes récentes.

La structure des dernières annonces est identique à celle des annonces dont la disposition est verticale. Cette disposition est moins courante dans les locations, la position horizontale étant souvent préférée.

Mon site se regardant par blocs disposés horizontalement, il me semble normal d'exploiter ce système. Ainsi, 4 annonces par ligne sont proposées à l'utilisateur. Il est donc possible d'en visualiser plus en faisant moins de manipulations (le fait de descendre dans la page).

L'avantage de cette structure verticale des annonces est de donner un peu plus envie de descendre dans la page.



Fiche d'une annonce fictive

La structure d'une annonce n'a pas été faite au hasard. Comme 90 % des êtres humains, notre regard sera souvent plus attiré par des images que par du texte. Ainsi les images sont disposées en haut de l'annonce et occupent une hauteur importante. C'est le premier élément qu'on devrait regarder, cette image nous conduisant à la suite de l'annonce.

La suite de l'annonce inclut le titre, le type de logement et la région. Avec ces trois informations, l'utilisateur sait si ça peut ou non lui convenir. La description n'est pas présente, le but n'étant pas de lire, je ne souhaite pas afficher du texte.

Ensuite, nous retrouvons les éléments de petite taille dont le but est de communiquer les informations importantes rapidement. Des icônes sont utilisées pour les particularités du logement (wifi, lavabo privé,...), pour la note donnée par les utilisateurs, pour le nombre de places disponibles (et sur certaines annonces «groupées» le nombre de locations encore disponibles).

La fin concerne les charges et le prix du logement par mois mis en évidence via un fond de couleur uni.

Grâce à cette structure, mon but principal a été de faire passer un maximum d'informations dans le minimum d'espace. Ceci est l'inverse de la structure générale de mon site.

Dans ces annonces se trouvent également des données cachées, utiles pour les moteurs de recherche (micro data).

La suite des éléments de la page dispose d'une grille dans le but de former des colonnes.

De manière masquée, des éléments tels que le formulaire de connexion, le choix de langues,... ont également une structure particulière de style popup.

CHAPITRE IV.

OUTILS UTILISES

PHP, MYSQL ET LARAVEL

THE PHP FRAMEWORK FOR WEB ARTISANS.

PHP THAT DOESN'T HURT. CODE HAPPY & ENJOY THE FRESH AIR.

Ce projet est réalisé en HTML / CSS / JS et programmé en PHP avec une base de données gérée en MySQL. Pour ce faire, j'ai opté pour un framework PHP nommé Laravel.

PHP est un langage de programmation dit «côté serveur», c'est-à-dire que les scripts seront exécutés par le serveur et non par le navigateur du client. Par conséquent, il est par exemple possible d'utiliser des données stockées sur une base de données pour exécuter des actions sur celles-ci.

MySQL permet de stocker des données, d'en insérer, d'en supprimer et de les catégoriser.

Laravel est une base de travail pour un projet PHP utilisant un modèle MVC c'est-à-dire une architecture de structure de projet. Laravel nous simplifie la vie dans certains cas en gérant certaines choses pouvant devenir plutôt fastidieuses. En résumé, Laravel est une amélioration du langage PHP.

D'autres framework existent tels que Symfony, Zend,... Mon choix s'est porté sur Laravel parce que nous l'avons étudié en cours et qu'il s'agit du seul framework que je connaisse.

Malgré que je ne connaisse que ce framework, il a répondu à toutes mes demandes et m'a facilité la tâche dans la majorité des cas, je compte bien continuer à l'utiliser.

Coder devient ainsi plus plaisant et l'apprentissage n'en est que plus positif.

SASS & COMPASS



SASS est un langage améliorant le langage CSS afin de styliser une page web via l'ajout de couleurs, d'images,...

SASS offre des avantages supplémentaires tels que l'utilisation de variables (enregistrer une valeur), fonctions,...

Il est également possible de programmer à l'aide de SASS ce

qui n'est pas le cas de CSS.

Compass, quant à lui, est un framework de SASS présentant des fonctionnalités intéressantes nous permettant ainsi de ne pas devoir les créer nous-mêmes. Par exemple, une fonction dont le but est de saturer ou de «désaturer» une couleur.

Le souci : il faut compiler le langage SASS en CSS pour qu'il soit compris du navigateur. Pour cela, on peut utiliser le programme Codekit expliqué ci-dessous.

CODEKIT



Lors du commencement du projet je me suis servi du programme Codekit. Codekit permet de réaliser des tâches sur les langages tels que le CSS, le SASS, le Javascript,...

Par exemple si nous enregistrons un fichier Javascript, Codekit le compresse afin de réduire sa taille. A l'aide de ce programme il est également possible de combiner plusieurs fichiers de ce même type en un seul sans besoin d'une intervention de notre part.

```

@mixin at2x($path, $ext: "jpg", $w: auto, $h: auto) {
  $at1x_path: "#{$path}.#{ $ext}";
  $at2x_path: "#{$path}@2x.#{ $ext}";

  background-image: image-url("#{ $at1x_path}");

  @media all and (-webkit-min-device-pixel-ratio : 1.5),
  all and (-o-min-device-pixel-ratio: 3/2),
  all and (min--moz-device-pixel-ratio: 1.5),
  all and (min-device-pixel-ratio: 1.5) {
    background-image: image-url("#{ $at2x_path}");
    background-size: $w $h;
  }
}

```

Ma principale utilisation est liée à SASS et Compass. Lors de précédents projets, j'avais créé des mixins (des morceaux de codes à coller en une ligne, exemple ci-dessus) avec des éléments utilisés par défaut tels que les différentes tailles de police, une grille, des helpers (aides) et d'autres éléments de cet ordre.

Toutefois, le défaut de ce langage SASS est qu'il doit être compilé (converti) en fichier CSS et c'est à ce moment que Codekit entre en jeu.

Notons cependant que j'ai décidé de ne plus utiliser ce logiciel Codekit afin de me servir de Gulp, plus rapide lors de la compilation des fichiers SASS en CSS.

GULP



Gulp est basé sur NODEJS qui est du Javascript dit «côté serveur» et non «côté client».

Via un simple fichier de configuration (<https://github.com/julienroland/akoter/blob/master/public/gulpfile.js>), Gulp exécute des tâches en tout genre et, dans mon cas, il me sert à remplacer Codekit et me compile mon Javascript, mes fichiers SASS et est capable de réaliser d'autres tâches supplémentaires telles que concaténer des fichiers entre eux.

Pourquoi avoir préféré Gulp ? Le temps de compilation du fichier SASS est de 4 à 5 fois plus rapide avec Gulp motivant ainsi le changement.

De plus, cet outil dispose d'une communauté lui créant des outils, renforçant son utilité grâce à la diversité de ses possibilités.

GITHUB



GitHub est un service en ligne réalisant diverses tâches mais, dans mon cas, il est utilisé pour l'enregistrement de mon projet sur le serveur afin d'avoir une sauvegarde distante de celui-ci (<https://github.com/julienroland/akoter>).

La sauvegarde sur GitHub est assez aisée puisqu'il se sert de la ligne de commande. Une fois le projet installé, il suffit d'écrire la commande «git push» pour que le service lance automatiquement l'envoi de fichiers sur le serveur sans nécessiter notre intervention.

GitHub m'est également utile pour la gestion de versions et pour l'historique des modifications apportées à mon projet.

GOOGLE MAP

Le service Google Map API permet l'affichage d'une carte interactive de mes logements et permet d'interagir avec les logements et la carte.

Ce service me donne l'opportunité d'offrir un système de recherche avancé et plus poussé que ceux de mes «concurrents».

Ainsi, les utilisateurs peuvent précisément rechercher à partir d'une position et d'une distance.

Google Map API présente également de grande polyvalence qui pourra être davantage exploitée dans le futur (application mobile avec GPS via la carte afin de visiter un logement,...)

Mon choix s'est porté sur Google Map parce que je l'ai découvert durant les cours de RIA (Rich Internet Application). Etant donné qu'il satisfaisait à mes besoins et n'ayant rencontré aucun soucis, je n'ai pas recherché une autre alternative. Et ce malgré que le site populaire Foursquare soit passé sur un autre service de carte semblant plus rapide mais disposant peut-être de moins de possibilités.

SUBLIME TEXT

Éditeur de texte découvert lors des cours de web en 2ème année, il permet, via des raccourcis clavier, de coder plus rapidement.

Cet éditeur me semble agréable tant pour un designer que

pour un développeur. Cependant il est moins optimisé pour le développement : les IDE sont généralement plus performants dans ce domaine.

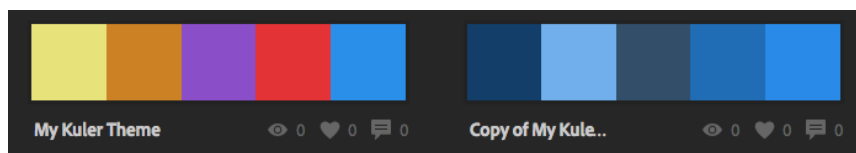
J'ai également essayé, à plusieurs reprises, PHPStorm mais à chaque tentative des détails au niveau de la configuration m'ont bloqué. N'ayant guère de temps je suis à chaque fois retourné vers Sublime Text.

D'autres éditeurs ont également été testés tels que Bracket, conçu uniquement pour les designers, les changements se faisant en temps réel.

Après ma session d'examens j'ai pour objectif d'essayer à nouveau d'utiliser avec plus d'ardeur Atom et surtout PHPS-torm.

OUTILS MINEURS

D'autres outils ont également été employés afin d'améliorer la qualité générale du site dans divers domaines : optimisation d'images pour les écrans rétina, Adobe Kuler pour sa palette de couleurs, Typs pour l'échelle des polices,...



Scale table

Pixels	EMs	Percents
5.7px	0.382em	38.2%
7.3px	0.486em	48.6%
9.3px	0.618em	61.8%
11.8px	0.786em	78.6%
15px	1em	100%
19.1px	1.272em	127.2%
24.3px	1.618em	161.8%
30.9px	2.058em	205.8%
39.3px	2.618em	261.8%
50px	3.33em	333%
63.5px	4.236em	423.6%
80.8px	5.388em	538.8%

CHAPITRE V.

ÉTAPES DE CRÉATION

WIREFRAME

Wireframe est un schéma définissant les zones d'un site web ou d'une page web.

The wireframe shows a web browser window with the title 'AKOT-E'. The browser's address bar is empty. The page has a header with a logo placeholder, the text 'FR | NL | EN', and links for 'Login' and 'Inscription'. Below the header is a breadcrumb trail: 'Accueil > Inscription > propriétaire'. A progress bar with five dots is shown, with the fourth dot highlighted. The main content area is titled 'Informations personnelles' and contains a form with the following fields: 'Nom' (Roland), 'Prenom' (Julien), 'Civilité' (Mr.), 'Date naissance' (28/02/92), 'Localisation' (Wepion), 'Adresse' (Rue Basse-Montagne), 'n°' (20), 'Tel.' (08146 24 12), 'GSM' (0498/00 00 00), 'Email' (moi@moi.com), and 'Mot de passe' (*****). There is a 'Langue' dropdown menu set to 'Français'. Below the form is a checkbox labeled 'Acceptez-vous que des locataires parlent une autre langue ?' with a 'CN' button. At the bottom of the form is a button labeled 'Etape 2' and a link 'ne pas inscrire une propriété maintenant'. The footer contains links for 'Qui sommes-nous?', 'Mentions légales', 'Contactez-nous', and an email icon.

AKOT-E

FR | NL | EN Login InSCRIPTION

Accueil > Inscription > propriétaire

Informations personnelles

Nom: Roland Prenom: Julien Civilité: Mr.

Date naissance: 28/02/92 Localisation: Wepion

Adresse: Rue Basse-Montagne n°: 20

Tel.: 08146 24 12 GSM: 0498/00 00 00

Email: moi@moi.com Mot de passe: *****

Langue: Français

Acceptez-vous que des locataires parlent une autre langue ?

Langue(s): Anglais

Etape 2 ne pas inscrire une propriété maintenant

Qui sommes-nous? Contactez-nous

Mentions légales

Mon wireframe a été créé à l'aide du programme Balsamiq mockup me permettant de définir la ligne directrice de mon site.

Le wireframe comprend le schéma de la page d'accueil, les différentes pages de recherche, ce qui concerne l'inscription propriétaire / locataire et certaines pages du compte utilisateur.

Grâce à ces différents points, mes objectifs finaux ont pu être définis : la recherche de logements, l'inscription d'un propriétaire / locataire, l'inscription du bâtiment et la relation entre propriétaire et locataire.

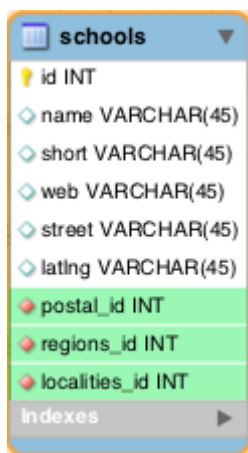
Dans ce mockup, il n'a pas été question de réaliser du design ni la structure des pages. En effet, il a pour but de définir les lignes directrices du projet, les objectifs.

Veuillez retrouver celui-ci dans les annexes en pièces jointes.

SCHEMA DE LA BASE DE DONNEES

La création du schéma de la base de données a été accomplie à l'aide du programme MySQL WorkBench.

Le but de ce schéma est de pouvoir visualiser la structure de la base de données ainsi que chaque table qu'elle contient.



Par exemple, dans notre cas, on affiche la table schools, on peut apercevoir que cette table contient un «ID», un «name», un «web», ...

Le programme permet de créer les relations entre les tables. Par exemple, la table users peut avoir plusieurs buildings et une table building appartient à un user.

Lors de mes premiers essais, plusieurs «problèmes» sont survenus et m'ont été reprochés tels que nommer ma base de données en français alors qu'il faudrait la nommer en anglais. En effet, si un jour quelqu'un reprend ce projet, il faut qu'il soit capable de comprendre même s'il n'est pas francophone.

Le second problème est en rapport avec le nom de mes tables : leurs noms étaient trop précis et manquaient donc de cohérence dans certains cas.

Après une réunion avec mon professeur superviseur nous avons apporté des modifications sur les tables et avons donc corrigé les «problèmes».

A la création de la base de données, les conventions Laravel n'avaient pas été correctement respectées. Il m'a fallu prendre la décision d'apporter les modifications nécessaires afin de respecter les conventions.

Concernant les traductions, penser au système à utiliser était nécessaire. Toutefois, n'ayant aucune connaissance sur ce sujet, je n'ai pu qu'émettre des hypothèses sans pour autant connaître la meilleure technique de gestion des traductions.

Pour ce faire, poser la question sur le Google Group de Monsieur Vilain m'a semblé être la meilleure option afin d'avoir une idée de la meilleure technique. Plusieurs personnes m'ont répondu et m'ont conseillé différentes solutions avec leurs avis sur chacune d'entre elles. Finalement, j'ai pris la décision d'opter pour une méthode qui me semblait être la meilleure.

L'objectif est d'avoir une seule table de traduction rassemblant toutes les traductions du site afin d'éviter d'avoir pour chaque table une deuxième table de traduction et par conséquent doubler le nombre de tables uniquement à cause des traductions. Ceci représente un énorme gain de temps pour la conception de la base de données.

La table fonctionne avec un format de ce type :



The screenshot shows a database table named 'translations'. It has six columns: 'id' (INT, primary key), 'content_type' (VARCHAR(45)), 'content_id' (VARCHAR(45)), 'key' (VARCHAR(45)), 'value' (VARCHAR(45)), and 'languages_id' (INT). Below the columns is an 'Indexes' section with a right-pointing arrow.

translations
id INT
content_type VARCHAR(45)
content_id VARCHAR(45)
key VARCHAR(45)
value VARCHAR(45)
languages_id INT
Indexes

content_type, content_id, key, value, languages_id.

Cette méthode utilise des colonnes génériques permettant d'avoir une table plus souple et, dans notre cas, de recevoir les données de plusieurs tables.

Ce système peut être employé dans d'autres situations telle que la suivante : mon site de location dispose d'options en tout genre : outils de cuisines, literie, objets extérieurs. Avec le système proposé, il est possible de créer la table comme ceci : id, clé, valeur, type_option. Les valeurs seraient les suivantes: 1, lit double, 2, literie, 2, télévision, 2, objet intérieur.

Je tiens à remercier Hugues Lismonde pour m'avoir fourni ces informations et également les autres personnes ayant répondu à ma demande.

Tout au long de l'avancement du projet il m'a fallu apporter des modifications à cette base de données en ajoutant des in-

formations ou en retirant d'autres.

MISE EN PLACE DE LARAVEL

Une fois le schéma terminé, j'ai procédé à l'installation de Laravel dans mon projet via composer et l'ai configuré afin qu'il corresponde avec ma base de données. Tout au long de la conception du projet, des modifications y ont été apportées telles que le système de mail et les traductions.

J'ai également dû procéder à des mises à jour de Laravel et d'autres packages.

CREATION DE MIGRATIONS ET DES SEEDS

Laravel nous propose un système permettant de créer une base de données via des fichiers PHP et facilite ainsi la mise en place de notre base de données sans devoir toucher à MySQL.

Les migrations sur Laravel offrent l'avantage de pouvoir rapidement supprimer / recréer nos tables mais également de partager facilement sa structure SQL avec d'autres personnes. Etant seul sur ce projet, on pourrait croire qu'il n'y a pas de réel intérêt à utiliser des migrations plutôt que du SQL. Toutefois, mon projet étant en PHP et Laravel offrant la possibilité de stocker sa structure de manière textuelle, pourquoi ne pas le faire surtout qu'il entre dans mon workflow contrairement au SQL.

CHAPITRE VI.

DESIGN

De plus, comme les migrations sont intégrées à Laravel, lorsque je réalise une sauvegarde de mon projet sur Github, celles-ci sont également enregistrées. Dans le cas où on n'utiliserait pas de migrations, il faudrait enregistrer la base de données à chaque fois pour ensuite pouvoir l'ajouter à notre projet Laravel pour réaliser une sauvegarde, occasionnant ainsi une étape supplémentaire.

Notons toutefois que l'ordre des migrations est important par rapport aux relations qu'elles ont entre elles. Par exemple, une table «users» contient des relations avec la table «messages», «regions», ... Il est impossible de créer la table «users» si les tables «messages» et «regions» n'ont pas été préalablement créées.

J'ai également créé des fichiers de «seeds» dont le but est d'envoyer des données fictives ou non à la base de données. Dans le cadre de mon projet, cet outil a été utilisé au début

mais m'a également servi à ajouter des éléments permanents tels que les localités, les codes postaux et les régions.

Comme pour les localités, j'ai utilisé un fichier JSON comportant les données nécessaires. Ainsi, des scripts ont été réalisés pour générer un certain nombre de kots fictifs ainsi que des écoles et divers éléments utiles au site.

```
{
  "id": 1728,
  "postal": "6120",
  "name": "Nalines"
},
{
  "id": 1729,
  "postal": "5300",
  "name": "Namêche"
},
{
  "id": 1730,
  "postal": "5000",
  "name": "Namur"
},
{
  "id": 1731,
  "postal": "4550",
  "name": "Nandrin"
},
}
```


CHAPITRE VI.

DESIGN

Pour ne pas commettre les mêmes erreurs que sur certains sites précédemment analysés, il m'a semblé primordial de définir les points prioritaires à visualiser par les utilisateurs.

Ainsi, il m'a semblé utile de donner à ces points prioritaires une certaine importance au niveau de la taille, des couleurs et de la position.

DESIGN DU LOGO



Avant d'être parvenu au résultat actuel, plusieurs essais fort médiocres ont été effectués.

Dans des moments de doutes, il me semblait qu'aucune inspiration ne viendrait à moi et qu'il me serait impossible de réaliser un logo.

Toutefois, l'inspiration m'est venue et le logo m'a alors semblé évident : deux lettres sous forme de perspectives illustrant deux bâtiments proches l'un de l'autre. Ces lettres avaient pour but de définir des logements, la perspective évoquant le jeu de mot de la recherche à proximité d'une école ou d'un endroit déterminé. On y retrouve également le mot «kot» signifiant «logement étudiant».

DESIGN DE LA CARTE

L'élément principal de la page étant la carte, lui accorder la plus grande importance me semble évident. Ainsi elle représente environ 80 % de l'écran.

Tous les éléments d'interaction et les formulaires de recherche sont disposés par-dessus la carte.

Au niveau du bouton «rechercher» et de la vue d'une location, j'ai décidé de les disposer à droite de la carte.

Ainsi, la largeur de la carte est de 100 % - la largeur de ce «bloc».



DESIGN DU «BLOC» INSCRIPTION

Le deuxième élément principal de la page est l'inscription placée dans la barre de navigation et possédant approximativement la même dimension que le bouton connexion. Afin de parvenir à les différencier, le bloc inscription dispose d'un fond légèrement orangé contrairement au bloc connexion dont le fond est noir.

DESIGN DES DERNIÈRES LOCATIONS AJOUTÉES

Le dernier élément à mettre en évidence est la vue des dernières locations ajoutées, situé en-dessous de la carte. Le titre apparaît sur la première page ce qui permet de comprendre que du contenu se trouve en dessous de la carte et qu'il concerne les dernières locations.

Le fond de la liste des locations est de couleur bleu clair atti-

rant ainsi le regard de l'internaute.

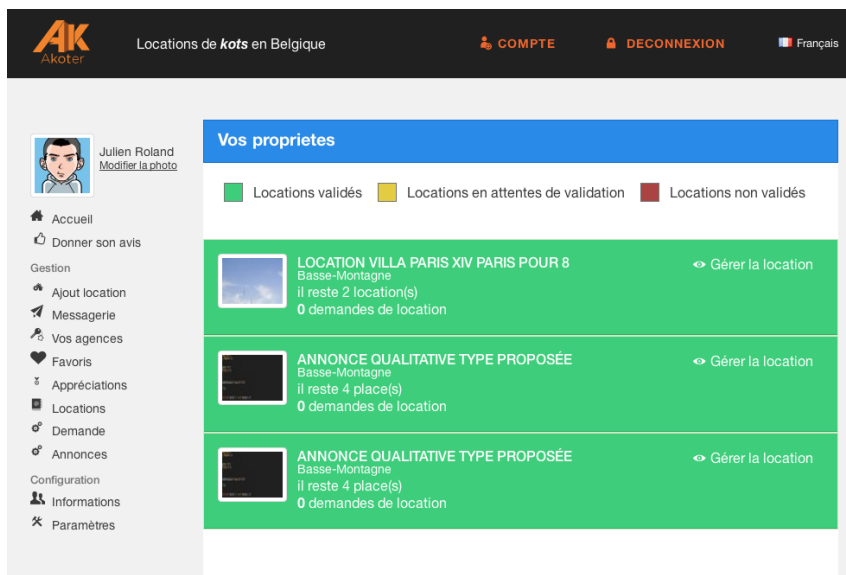
CRÉATION DU LAYOUT

La mise en page est basée sur le visuel de la page d'accueil et sera présente partout avec la navigation au-dessus le footer dans le bas et le contenu au centre disposé majoritairement en lignes.

Plusieurs styles différents de layout sont présents sur le site comme pour les pages du compte utilisateurs, les pages de l'administration et de l'API, ...

DESIGN DU COMPTE UTILISATEUR

Un nouveau style de mise en page employé est celui du compte utilisateur. Pour cela, j'ai décidé d'utiliser le layout classique mais en y ajoutant une mise en page supplémentaire à l'endroit où se trouve le contenu. Cette fois, le contenu est disposé en deux colonnes : une à gauche (étroite) contenant la navigation au sein du compte utilisateur et une colonne occupant la majorité de la largeur comprenant le contenu du compte utilisateur.



DESIGN DES POP UP

Des «pop-up» ont été créés pour certains cas et comportent également une mise en page qui leur est propre.

Ceux-ci sont disposés dans un fond blanc, de manière «absolue» et non fixe par rapport à la page.

L'avantage ? Pour les écrans de petite hauteur, les pop-up fixes ne permettent pas de voir tout le contenu, une partie restant masquée.

La mise en page des pop-up comprend un bouton «fermer» avec le même style pour chaque pop-up.

Les pop-up ont des largeurs variables en fonction du contenu et sont accessibles via le clavier.

Connectez-vous
Louez vos logements étudiants ou
trouver le logement étudiant de vos
rêves !

 Connexion avec Facebook

ou

Votre email
roland.julien.perso@gmail.com @

Votre mot de passe
..... 🔒

☒ Se souvenir de moi

Se connecter

Pas encore de compte? [Inscrivez-vous!](#)

DESIGN D'UNE FICHE D'ANNONCE

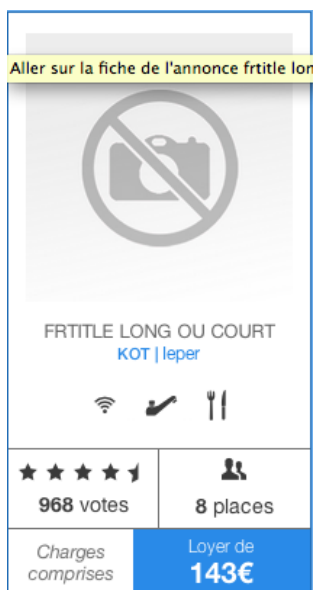
Contrairement à la plupart des styles d'annonce disposés horizontalement, j'ai choisi une méthode verticale.

Selon moi, le défaut des annonces horizontales est l'affichage d'une seule annonce par ligne. Si nous disposons d'un écran à hauteur réduite, on pourra afficher moins de lignes donc moins d'annonces.

En premier lieu, l'image de l'annonce occupe 30 % de la hauteur et 100 % de la largeur.

Ensuite, le titre de l'annonce disposé sur une ligne et, pour finir, l'affichage du type de location et de la localité de la couleur principale du site, le bleu.

Les éléments suivants constituent les particularités d'un logement : le wifi, la salle-de-bain privée,... et sont affichés sous forme d'icônes.



La ligne se trouvant en dessous des particularités est scindée en deux parts égales : la première contient la note de l'annonce sous forme d'icônes et le nombre de votants tandis que

la deuxième correspond au nombre de places disponibles.

La dernière ligne de l’annonce est également divisée en deux parties non égales : les charges à gauche et le loyer à droite dans un fond bleu pour le mettre en valeur.

DESIGN DES PAGES ADMINISTRATEUR ET API

Ces pages utilisent Bootstrap et disposent d’un menu positionné sur le dessus de chaque page et, ensuite, le contenu occupe le reste de la hauteur.

Il n’y a pas de footer étant donné que ces parties du site sont réservées à un public aguerri et ne requièrent pas d’informations supplémentaires.

Admin - Julien

Etats

Annouces

Articles

Utilisateurs

Traductions

Avs

Deconnecter

Quitter l'admin

Filtrer

Référence	Nom	Email	Rôle	Langue	Email-Confirm	Validé	Propriétaire	Supprimé	Dernière connexion	Crée le	Mis à jour le	Actions
1	Julien Roland	Roland-Julien@email.com	superadmin	Français	✓	✓	✗	✗	Il y a 2189 years	24 12:00:00-12-2014	11-0001-	<div><div>✓</div><div>✕</div><div>✎</div><div>✖</div></div>

CHAPITRE VII.

L'INTERFACE

LES COULEURS

La première version du site était fort épurée, la couleur dominante étant le blanc. Des couleurs assez joyeuses telles que le rouge (entre pâle et clair) ainsi que le bleu (entre pâle et clair), étaient les couleurs principales afin de faire ressortir certains éléments. Toutefois, certaines personnes ont trouvé que mon design manquait de caractère.

Ces remarques m'ont fait réfléchir et c'est pourquoi j'ai légèrement adapté le design en fonction de ces remarques, notamment la palette de couleurs.

Pour cela, j'ai opté pour l'outil Adobe Kuler. Premièrement, j'ai ciblé une couleur principale dominante qui, contrairement au design d'avant était le bleu et est devenue l'orange. Le choix de cette couleur s'explique par le fait qu'un certain type d'orange permet une visibilité correcte avec une écriture blanche ou noire, ce qui est plutôt rare dans les couleurs.

De plus, son harmonie avec le noir offre un impact visuel assez fort et dominant. A l'aide d'Adobe Kuler, j'ai cherché les couleurs à associer à l'orange et une est ressortie du lot : le rouge.

En fonction de ce rouge, j'ai à nouveau cherché une palette étant donné que je ne souhaitais pas que l'orange soit la couleur principale de tout le site. Trouver une deuxième couleur principale était donc important et celle-ci s'est révélée être un bleu dans les tons clairs.



Avec ces résultats, j'ai deux palettes : une avec un orange et du noir et une avec du bleu et du rouge.

Les couleurs orange et noire ont été principalement utilisées au niveau de la navigation.

Quant au rouge, il sera mis en alliance avec le bleu pour les autres éléments. Le bleu deviendra également la couleur principale pour les éléments typographiques à mettre en évidence tels que les liens.

Par la même occasion, j'ai défini la couleur principale des éléments typographiques : du noir atténué et mes nuances de gris ont également

été déterminées (du très clair au foncé).

Ensuite ma tâche a été de fixer les couleurs pour des éléments souvent utilisés tels que les messages d'erreurs, de réussite ou d'informations.

Toutes ces couleurs sont enregistrées dans des variables SASS permettant de les retrouver facilement dans mes feuilles de style.

LA TYPOGRAPHIE

Pour la typographie, j'ai opté pour l'outil Typs permettant de créer une échelle typographique selon certains ratios tels que le nombre d'or et avec des «paliers» entre ces valeurs.

La police choisie est Helvetica Neue, adaptée de par sa simplicité de forme, de la bonne visibilité qu'elle offre et du nombre de variante dont elle dispose.

Son style «gras» est un point positif car il permet de donner un bon impact aux titres.

Au vu de la complexité du site, il m'a semblé préférable d'opter pour une police plutôt «classique» et populaire, sans prendre trop de risques.

LES ICÔNES

Dans mes précédents projets, des librairies d'icônes pré faites étaient utilisées, ma préférence allant la plupart du temps à Font-awesome.

Les librairies de police ont l'avantage d'être simples à obtenir, à utiliser et représentant un gain de temps important. Cependant nous sommes limités aux icônes qu'ils proposent étant donné que nous sommes dans l'impossibilité d'en ajouter de nouvelles.

Pour mon projet de fin d'études, des icônes plus particulières étaient nécessaires mais celles-ci n'étaient pas disponibles dans cette librairie.

Afin de ne pas utiliser plusieurs librairies d'icônes, dans le but d'éviter de réaliser trop de chargements pour optimiser les performances, il m'a fallu opter pour une méthode de création de pack via le site flaticon.com.

L'avantage de ce choix est que toutes les icônes comprises dans ce pack sont toutes utilisées pour mon site web, contrairement aux librairies offrant des icônes pour la gestion des sons ainsi que d'autres éléments inutiles dans la majorité des cas. Notons que ce pack peut être agrandi et rétréci à volonté.

Les icônes employées ont été converties en icônes de type webfont dont l'avantage est de pouvoir les utiliser comme une police d'écriture et elles sont redimensionnées de façon vectorielle.

Son désavantage est en rapport au SVG dont la qualité est supérieure. Toutefois ce dernier n'est pas supporté par tous les navigateurs. Il faudra attendre la «fin» de Windows XP pour pouvoir enfin utiliser les icônes SVG.



.flaticon-address14
Author: [Freepik](#)



.flaticon-approve
Author: [Freepik](#)



.flaticon-arroba
Author: [Freepik](#)



.flaticon-arrow
Author: [Freepik](#)



.flaticon-arrow198
Author: [Freepik](#)



.flaticon-arrow259
Author: [Freepik](#)

LES IMAGES

La plupart des images statiques du site ont pour but l'illustration du contenu et elles ont été trouvées pour la majorité sur le site unsplash.com (libre de droit).

Celles-ci, comme toutes les autres images du site, sont enregistrées dans un certain format avec des dimensions spécifiques dont le but sera d'avoir deux types de tailles d'image : un classique pour tous les écrans et un type pour les écrans rétina dont la taille des images est deux fois plus grande.

Ces images rétina ont le même nom que les images classiques mais avant l'extension, il faut ajouter «@2x». En fonction du type d'écran, ceci permettra de retrouver la bonne image.

Pour ce faire, deux techniques sont utilisées : la première en SASS correspond à la création d'une mixin vérifiant le ratio de l'écran et, en fonction du ratio, charge la bonne image. La deuxième utilisée pour les images intégrées au HTML, un script Javascript va vérifier si l'image rétina est également

présente (à l'aide du @2x). Dans l'affirmative, l'image rétina sera chargée à la place de l'image de base.

Concernant les images enregistrées via le site telles que celles d'une location ou les photos de profil, elles sont enregistrées dans un dossier «images».

Ensuite, un dossier est créé avec l'ID de l'utilisateur dans lequel on peut retrouver un dossier image de profil, location et agences. Dans les dossiers locations et agences, nous y retrouverons l'ID de la location ou de l'agence et ensuite il est possible de catégoriser à nouveau en fonction du logo et de l'image du profil pour finalement y ajouter les images nommées avec le «timestamp» et l'extension.

Exemple de chemin de dossier : *«images/41/agences/1/logo/123456789.jpg»*

Cela permet d'avoir une bonne structure des fichiers, de pouvoir facilement les retrouver sur le site et de pouvoir faciliter la suppression et, dans certains cas (ex : suppression d'une agence) nous pouvons supprimer tout le dossier sans devoir supprimer image par image.

L'avantage de nommer les images avec un «timestamp» est qu'il est unique.

Chaque enregistrement d'image de locations sur le site s'enregistre en 6 formats différents : format de base mais est redimensionné et compressé pour éviter de prendre trop de place, un grand format, trois formats moyens et un petit for-

mat, chacun employé dans différentes situations.

ADAPTABILITÉ DU SITE

Par adaptabilité, nous devons comprendre la disponibilité du site pour tout le monde que ce soit via un téléphone ou un ordinateur.

Concevoir l'adaptabilité du site nécessite énormément de temps étant donné la complexité de certaines pages comme la page d'accueil à cause de sa carte. Dans le cas de mon site, il ne s'agit plus d'une simple adaptabilité mais de repenser la structure des éléments pour l'adapter à chaque taille d'écran.

Malheureusement, à l'heure actuelle, il ne m'a pas encore été possible de m'attarder sur ce point mais cela sera fait lorsque le reste du site sera fini.

Toutefois la création de mon responsive est prévue via des calculs en «EM». Un «EM» étant égal à la taille de la police définie.

Quels sont les avantages des «EM» au niveau du responsive ? En règle générale, lorsque nous travaillons dans le responsive design, nous mesurons la largeur de l'écran et, en fonction de cette mesure, les éléments sont parfois repositionnés.

Lors du zoom avec certains navigateurs, ils ne redimensionnent pas la taille de la page ce qui a pour conséquence de fournir une donnée fausse. Ainsi, le responsive ne fonctionnera pas correctement, n'adaptant pas le site de manière

optimale.

Avec la majorité des navigateurs, la police d'écriture s'adapte lors du zoom. De ce fait, notre calcul via les «EM» fonctionne correctement.

LES FORMULAIRES

Actuellement Akoter dispose de deux types de formulaires : un dont les labels sont disposés au-dessus des champs et, sur l'autre type, les labels sont alignés sur la même ligne que les champs.

Chacun de ces styles est créé via une mixin mais ils disposent tous les deux de caractéristiques globales aux formulaires.

Le style : le fond de couleur du champ est entre le gris et le jaune très pâle avec une bordure gris pâle permettant de comprendre qu'il n'est pas en action . Par contre, lorsqu'il est en action, le champ est de couleur blanche avec une bordure grise plus foncée munie d'une légère ombre de portée de couleur grise. A ce moment, on comprend bien qu'on est actif sur ce champ.

Rue *

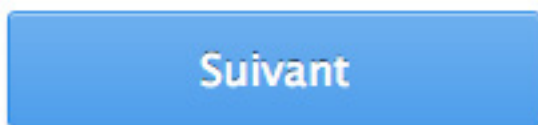
Le champ Adresse est obligatoire.

Code postal *

La plupart des champs disposent d'un «placeholder» gris clair. Un placeholder est un texte indiquant des informations sur le champ et, une fois le premier caractère inscrit dans le champ concerné, le texte disparaît pour laisser place au contenu écrit.

Au niveau des boutons d'envoi, ceux-ci disposent d'un dégradé dont la couleur correspond à la couleur principale du site.

Le but est de donner plus d'impact grâce aux couleurs vives et aux dégradés offrant une forme arrondie donnant du volume.



Les listes déroulantes disposent de deux styles particuliers : le style par défaut du navigateur offrant une liste classique alors que le deuxième style, «chosen», remplace le style par défaut en offrant des avantages. Ainsi nous avons droit à un style personnalisable via CSS intégrant une recherche interne utile lors d'une sélection comportant de nombreux éléments, le tout accessible aux claviers.

Chose offre également d'autres avantages pour les «select multiple». Les styles liés aux champs valides et invalides ont tous deux les mêmes styles mais les couleurs ont été adaptées : d'un côté un rouge «bordeau» et de l'autre un vert clair.

Cependant, lorsqu'une erreur est présente après l'envoi du formulaire et la vérification de celui-ci, les champs erronés sont entourés d'un contour rouge pour indiquer où se situent les erreurs. Toutefois, notons qu'un message d'erreur global s'affiche aussi pour nous signaler nos erreurs.

Le champ Région est obligatoire.
Le champ Localité est obligatoire.

Région *

Région ▼

Localité *

localité ▼

Lors de l'envoi du formulaire, s'il contient toujours des er-

reurs un groupe de message s'affiche nous permettant de comprendre les erreurs.

Savoir si ce qu'on indique dans le formulaire est correct en temps réel est très important. Ne pas devoir recharger des pages en permanence représente un énorme gain de temps et diminue les risques de perdre des internautes.

Par exemple, concernant la vérification du mot de passe, il arrive que certaines personnes fassent des erreurs lorsqu'elles doivent taper deux fois le mot de passe. Si le système ne nous donne pas d'informations quant à l'exactitude du mot de passe, les deux devant être identiques, il faut à chaque fois recharger la page ce qui n'est pas le cas avec mon système.

Actuellement, mon système n'est pas le meilleur par manque de temps. Il ne m'a pas été possible de refactoriser toutes les fonctions de validation écrites en PHP vers Javascript. Une requête Ajax me permet donc de réaliser la validation en temps réel.

L'ACCESSIBILITÉ

Au niveau du web, l'accessibilité est importante car la visite des personnes malvoyantes, voire aveugles sur des sites non optimisés est extrêmement compliquée et même impossible dans certains cas, raison pour laquelle cette thématique doit être abordée lors de la création d'un site web. Toutefois, notons que ce n'est pas toujours facile à réaliser notamment avec l'utilisation de cartes interactives.

Pour cela, je me suis basé de la checklist d'AnySurfer et de celle du Pompage.

Chaque page du site dispose d'une hiérarchie des titres réfléchi (utile également dans le point suivant: SEO). Elle est compatible dans deux types de cas particuliers (HTML4 et HTML5) et aide les utilisateurs malvoyants en indiquant où ils se situent sur le site.

Des liens sont également cachés visuellement dont le but est d'aller directement au contenu en évitant certaines parties du site.

Tout au long du projet, il m'a fallu insérer des balises de type ARIA sur des titres, sur des éléments invisibles pour les personnes malvoyantes et sur les éléments de type popup,... Indiquer les «rôles» est également important.

Des «titles» ont également été ajoutés sur des liens contenant un texte peu explicite. Par exemple, un lien suite pourrait contenir un titre informant le passage sur la page suivante de l'inscription, celle-ci étant la description du bâtiment.

Il est également important d'éviter de faire l'erreur inverse, c'est-à-dire d'ajouter des titres sur des liens déjà explicites. Par exemple, un lien «aller à l'accueil» n'a pas besoin d'une information semblable pour éviter les répétitions.

Par contre, il m'a fallu employer des «tooltip» lorsqu'il était utile de donner davantage d'informations, tooltip visible uniquement au hover en remplaçant la balise titre. Ainsi l'ac-

cessibilité n'est pas dérangée.

L'accessibilité sur les formulaires constitue également un point important. Lors de la création de ces formulaires, certaines pratiques m'étaient inconnues telles qu'indiquer dans le label si le champ est requis. Il m'a donc été nécessaire de repasser sur tous mes formulaires pour corriger les erreurs et en améliorer l'accessibilité. Notons que l'accessibilité via le clavier est également important et il m'a également fallu vérifier ce point.

En conclusion, soulignons l'importance de l'accessibilité. Toutefois, elle doit être optimisée à la fin du projet mais n'ayant pas disposé de suffisamment de temps pour peaufiner ce point, il se peut qu'il y ait encore des manquements.

SEO

SEO est l'abréviation de Search Engine Optimize pouvant être traduit par optimisation des moteurs de recherche. Cette étape est tellement importante que des personnes sont spécialisées là-dedans.

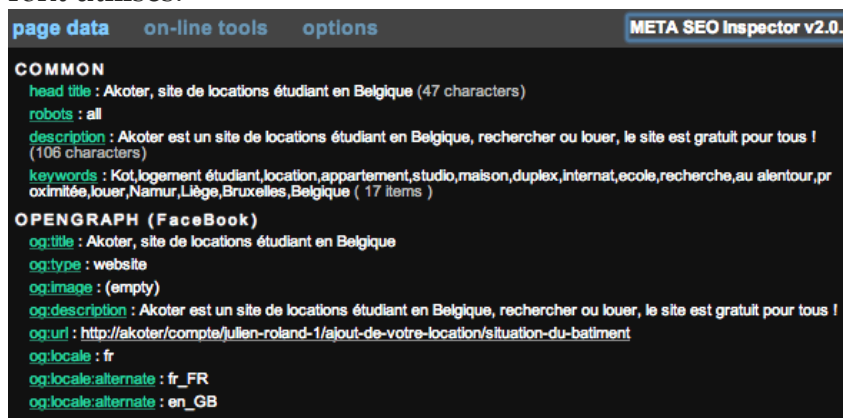
Cela consiste à placer son site en premier lors de recherches sur le site en question. Plus haut on est placé, plus on est visibles et on augmente ainsi le potentiel de visiteurs.

Cette thématique est également vaste et nécessite notre attention pour beaucoup de points. Ci-dessous, vous trouverez la liste non-exhaustive des éléments pris en compte pour mon site web :

- URL propre. Éviter les URL du type : user / 02132.php sur un site en français et préférer les URL tels que utilisateur / kot-liège-02132, augmentant ainsi l'impact).
- Être présent sur les réseaux sociaux tels que Google, Facebook, Foursquare,...
- Tenter de récolter un maximum de «j'aime», «partages» et autres actions de cet ordre liées aux réseaux sociaux.
- Avoir un titre, une description pour chacune de ses pages donnant des informations utiles à la page en question. Éventuellement des mots clés mais ceux-ci possèdent un faible impact.
- Disposer de pages de contenu avec un minimum de texte.
- Posséder un site vivant avec de nouvelles pages,...
- Être pourvu de liens vers notre site via d'autres sites.
- Avoir une architecture propre et correcte des titres.
- Utiliser les microdata indiquant aux moteurs de recherches de quel type de contenu il s'agit et offrant la possibilité de fournir des informations détaillées sur les produits,...
- Employer les microformat au niveau des comptes utilisateurs.

Certains points de cette liste ont été mis au point tandis que d'autres le seront dans le futur.

Des outils existent afin d'optimiser la place du site lors de recherche tels que WooRank, SEO Inspector qui ont été et seront utilisés.



L'objectif final ? Lors de la recherche d'une annonce via un moteur de recherches, les annonces de mon site devraient idéalement se trouver avant celles d'autres sites et si possible via l'utilisation des trois langues.

Evidemment cela ne sera pas aisé étant donné que mon site sera peu visité au début, possèdera peu d'annonces et ne disposera d'aucune ancienneté dans le monde du web.

Se montrer patient sera nécessaire et réaliser une bonne promotion de mon site sera important pour le mettre sur le marché.

CHAPITRE VIII.

GESTION DES LANGUES

AKoter propose 3 langues : le français, le néerlandais et l'anglais tout en sachant que la langue de base du site est le français.

Pour mettre en place il a fallu réfléchir à la gestion des langues des manières suivantes :

- «statique» : textes, phrases, mots fixes dans le site et ils sont donc définis de cette façon.
- «dynamique» : phrases enregistrées dans la base de données.

Pour vous donner une idée, voici les chiffres actuels concernant le nombre de traductions dans chacune des catégories :

- 2 001 traductions statiques, c'est-à-dire 667 traductions par langue.

- 12 393 traductions dynamiques, c'est-à-dire 4 131 traduction par langue.

Contrairement aux traductions dynamiques, les traductions statiques sont uniques.

MISE EN PLACE DES TRADUCTIONS STATIQUES

Laravel propose un système d'ordre classique via des fichiers PHP.

Chaque langue dispose d'un fichier nommé pareillement. Les dossiers disposent de fichiers PHP nommés semblablement de dossiers en dossiers.

Ces fichiers sont simplement une «array associative». Chaque ligne est définie par une «clef» qui doit être la même pour les différents dossiers.

Chaque clef dispose également d'une valeur changeante selon la langue.

Voici comment récupérer la valeur d'une clef : «Lang::get('validation.accepted')»

```
"accepted" => "Le champ :attribute doit être ac
```

Laravel possède également un «helper» pour simplifier l'écriture : «trans('validation.accepted')»

Le système offre également de la souplesse. Dans certains

cas, nous pourrions souhaiter entrer proprement des données au milieu d'une phrase traduite. Voici la manière de procéder : 'email' => 'Entrez votre email :prenom',

Lors de la récupération de la traduction on devra y ajouter un paramètre afin de remplacer «:prenom» comme ceci : «trans('validation.email', array('prenom'=>'Julien'))»

Le point faible de ce type de traductions réside dans le fait de devoir écrire la traduction autant de fois que le site possède de langues disponibles. De plus il faut écrire dans les fichiers ce qui n'est ni pratique ni simple pour des traducteurs.

Ainsi il m'a semblé judicieux de rechercher un moyen de les rendre plus facilement administrables voire de les faire traduire par un service tel que «Google Translate» ou «Bing».

J'ai commencé par utiliser Google Translate API mais celui-ci étant payant, je me suis vite retrouvé limiter par ce service. Je me suis tourné vers le service de Bing qui est quant à lui gratuit. J'ai du également trouver un service de détection des langues, gratuit également.

De cette manière j'ai découvert un «package» crée par Barryvdh(<https://github.com/barryvdh/laravel-translation-manager>). m'offrant ce que je désire.

Son «package» liste toutes les traductions créées. Ensuite, il crée une table sur notre base de données et la remplit avec nos traductions.

Toutefois notons qu'un panneau nous permet de réaliser des actions sur ces traductions (ajouter, modifier, supprimer,...).

Une fois toutes les étapes exécutées, nous pouvons enregistrer les traductions modifiées à la place des anciens fichiers.

Key	fr	en
accept	Accepter	Empty

Un point non géré par ce système mais sur lequel je n'ai pas eu le temps de me pencher serait de traduire automatiquement via un service les éléments manquants. Ensuite il resterait la vérification de la traduction.

MISE EN PLACE DES TRADUCTIONS DYNAMIQUES

La première étape correspondant à la préparation de la base de données.

Un chapitre concernant la gestion des tables de traductions existe déjà donc ce thème ne sera pas abordé dans ce chapitre ci.

Laravel offre la possibilité d'utiliser l'Orm Eloquent nous proposant une gestion de ce type de relations entre mes tables. Cette relation se nomme «Polymorphic».

Grâce à cette relation proposée par Laravel il est aisé de retrouver les traductions d'un ou de plusieurs éléments comme par exemple :

«`$user->translation()->get()`» dans mon cas permet de retrouver les traductions de la langue en cours d'utilisation.

«`$user->translations()->get()`» la méthode avec un «s» permet de retrouver les traductions dans toutes les langues.

DÉTECTER LA LANGUE À UTILISER

Pour avoir les bonnes traductions, il est nécessaire d'informer Laravel de la langue employée.

Segment

Ma première étape est de vérifier la présence d'un «/fr» ou de tout autre code pays dans l'URL.

Pour ce faire, nous examinons le segment n°1 c'est-à-dire le premier élément après le nom de domaine tel que : `www.akoter.be/jesuislepremiersegment/jesuislesecond`).

Si le segment correspond à l'une des langues proposée, j'en informe Laravel en redéfinissant la variable dite locale. Cette variable est réinitialisée à chaque requête et permet de savoir quel dossier de langue charger.

Si le segment est «/en» il lui sera assigné la valeur «en».

Utilisateur

S'il n'y a pas de segment correspondant à une langue, il

convient de regarder si l'utilisateur est connecté ou non. S'il l'est, il faut alors récupérer la langue définie dans les paramètres de son compte.

Le processus est le même par rapport à l'étape précédente : j'assigne une nouvelle valeur à la variable locale.

Navigateur

La dernière possibilité est de récupérer la langue employée par le navigateur du visiteur.

Si aucune de ces méthodes ne fonctionne, la langue de base est alors utilisée.

Changement de langue

Pour changer la langue via l'interface du site, l'utilisateur peut cliquer sur le bouton en haut à droite de la barre de navigation. Ce lien ajoute le segment souhaité dans l'URL et redéfinit la langue comme expliqué au point un.

Gestion des slugs

La gestion des slugs (remplace user/1 par user/julien-roland par exemple) avec cette unique table de traduction m'a posé quelques soucis.

En principe, le package créant automatiquement les slugs les ajoute dans la table courante. Cependant, si la décision d'optimiser la gestion multilingue est prise, il me faut créer un slug pour chaque langue lorsqu'il y a utilité.

Le package ne permet pas une gestion de ce genre. Il m'a donc fallu créer moi-même les slugs avec la fonction : `Str::slug()` de Laravel.

Afin d'être certain que le slug soit bien unique, je lui ai ajouté son identifiant à la fin. Voici un exemple : `annonces/192` devient `annonces/kot-namur-300-euros-192`

Gestion de recherche

Il m'a également fallu faire face à un problème au niveau de la recherche par mots clef sur la table de traductions.

Prenons un cas concret : nous souhaitons faire une recherche sur le nom de la région et également sur la valeur du titre des annonces. Ces deux éléments sont donc stockés dans la table des traductions, il me faut donc faire des «jointures» sur la même table.

N'ayant jamais rencontré ce problème, il m'a fallu y consacrer du temps pour pouvoir y remédier en donnant des «alias» à mes jointures.

CONCLUSION DES SYSTÈMES

Mes systèmes sont bien fonctionnels et mon site est donc bien disponible en trois langues.

Malgré cet avantage de proposer trois langues différentes, la gestion des langues nécessite beaucoup de temps. Au lieu de simplement écrire «bienvenue à vous», il me faut à chaque

fois utiliser la fonction, ajouter la phrase dans le fichier.

De même pour les traductions dynamiques, il faut réaliser une relation supplémentaire afin de rechercher une donnée. Ce processus complique la gestion dans certains cas de triage par exemple.

Toutefois, je suis plutôt content de moi et ne retire que du positif de cette expérience multilingue.

CHAPITRE IX. GESTION DES LISTINGS

LISTINGS

Généralement, les utilisateurs voient le listing comme rébarbatif et plutôt ennuyant alors qu'il devrait être agréable et efficace.

Pour réussir son listing, l'utilisabilité doit être le point principal, c'est-à-dire la simplicité, la compréhension et la satisfaction de l'internaute lors de la navigation.

Penser la disposition des éléments est la base des listings. Ainsi disposer le système de tri en haut du listing permet de le rendre disponible n'importe où dans la page.

Le listing occupe quant à lui toute la largeur et tout le reste de la page. En taille maximale, il est possible d'afficher 4 annonces par ligne.

Une pagination a été mise en place pour la création du lis-

ting. Un nombre de locations déterminé est affiché par page créant donc plusieurs pages en fonction du nombre de locations.

Dans le but de rendre la pagination plus agréable et éviter de cliquer en permanence sur suivant pour afficher chaque nou-

Recherche par mots clefs, référence d'annonce

Chercher

Filtrer par: Type de location Prix Dates Charges Superficie Options Position

Carte Liste

1202 résultats

Location Villa Paris XIV P... KOT Wépion	ANONCE QUALITATIVE TYP... MAISON Wépion	FRITITLE LONG OU COURT DUPLEX Koolkerke	FRITITLE LONG OU COURT KOT Bovelingen
0 votes	0 votes	549 votes	401 votes
1 place	4 places	9 places	8 places
Charges sur consommation	Charge de 95€	Charge de 241€	Charges comprises
Loyer de 1600€	Loyer de 300€	Loyer de 961€	Loyer de 686€

velle page, un système différent a été mis en place.

Lorsque l'utilisateur a un niveau de «scroll» (hauteur dans la page) égal à la hauteur maximale de la hauteur finale du listing - la hauteur de deux lignes d'annonces, les données contenant la suite des annonces de la page suivante sont récupérées et sont rajoutées à la page en cours et ceci à chaque fois que les conditions sont remplies.

Ce système rend plus simple le scan des annonces, il permet d'éviter de devoir faire attention à d'autres éléments tels que les pages.

Notons toutefois que si l'utilisateur ne dispose pas de Javascript la pagination classique sera affichée.

TRIS

Le système de tri est affiché sous forme de barre de navigation, scindée en trois parties :

- la recherche par mots clés (grande largeur) ;
- les filtres et les tris ;
- les éléments d'actions (filtrer).

Recherche par mots clés

Le but est la recherche de locations via des mots clef évitant de cibler un élément particulier ou d'entrer des valeurs. Notons toutefois que le rayon de recherche est limité au titre et au type de location.

Les filtres et les tris

Ils doivent être cliqués pour pouvoir être complétés par l'internaute sur base du type de location, des prix, des dates, des charges, de la superficie, ..

Ceux-ci offrent deux boutons d'actions supplémentaires : un pour continuer de filtrer et l'autre pour activer les filtres définis.

CHAPITRE X.

INSCRIPTIONS

Les filtres requièrent un rechargement de page. En effet cela devenait compliqué et fastidieux de réaliser ce processus via Ajax.

Les éléments d'action

Ils sont au nombre de deux : la recherche et le changement de vue entre le listing et la carte.

Le deuxième point permet à l'utilisateur de pouvoir faire une nouvelle recherche via la carte sans devoir retourner sur la page d'accueil.

La première partie du formulaire d'inscription est l'inscription du compte, commune aux propriétaires et locataires.

Cette inscription peut être réalisée de différentes manières : via le formulaire d'inscription qui comporte 8 champs à compléter ou via Facebook en un seul clic.

Une fois l'inscription terminée, si nous souhaitons devenir propriétaire certaines conditions devront être remplies de même que pour les locataires.

PROPRIETAIRES ET AGENCES

Afin d'être propriétaire nous nous retrouvons sur une page indiquant les différentes étapes à réaliser : compléter le profil, valider son e-mail, inscrire une location, valider l'inscription par un administrateur et la mise en ligne du bien.

Dans un premier temps, il faut compléter les champs requis du profil, repérables à l'aide des astérisques (*) rouges à droite des champs concernés et faire vérifier son adresse e-mail via un lien classique.

Une fois les informations indiquées, nous sommes redirigés vers le formulaire d'inscription de location.

Chaque étape dispose d'un titre et d'une explication. Les étapes à suivre sont clairement affichées et compréhensibles.

Ainsi il faut remplir les informations en rapport à la localisation du bien telles que l'adresse, la région, la ville et localiser le bâtiment sur la carte à l'aide des informations données avant de pouvoir passer à l'étape suivante.

Dès le passage à l'étape suivante, nous avons une information claire par rapport aux nombres d'étapes complétées. En effet, un message indique que la localisation du bien a été

enregistrée et le «schéma» des étapes se colorie en vert au fur et à mesure de l'avancement des étapes validées.

La prochaine étape correspond à l'information par rapport au nombre de logements mis en location.

Un bien peut contenir plusieurs types de location qui sont à ajouter dans le formulaire.

Une annonce spécifique peut être créée pour chaque location dans un même type de location. En effet, nous pouvons mettre 5 kots en ligne et décider d'indiquer une annonce spécifique pour chacun d'entre eux.

Ensuite, via des checkboxes, les particularités du bâtiment sont à ajouter telles que la présence d'une terrasse, ...

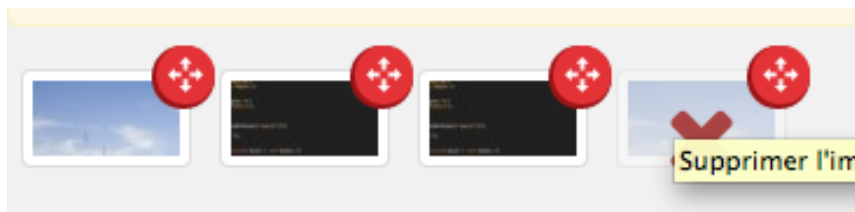
Lors des étapes suivantes, il sera possible de rédiger l'annonce. Celle-ci sera automatiquement traduite via Bing Translate mais, toutefois, l'utilisateur peut décider d'écrire lui-même dans les trois langues voire de modifier le texte de l'annonce une fois la traduction réalisée.

Pour le choix de langue lors de la rédaction de contenu, j'ai opté pour JQuery UI Tabs avec un style basé sur Bootstrap pour avoir un impact visuel moins prononcé. Dans certains cas, j'ai également employé TinyMce en remplacement des textareas.

TinyMce est limité au gras, italiques et aux commandes copier / coller,...

Au niveau des photos, elles sont gérées en Ajax permettant de les visualiser et également de les trier via JQuery UI Sort.

Le système permet l'envoi d'une ou de plusieurs images. Elles sont ordonnées dans des dossiers et sous-dossiers.



Chaque image comporte 6 versions dont une originale redimensionnée dans une largeur de 1024 px. La taille des autres images varie en fonction de leur utilité. Voici la liste de leur situation dans le site :

- Miniature : Utilisé dans l'inscription et dans la liste des images disponibles dans le slider d'une annonce.
- Moyenne : Utilisé sur la fiche d'une annonce et dans diverses situations du compte utilisateur.
- Carte : Utilisé dans le slider de la carte lors de la vue d'un bâtiment.
- Galerie : Utilisé dans la galerie d'images sur la vue d'une location.
- Lightbox : Utilisé dans le slider.

L'étape des annonces est particulière. On retrouve un système de «Tabs». Ceci dépend du nombre de locations créées auparavant. Si nous avons choisi 5 kots avec des annonces spécifiques, chacun disposerait d'un panneau.

La référence est indiquée en supplément du type de logement et la suite correspond à un formulaire d'inscription classique.

Une fois l'inscription finie, il faut qu'un administrateur vienne confirmer l'annonce pour la rendre visible par tout le monde. Afin d'accélérer ce processus, il est possible d'envoyer un message ceux-ci étant prioritaires. Notons que dans le futur, devenir prioritaire pourrait être une action payante.

LOCATAIRES

Tout comme pour le propriétaire, le locataire devra compléter les champs requis du profil, repérables à l'aide des astérisques (*) rouges à droite des champs concernés et faire vérifier son adresse e-mail.

Pour pouvoir se proposer pour une location, le locataire devra disposer d'un profil valide et validé par un administrateur.

CHAPITRE XI.

COMPTES UTILISATEURS

Les comptes utilisateurs permettent de s'inscrire en tant que propriétaire ou locataire, raison principale de la création.

Globalement il offre la possibilité de configurer ses informations personnelles, son compte (suppression, suspensions, activation) et permet diverses manipulations telles que l'envoi de messages, la mise en favori d'annonces,...

Du point de vue du propriétaire, il possède davantage d'informations telles que celles en rapport à ses clients et à ses locations ...

Notons qu'un locataire peut également être un propriétaire ainsi que le contraire.

Les comptes sont gratuits et le resteront. Certains bonus, des nouveautés, pourront éventuellement être payantes si elles ne sont pas «vitales».

PROPRIÉTAIRES ET AGENCES

Lorsqu'un propriétaire gère son compte, il a la possibilité de s'enregistrer en tant que professionnel.

Dans la configuration des informations personnelles, nous pouvons cocher la case «s'inscrire en tant qu'agence» et valider l'inscription.

Un onglet agence apparaît sur la navigation du compte permettant de nouvelles actions : créer un «compte» pour l'agence avec son logo.

La personne créant le compte devient le propriétaire du compte de l'agence et pourra transmettre les codes d'accès aux autres personnes de l'agence pour qu'elles puissent avoir accès au compte de l'agence.

De nombreuses possibilités existent pour développer ce système d'agence mais doivent encore être mises en place.

Le but est de signaler que, la location en cours de création, est un bien de l'agence. Cette agence serait mise en avant à la place de l'utilisateur.

Un visiteur pourrait cliquer sur l'agence pour visualiser la liste des locations dont elle dispose et d'autres informations utiles.

Une API personnalisée serait également prévue permettant d'afficher leurs locations sur leur propre site.

Ce service serait probablement payant étant donné qu'il s'agit d'une activité professionnelle (agence) et de la visibilité offerte.

LOCATAIRES

En complément au compte basique, des informations par rapport aux biens loués seront disponibles ainsi que certains outils de gestions tels que les contacts directs avec le propriétaire,...

VÉRIFICATION DE L'E-MAIL

Pour être actif en tant que propriétaire ou locataire, l'adresse e-mail doit préalablement être vérifiée via un système classique.

SUSPENSION ET SUPPRESSION DE COMPTE

A tout moment, un utilisateur peut décider de suspendre son compte. Ainsi, celui-ci n'est plus accessible et, dans le cas d'un propriétaire, ses annonces ne sont plus accessibles.

En se reconnectant sur son compte, l'utilisateur peut réactiver son compte sans nécessité de faire appel à un administrateur.

Lors de la suppression d'un compte, le site ne procède pas

réellement de cette manière. En effet, il garde le compte en mémoire pendant 3 ans et si l'utilisateur ne refait pas surface au bout de ces 3 ans, le compte sera alors définitivement supprimé.

Si l'auteur du compte désire revenir sur le site, il est ainsi facile pour les administrateurs de réactiver son compte et donc de récupérer ses données.

GROUPES UTILISATEURS

Actuellement, le site dispose de 3 groupes d'utilisateurs : le groupe membre dont tout inscrit fait partie, le groupe admin ayant accès à l'administration du site pour pouvoir le gérer et, enfin, le groupe super admin possédant des droits supplémentaires dont celui de supprimer les comptes admin.

D'autres rôles feront leur apparition tels que le rôle de traducteur.

Les rôles particuliers disposeraient de permissions définies sous forme «JSON» mais la réalisation de ce point n'est pas encore terminée.

Un point spécifique est consacré à la partie d'administration et ne sera donc pas abordé actuellement.

CHAPITRE XII.

COMMUNICATION

La communication est un de mes objectifs personnels : celui de rendre le site facilement utilisable sans pour autant devoir simplifier les outils.

Par conséquent, le rôle de la communication est primordial afin d'aider les personnes moins à l'aise avec les sites web à parvenir à l'utiliser de manière optimale.

La communication figure à différents niveaux du site :

- Informations importantes données au niveau des formulaires, informations d'erreurs, de réussite mais surtout d'utilisation.
- Informations retrouvées lors de certains tutoriels tout au long du site pour guider l'utilisateur avec les outils plus difficiles tout en laissant le choix à l'internaute aguerri de masquer l'information temporairement ou définitivement.

- Lors de l'inscription, une introduction est présente dont le but est de comprendre l'utilité des formulaires, message indiquant la validation d'une étape, expliquant l'utilisation de certains points, ...

Une fois l'étape de l'inscription passée, des pages nous informent des étapes à suivre lors de certaines actions.

La communication visuelle est principalement réalisée au niveau des couleurs. La plupart des couleurs claires ont pour objectif de donner une certaine envie, une certaine gaieté lors de la visite.

Les couleurs peu agressives sont présentes dans les formulaires et lors d'envoi d'informations importantes.

Ainsi, dans certains cas, comme dans les formulaires, un fond blanc est utilisé pour éviter de distraire l'internaute et de ne pas attirer l'attention sur le fond mais sur les données à compléter.

Lors de certains messages d'informations d'erreurs, la couleur est utilisée pour donner un aperçu du message. C'est par exemple le cas du rouge indiquant qu'un champ n'a pas été complété alors qu'il est requis.

Des icônes sont également utilisées pour donner un aperçu d'un message.

La plupart du temps, lors d'un survol, un message nous donne une indication pertinente.

Les e-mails doivent également être bien réalisés afin de ne pas nuire à la communication tout en offrant un visuel attractif.

CHAPITRE XIII.

ADMINISTRATION

Comme pour tout site dont l'ambition est d'être géré sans toucher aux ligne code, créer une administration est nécessaire et se trouve sur l'adresse : Akoter/admin

Pour y avoir accès, il faut être connecté et disposer d'un compte de type admin.

Pour ce faire j'ai opté pour Twitter Bootstrap comme framework HTML / CSS étant donné la simplicité d'utilisation et le style qui m'a tout de suite conquis. En effet, je trouve que cela convient parfaitement pour une administration.

Les styles proposés sont communs facilitant ainsi la prise en main pour les habitués du web.

Cette admin dispose d'un menu redirigeant vers les catégories en question. Sur la page d'accueil, on peut trouver des statistiques ainsi que le nombre de demandes en attente,...

Dans chaque catégorie, une page listant les éléments est disponible. Cette page possède également une pagination et un système de recherches.

Chaque élément dispose d'éléments d'actions pour pouvoir visualiser, éditer et même contacter dans certains cas.

Le contact offre la possibilité de traduire le message dans la langue de l'utilisateur. Dans cette partie, on retrouve également le gestionnaire des traductions statiques.

La partie destinée à l'admin nécessite encore d'évoluer et de donner plus de possibilités aux personnes l'employant.

CHAPITRE XIV.

GESTION DES ARTICLES

Des articles sont présents un peu partout sur le site mais surtout dans la partie blog.

Ces articles doivent être rédigés par des administrateurs. Pour ce faire, une section article doit être créée dans l'administration.

Cette interface serait semblable à celle de «CMS», système de gestion de contenu, très populaire.

Titre <input type="text" value="Akoter en chiffre ça donne quoi"/>	Posté par : Julien Roland
Uri généré <input type="text" value="akoter-en-chiffre-ca-donne-quoi"/>	Crée le : Mercredi 30 Novembre -1 0h00
Contenu <div>Editer ▾ <div>B <i>I</i> </div><div> Lorem ipsum dolor sit amet, consectetur adipiscing elit. Aperiam porro error sed et rem ut optio eaque recusandae ipsum. Nemo, dolore enim obcaecata iure aspernatur ipsam tempore sunt suscipit quod.</div><div><p>p</p><p>Mots : 30</p></div></div>	Dernière modification le : Mercredi 30 Novembre -1 0h00
	Publié:
	Page: <input type="text" value="home"/>
	Position: <input type="text" value="mod5"/>
	Modifier
	Photo la une

L'objectif serait de développer complètement un outil autonome et performant tout en alliant une simplicité d'utilisation.

Les «CMS» m'inspirant sont les suivants :

- Wordpress ;
- Prestashop ;
- Perch ;

Toutefois, pour l'instant je me suis uniquement basé sur Wordpress.

Chaque article dispose d'un titre, d'un contenu, d'informations en rapport avec la date de création,... et d'une image à la une.

Les articles peuvent être accrochés à une page et un emplacement. Contrairement aux «CMS» ce n'est pas un réel hook. Ceux-ci doivent être préalablement définis, c'est-à-dire qu'à l'emplacement mod5 de la page home, une fonction recherchant si un article possédant ces attributs est disponible.

Les articles peuvent être modifiés et supprimés mais on peut également prendre la décision de ne plus les publier ou inversement.

CHAPITRE XV.

PERFORMANCE

Pour tous les projets conséquents, principalement ceux disposant d'une base de données volumineuse, il est important de ne pas négliger les performances.

L'optimisation se fait sous différents aspects tels que :

- Le nombre de requêtes établies avec la base de données;
- La conception des scripts en tout genre mais principalement ceux en PHP et JS;
- Le nombre de requêtes envoyées et reçues par le navigateur;
- Le poids des éléments envoyés;

En principe, cette étape est effectuée une fois le site fini et on peut employer des outils tels que GTMetrix, page speed,...

Mon site n'était pas encore fini il n'est donc pas optimisé comme il devrait l'être.

GESTION DES SCRIPTS ET DES STYLES

Ce genre de fichier va être minifié, les retours à la ligne seront supprimés et les mots raccourcis de manière à ce qu'ils deviennent des abréviations.

```
(function(n){var e=n(".mapContainer"),t=n(".main"),o=n('
```

Les fichiers seront concaténés (réunis en un fichier) permettant de ne plus charger qu'un seul fichier au lieu de plusieurs.

GESTION DE CACHE

Dans le cadre de mon projet Laravel, le système Eloquent a été employé afin de récupérer les données en base de données.

Ce système a tendance à augmenter le nombre de requêtes exécutées comparé au système classique. Dans ce cas, le cache nous permet de sauvegarder chaque requête établie par Eloquent.

Des résultats complets de requête peuvent également être enregistrés.

Pour mon site le cache a été utilisé dans certains cas mais pas dans l'entièreté du site.

Toutefois, soulignons qu'il est important de ne pas mettre en

cache des données susceptibles de changer rapidement.

Il faut également penser à supprimer le cache à un moment précis et ceci dans certains cas. Par exemple, lors de l'envoi de messages à un utilisateur, il faut supprimer le cache sur les messages avant l'envoi.

```
$dynamiqueTrans = Translation::remember(Config::get('var.remember') * co  
$staticTrans = Helpers::cache($this->lang->getAll()->count() * co
```

AUTRES

Via l'utilisation de GTMetrix, on découvre l'attention à apporter aux images, à l'importance de leur spécifier une dimension. Cet outil m'a permis de définir des priorités d'optimisation.

Depuis la découverte de cet outil, il m'est difficile de m'en passer. Le fait d'avoir un résultat précis sur les performances de notre site donne envie de chipoter et de grattouiller afin de gagner des pourcentages.

La rapidité du serveur est également importante. Toutefois, lors de la création de mon portfolio j'ai constaté que mon serveur OVH était très lent. A la fin de mon contrat, je vais envisager de me diriger vers un autre service tel que Gandi.

CHAPITRE XVI.

API

L'API offre des URL donnant, par exemple, la possibilité au développeur web de récupérer des données de locations et d'écoles.

L'API sera disponible via le sous domaine du site : akoter.julien.roland.be/api/doc et disposera d'une documentation dont le but sera de référencer ces URL et d'en montrer le fonctionnement.

Diverses méthodes seront également créées pour rendre l'API polyvalente

Locations method

Get all valid locations

Method all will retrieve all locations, here is how to use with jQuery \$.ajax

EXAMPLE

```
$.ajax({  
  url:"http://akoter/api/location/all",  
  dataType: "json",  
  type:"get",  
  success:function(oData){  
    console.log(oData);  
  }  
});
```

CHAPITRE XVII.

CONCLUSION

« Rome ne s'est pas faite en un jour ».

Opter pour un sujet avec un fort potentiel au niveau de la création était un minimum selon moi pour pouvoir améliorer mes connaissances générales et progresser dans plusieurs domaines.

Toutefois il me semblait évident qu'il n'était pas possible de vous présenter un site web complet dans les délais impartis. Néanmoins il est pour moi important de pouvoir vous démontrer sa fonctionnalité lors des tâches basiques et de vous faire découvrir le potentiel dont le site dispose.

Me sera-t-il un jour possible de clôturer ce projet ? Les possibilités de développement du site web étant infinies il ne me semble pas possible de le mettre sur le marché avant la rentrée scolaire 2015 – 2016 d'autant plus qu'il est préférable de le mettre en ligne lors d'une nouvelle année scolaire et pas d'une année déjà entamée.

« C'est en forgeant qu'on devient forgeron ».

Durant les cours nous avons eu le temps de commencer notre projet de fin d'études, les professeurs nous indiquant les outils à employer en fonction des projets de chacun. C'est ainsi qu'il m'a été conseillé d'utiliser l'outil nommé « laravel ».

Ne connaissant pas cet outil, il me faut avouer que j'avais quelques incertitudes à propos de sa conception. Pour y parer, apprendre à l'utiliser en supplément des cours m'a semblé être une bonne option. Toutefois il s'est avéré que cela a été une erreur : commencer par la structure des pages afin de définir un cahier des charges précis aurait été une bien meilleure solution d'autant plus que nous avons par après eu des cours nous permettant de découvrir cet outil.

Après les examens de Noël, mon niveau concernant « laravel » s'était nettement amélioré, ne me laissant pas d'autre alternative que de repartir sur de bonnes bases : coder en anglais et non en français, mettre en pratique les conventions « laravel »,...

Par conséquent je me suis retrouvé avec un projet sans maquette et n'ai pas eu d'autres choix que de réaliser la maquette et le développement en même temps.

D'autre part, mon erreur a été de commencer par le code, ne maîtrisant pas laravel, de m'occuper du visuel mais puisque mon apprentissage avait progressé, il m'a fallu recommencer tout le codage me faisant ainsi perdre un mois de travail pour la concrétisation de la maquette de la page d'accueil incluant

les divers changements nécessaires pour obtenir un résultat me convenant.

Les principales difficultés rencontrées au cours de la réalisation de cette maquette se situent au niveau du design.

« A l'impossible nul n'est tenu ».

Je me suis rapidement rendu compte qu'il m'aurait fallu développer les éléments essentiels en priorité car certains éléments importants du projet ne sont pas terminés alors que le délai imparti touche à sa fin.

Certains éléments représentaient une découverte tels que le système de langue, l'apprentissage d'un codage correct. Etant donné qu'il s'agissait de nouveautés, il m'a semblé utile de m'y attarder, de faire plusieurs essais, ceci me prenant plus de temps et retardant la réalisation des objectifs ?

Je suis cependant conscient qu'une fois sur le marché du travail, l'aspect « apprentissage » devra passer en arrière-plan.

Le mot de la fin ?

En l'état des choses mon projet me semble satisfaisant, posédant de bonnes bases lui permettant d'évoluer sans devoir le remanier de façon importante.

Plusieurs points positifs sont également à noter comme l'apprentissage de Laravel ainsi que la gestion multilingue d'un

site et la rapidité et la qualité de mon codage.

D'autre part, c'est également grâce à ce projet que j'ai enrichi mon expérience au niveau du développement de sites volumineux, me permettant d'aborder la réalisation de petits projets avec plus de facilité qu'auparavant.

Ce projet m'a permis de confronter mes qualités de développeur plutôt que celles de designer mais il me semble fondamental de rester polyvalent.



