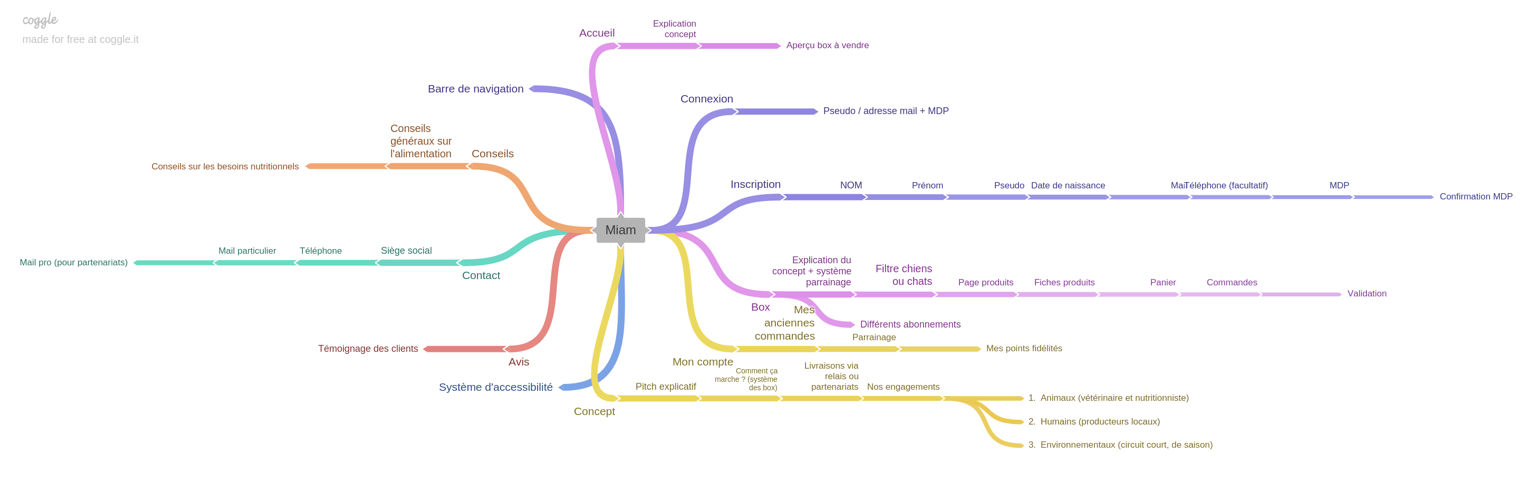
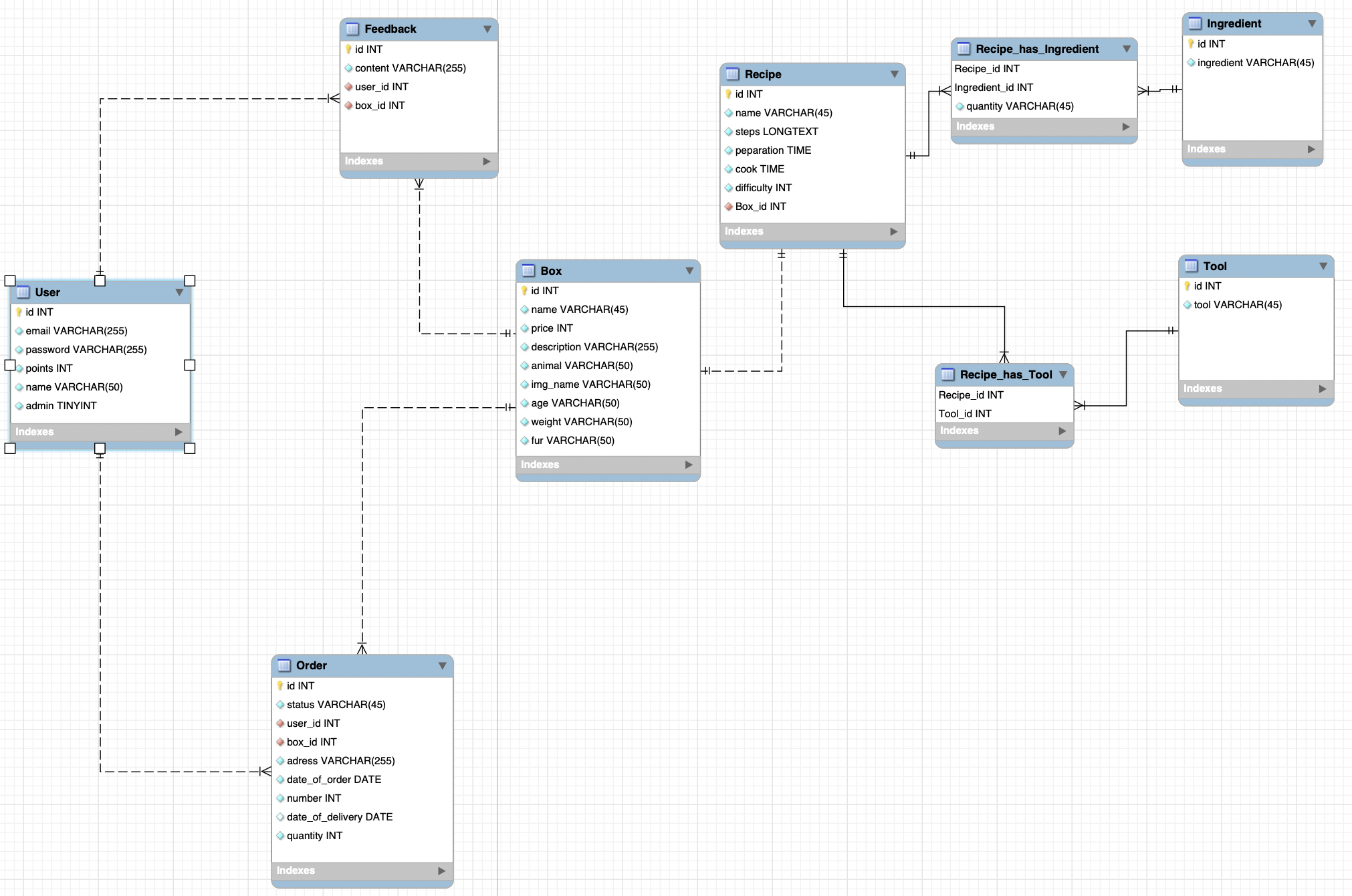
Dossier Développement Miam :

1/ Site map



2/ Schéma de la base de données



3/ Cahier des charges technique sur les outils utilisés

|  |  |
| --- | --- |
| Fonctionnalités | Fonctionnement |
| Header | Le header sera présent sur toutes les pages afin de pouvoir naviguer facilement peu importe la page sur laquelle nous nous trouvons. |
| Header - Logo | Le logo permettra de reconnaître le site et de savoir sur quel site nous nous trouvons lorsque nous sommes dessus. De plus, il servira de point d’accès à la page d’accueil peu importe la page sur laquelle nous nous trouvons.  Il sera situé sur la gauche du header. |
| Header - Menu | Le menu permettra de visualiser les différentes pages du site, ainsi que de pouvoir s’y rendre rapidement. Il sera composé de 7 pages : accueil, concept, boutique, conseils, contact et avis. |
| Header - Menu - Bloc accueil | Le texte sera affiché en bleu dans la police Verdana.  Lorsque l’on sélectionne ce bloc, le titre accueil devient alors blanc encadré dans un carré bleu. Celui-ci sera situé sur la gauche de la page, après le logo. |
| Header - Menu - Bloc concept | Le texte sera affiché en bleu dans la police Verdana.  Lorsque l’on sélectionne ce bloc, le titre concept devient alors blanc encadré dans un carré bleu.  Celui-ci sera situé au centre de la page, après le bloc accueil. |
| Header - Menu - Bloc boutique | Le texte sera affiché en bleu dans la police Verdana.  Lorsque l’on sélectionne ce bloc, le titre boutique devient alors blanc encadré dans un carré bleu.  Celui-ci sera situé au centre de la page, après le bloc concept. |
| Header - Menu - Bloc conseils | Le texte sera affiché en bleu dans la police Verdana.  Lorsque l’on sélectionne ce bloc, le titre conseils devient alors blanc encadré dans un carré bleu.  Celui-ci sera situé à droite de la page. |
| Header - Menu - Bloc contact | Le texte sera affiché en bleu dans la police Verdana.  Lorsque l’on sélectionne ce bloc, le titre contact devient alors blanc encadré dans un carré bleu.  Celui-ci sera situé à droite de la page, après le bloc concept. |
| Header - Menu - Bloc avis | Le texte sera affiché en bleu dans la police Verdana.  Lorsque l’on sélectionne ce bloc, le titre avis devient alors blanc encadré dans un carré bleu.  Celui-ci sera situé à droite de la page, après le bloc contact. |
| Header - Bouton s’inscrire | Le bouton s’inscrire permet à l’utilisateur de se créer un compte sur notre site afin de pouvoir profiter de nombreux avantages. Ce bouton se trouvera sur la droite du header. |
| Header - Bouton Panier | Ce bouton permettra d’accéder à tout ce que nous avons mis dans notre panier, que ce soit de notre visite du jour ou non. Il suffira de cliquer dessus pour y avoir accès.  Il sera situé à droite de la page, après le bouton s’inscrire mais avant celui de mon compte.  Il sera représenté par un doggybag avec un os à l’intérieur.  Le texte sera affiché en bleu dans la police Verdana.  Lorsque l’on sélectionne ce bloc, le titre panier devient alors blanc encadré dans un carré bleu. Celui-ci sera situé sur la gauche de la page, après le logo.  Il sera également affiché une petite icône précisant des notifications. Celle-ci sera orange avec l’écriture du chiffre en bleu. |
| Header - Bouton Mon Compte | Le bouton mon compte permettrait d’accéder à son compte fidélité, ses différents avantages, ses commandes en cours ainsi que ses anciennes commandes s’il souhaite recommander la même chose mais également ses parrainages.  Ce bouton se trouvera sur la droite du header.  Il sera sous la forme d’une icône représentant l’animal que nous possédons.  Le texte sera affiché en bleu dans la police Verdana.  Lorsque l’on sélectionne ce bloc, le titre mon compte devient alors blanc encadré dans un carré bleu. Celui-ci sera situé sur la gauche de la page, après le logo.  Il sera également affiché une petite icône précisant des notifications. Celle-ci sera orange avec l’écriture du chiffre en bleu. |
| Header - Fonction Désinscription | Cette fonction permettra à l’utilisateur de pouvoir se désinscrire du site s’il jamais il n’a plus envie de profiter de nos services. Pour se désinscrire il suffira de désactiver une fonctionnalité que nous avons activer lors de l’inscription. Après cela, un mail sera envoyé avant de valider l’action de désinscription. |
| Header - Barre de recherche | A côté des boutons ou juste en-dessous, nous trouvons une barre de recherche qui nous permet de rechercher tout ce dont nous avons besoin sur le site. Par exemple, rechercher un service particulier ou un produit spécifique. |
| Page inscription | Lorsque l’on clique sur le bouton inscription, nous sommes redirigés sur la page inscription où nous pouvons rentrer toutes nos informations. |
| Page inscription - Bloc Nom de famille | Ce bloc permet à l’utilisateur de rentrer les informations concernant son nom de famille. |
| Page inscription - Bloc Prénom | Ce bloc permet à l’utilisateur de rentrer les informations concernant son prénom. |
| Page inscription - Bloc Pseudo | Ce bloc permettra à l’utilisateur de se choisir un pseudo. Il s’agira du nom qui sera affiché lorsque l’utilisateur se connectera à son compte. |
| Page inscription - Bloc Mail | Ce bloc servira à rentrer l’adresse mail de l’utilisateur. Celle qui lui servira pour se connecter à son compte. |
| Page inscription - Bloc Mot de passe | Ce bloc servira à l’utilisateur de choisir un mot de passe pour son compte, qui lui permettra de se connecter facilement. |
| Page inscription - Bloc Confirmation du Mot de passe | Ce bloc permet de confirmer le mot de passe choisi auparavant, afin d’être sûr de celui-ci. |
| Page connexion - Bloc Mail | Ce bloc sert à rentrer l’adresse mail que l’utilisateur a choisi pour créer son compte. |
| Page connexion - Bloc Pseudo | Ce bloc permet de rentrer le pseudo choisi par l’utilisateur comme nom de compte. L’utilisateur a le choix de rentrer soit son adresse mail soit son pseudo pour se connecter. |
| Page connexion - Bloc Mot de passe | Ce bloc permet à l’utilisateur de rentrer le mot de passe choisi afin d’avoir l’accès à son compte. |
| Body - Page accueil - Présentation | Sur le haut de cette page, nous présenterons le concept de notre site, nous expliquerons l’histoire de notre site... |
| Body - Page accueil - Bloc paiement sécurisé | Ce bloc permet de rassurer les clients et de leurs montrer que le fait de donner leurs coordonnées bancaires ou autres ne risquent rien. Il s’agit également d’une norme à respecter. |
| Body - Page concept - Bloc présentation | Ce bloc contiendra toutes les informations à savoir sur notre concept (l’histoire, d’où est-il né, axe de réflexion…) |
| Body - Page concept - Bloc box | Ce bloc présentera le concept des box pour cuisiner pour son animal. Nous expliquerons donc comment cela fonctionne. |
| Body - Page concept - Bloc engagements | Ce bloc mettra en avant nos différents engagements quant à notre concept. Il présentera donc nos trois engagements : animaux, humains et environnementaux. |
| Body - Page boutique - Bloc filtres | Ce bloc permettra de filtrer les articles selon la catégorie d’animaux que nous possédons afin de faciliter la navigation et de ne pas perdre de temps. |
| Body - Page boutique - Page produits | Cette page permet de visualiser tous les articles proposés par le site par rapport aux filtres demandés. Il s’agit d’une vue générale des produits. |
| Body - Page boutique - Page produits - Fiche produits | La fiche produit est la carte d’identité d’un produit. Elle permet de voir le détail d’un produit ainsi que ses caractéristiques. |
| Body - Page boutique - Page panier | Une fois le produit ajouté au panier, nous pouvons avoir accès à la page panier qui récapitule tous les produits que nous avons sélectionnés et que nous souhaitons acheter. |
| Body - Page boutique - Page commande | Lorsque le panier est rempli, nous passons à la phase commande. Pour cela il suffit de dire la quantité souhaitée par produit… Nous pouvons également décider du lieu de livraison ainsi que la date. Il faut ensuite décider d’un mode de paiement et effectuer ce paiement. |
| Body - Page boutique - Page validation | Cette page est la dernière avant l’achat définitif. Il s’agit d’un récapitulatif de la commande, ainsi que des modalités de paiement et de livraison. |
| Body - Page conseils | Sur cette page seront donnés tous types de conseils concernant les animaux mais également en ce qui concerne leurs alimentations (ce qui est bon ou mauvais, à quelle fréquence certains aliments ont-ils besoin d’être consommés par l’animal…) |
| Body - Page contact | Cette page contient toutes les informations sur l’entreprise qui peuvent être utiles aux visiteurs (siège social, téléphone, mail particulier, mail pro…) |
| Body - Page avis | Cette page permettra aux différents visiteurs du site de pouvoir laisser un commentaire et un avis. Cela servira à se rendre compte si le site plaît ou non ainsi que ce que nous devons améliorer. |
| Page accueil - Accessibilité | Des boutons seront présents sur la page d’accueil afin de permettre l’accessibilité du site aux handicapés. Des boutons permettent de changer les contrastes du site. D’autres d’agrandir ou de réduire la taille du texte, de modifier l’espacement des lettres, la couleur du texte... |

4/ Veille technique

Nos choix premiers :

🡪 En front :

* ReactJs : React (aussi appelé React.js ou ReactJS) est une bibliothèque JavaScript libre développée par Facebook depuis 2013. Le but principal de cette bibliothèque est de faciliter la création d'application web monopage, via la création de composants dépendant d'un état et générant une page (ou portion) HTML à chaque changement d'état. Elle est utilisée par de grosses sociétés tel que facebook.
* VueJS : Vue.js est un framework Javascript créé par Evan You en il y a trois ans (en 2014), un ancien de chez Meteor. Il est assez proche de React dans son fonctionnement (flux monodirectionnel pour les données, avec le pattern Flux) et de Polymère pour sa gestion de composants (les customs elements pour Polymère). Là où vue se différencie de ses concurrents est par sa polyvalence extrême.
* From scratch: HTML5/CSS3 avec JS sans framework.

🡪 En back :

* Symfony : Un framework PHP très gros qui peut être utilisé en tant qu’API ou bien pour tout le site, c’est est un framework qui représente un ensemble de composants (aussi appelés librairies) PHP autonomes qui peuvent être utilisés dans des projets web privé ou open source.
* PHP vanilla : PHP en langage de back, sans framework , le PHP est un language de programmation côté serveur
* NodeJS : Un framework javascript qui permet de faire des serveurs et de faire du back en JS.

Ce que nous avons choisi :

🡪 ReactJS/NodeJS, nous avons opté pour ce couple car :

* ReactJs est plus simple d’utilisation, il permet de séparer les différentes portions du code de manière plus simple qu’en HTML/CSS grâce à l’utilisation de components qui peuvent se séparer autant de fois qu’il le faut.
* ReactJs donne accès à des librairies et des composants multiples comme par exemple react-router qui permet d’avoir une architecture encore plus simplifiée par la définition d’un fichier uniquement dédiée au routing et qui permet aussi de pouvoir définir des components qui seront sur toutes les pages comme le header et le footer.
* Il permet un rechargement instantané de la page, il re rend les éléments automatiquement à chaque changement dans la page, c’est-à-dire que seuls les éléments modifiés au sein de la page web seront mis à jour sitôt qu’un état sera changée . Ce qui permet à l’utilisateur une expérience bien plus agréable qui n’aura pas de temps de rechargement lors d’une action ou bien d’un clic sur un bouton.
* Une simplification de la compréhension des actions lors du clic sur un élément du DOM avec des fonctions Javascript directement appelé sur l’élément HTML
* Un chargement du DOM très rapide.

🡪 Pour le nodeJS :

* Avoir les deux mêmes langages en back et en front est une économie de temps et d’argent car elle permet de coder dans le même langage, ce qui est beaucoup plus pratique que de passer d’un langage à l’autre, (ce qui nécessite généralement dans une entreprise un développeur pour le back et un pour le front). Tout le monde peut comprendre les deux parties de l’application.
* Le NodeJS a une conception asynchrone, c’est un modèle non bloquant et cela le rend très performant, car il peut lancer plusieurs requêtes en même temps et il traitera les résultats au fur et à mesure du moment de leur chargement sera fini.
* Une communauté très active et très suivi avec un grand nombre de packages à importer et d’autres qui s’ajoutent très régulièrement.
* Une technologie stable et qui n’est pas près de choir au vu du nombre d’entreprises qui la supportent et l’utilisent : Netflix, Trello, PayPal, LinkedIn, Walmart, Uber, Medium, Groupon, Ebay ou la NASA.

Nous avons écarté :

* Le HTML5/CSS3 avec JS sans framework car il est moins bien structuré, il n’y a pas le principe des composants ni du router.
* Le vue car nous préférions le système de templating du react à celui du vue.
* Le PHP vanilla et le symfony car nous préférions avoir le même langage front et back que deux langages vraiment différents.