

Réflexions sur le projet Hotels Hypnos

Pour commencer, je n'ai pas souhaité utiliser de framework type Symfony, car je ne me sens pas à l'aise avec.

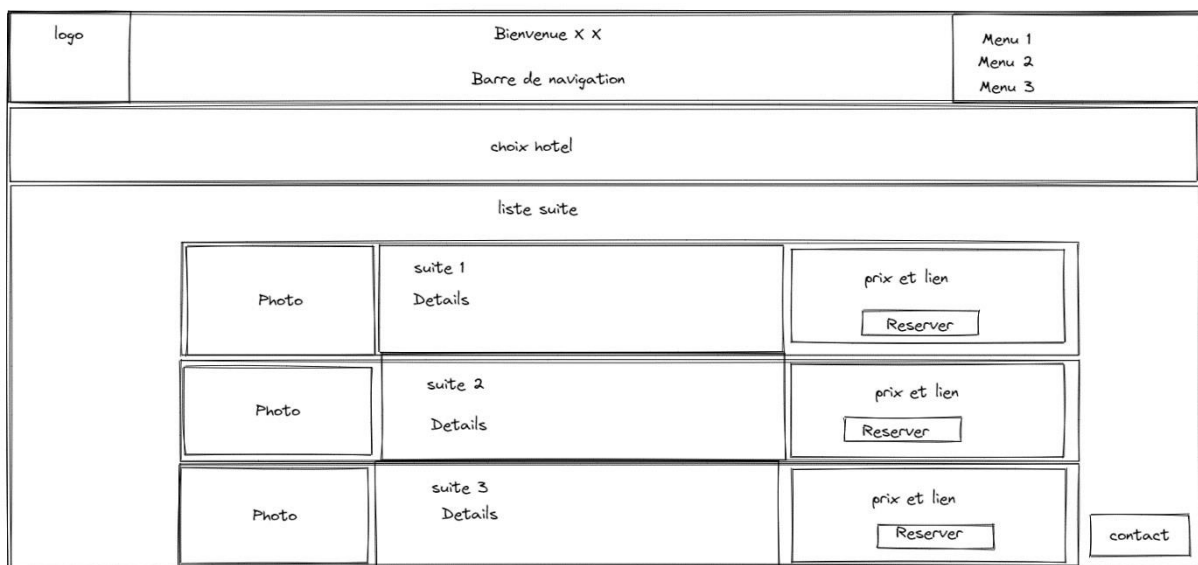
En effet, j'aime être maître de mon code, et quand on utilise un framework de ce type, à moins d'y avoir participé, il y a beaucoup de « surcouches » assez complexes à appréhender et qui nécessitent du temps. Je préfère connaître toutes les lignes de mon code et leurs fonctions.

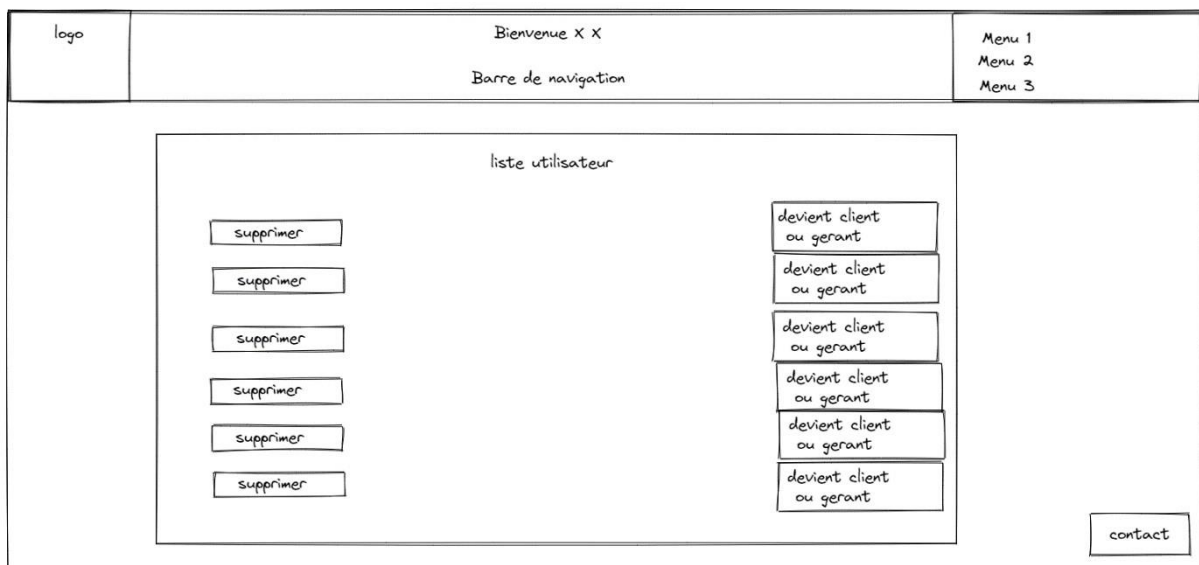
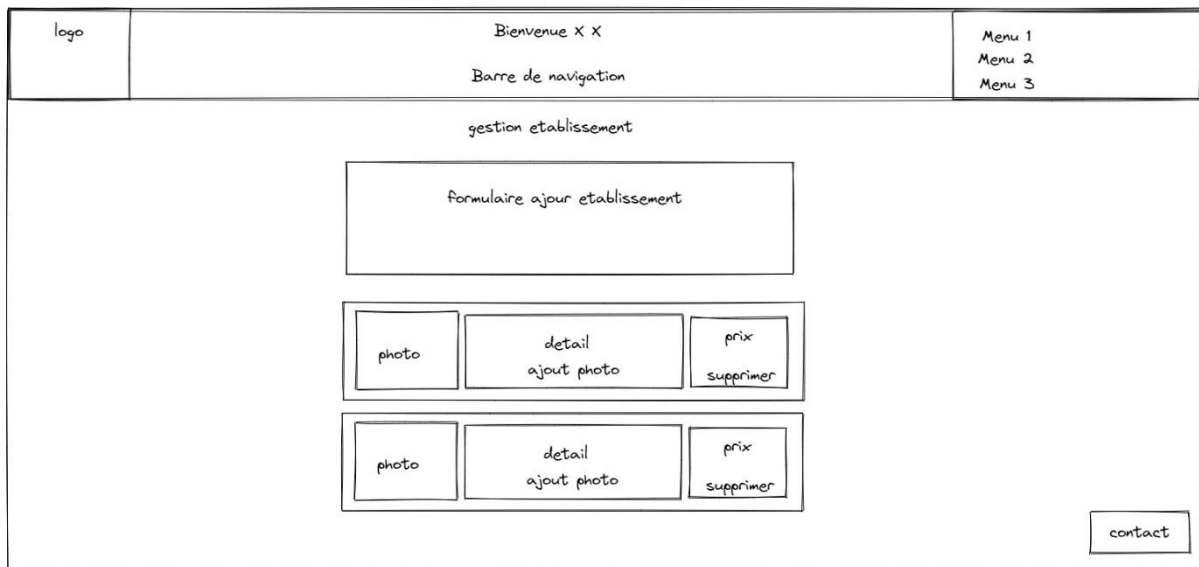
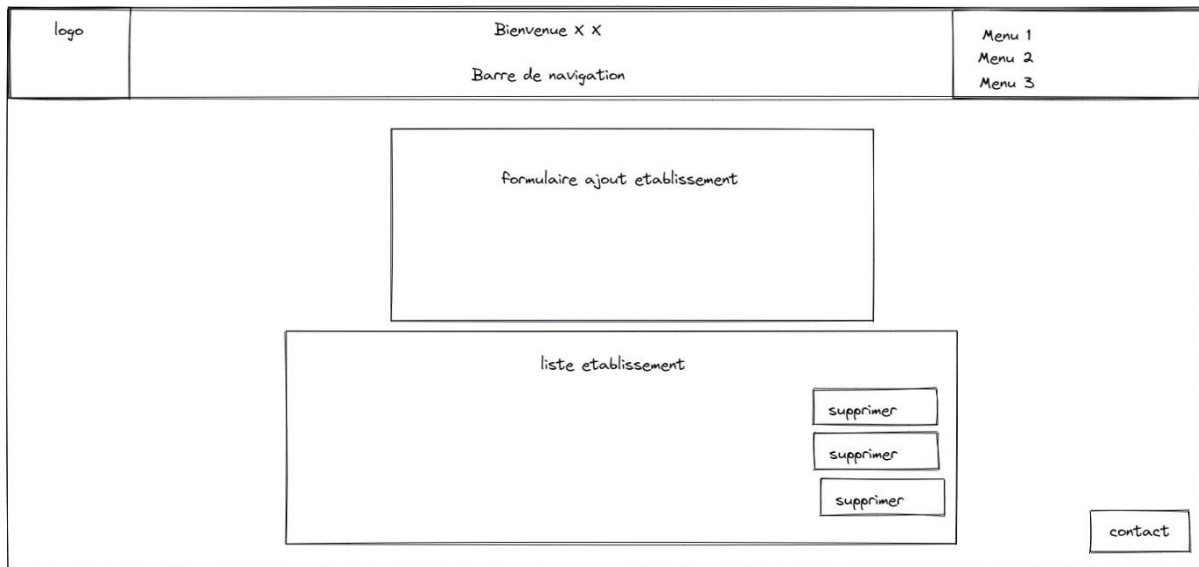
J'ai décidé d'utiliser du SASS pour styliser mon site. La possibilité de voir les modifications rapidement et la syntaxe me paraît plus claire.

J'ai utilisé un seul fichier style en dehors des « prérequis » plutôt que d'en faire un pour chaque page. En effet, de nombreux styles sont partagés et cela évite les redondances.

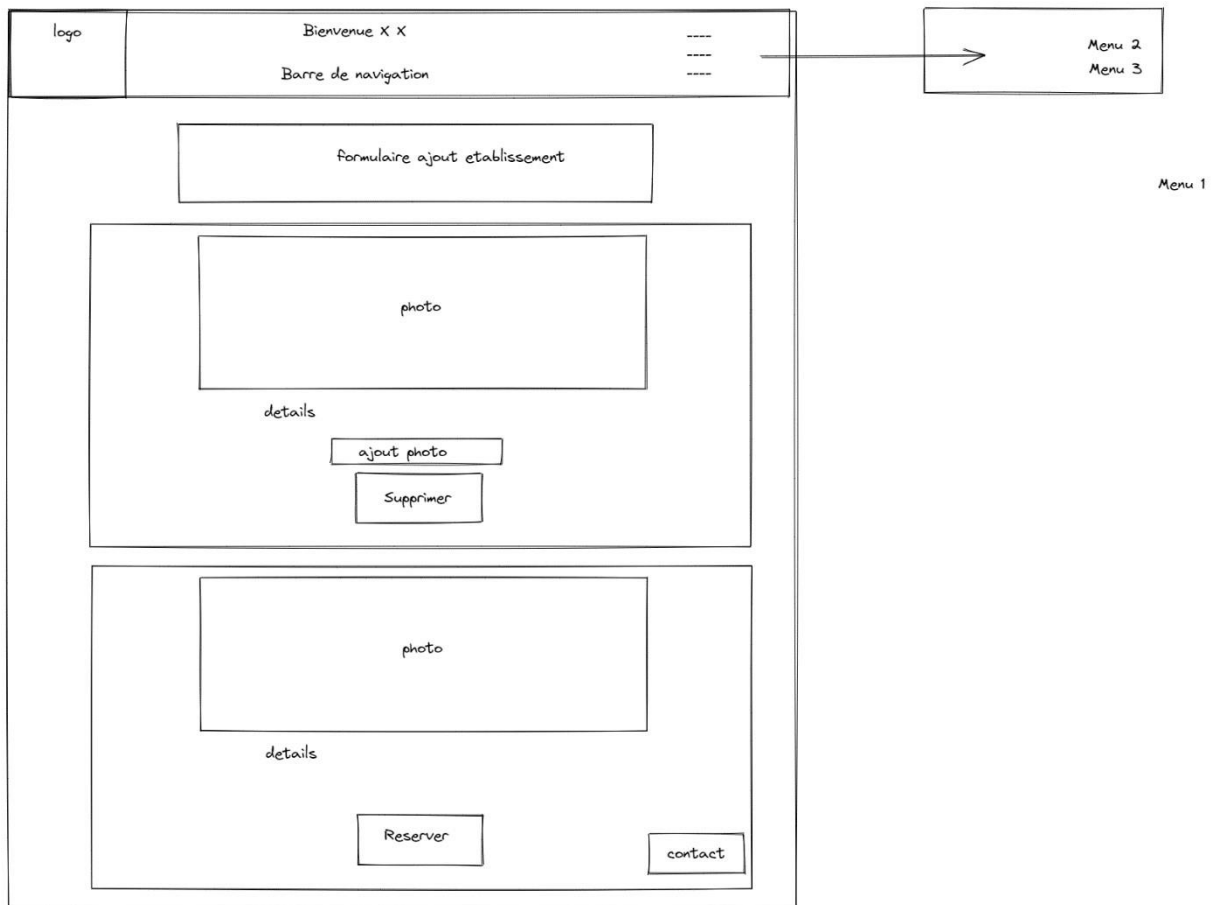
Une page d'accueil fait office de « hub » pour appeler chacune des autres pages sans temps de chargement. Chaque page a ses propres scripts PHP découpés, le tout fourni par ajax pour avoir des données dynamiquement.

J'ai fait plusieurs wireframes basiques pour avoir une idée de la direction dans laquelle je souhaitais aller mais j'ai dû également modifier pas mal de choses en cours de routes.



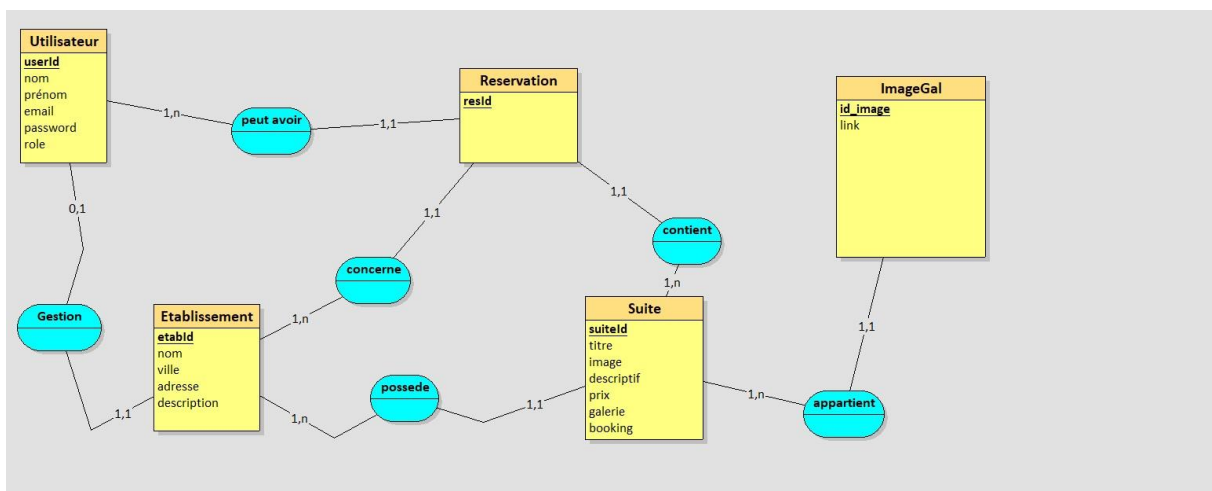


Je n'ai pas eu beaucoup de modification à faire pour la partir mobile grâce à Bootstrap entre autre.

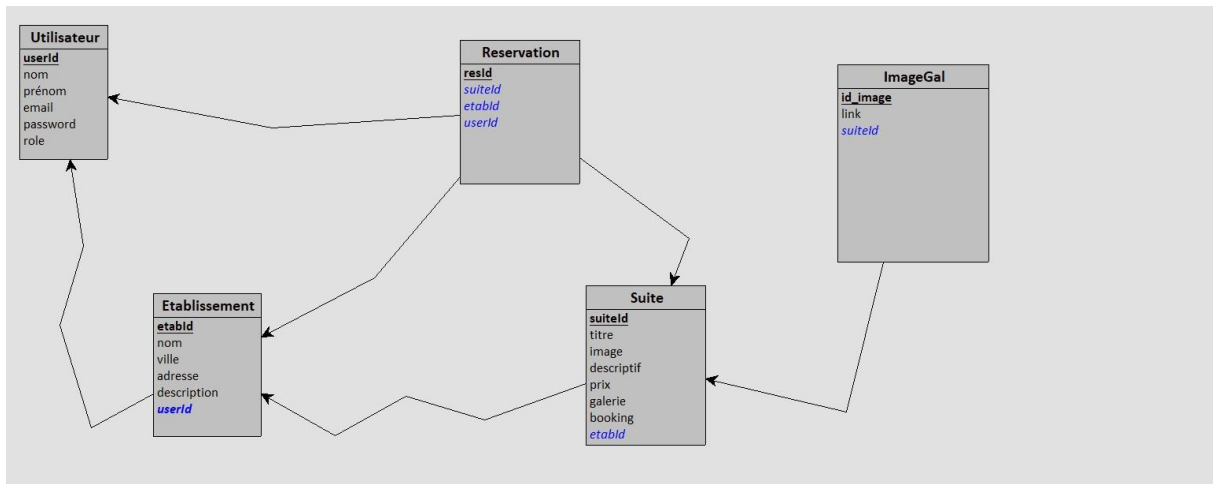


l'accueil apparait de la même façon sur écran réduit.

J'ai crée un schéma de la structure de ma base de données dans un premier temps sur papier, puis j'ai fait un diagramme avec looping :



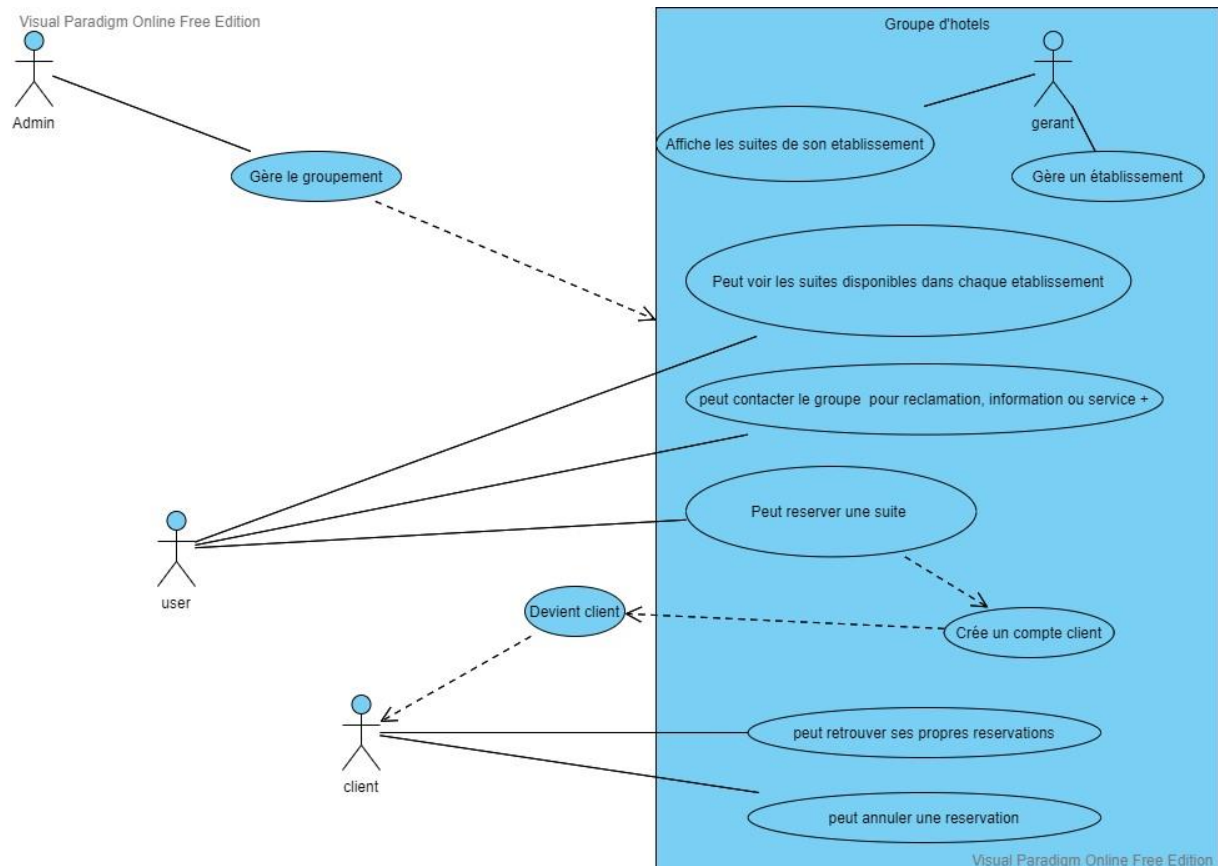
Que j'ai ensuite transformé en diagramme MLD pour voir les liaisons à effectuer ce qui m'a donné ceci :



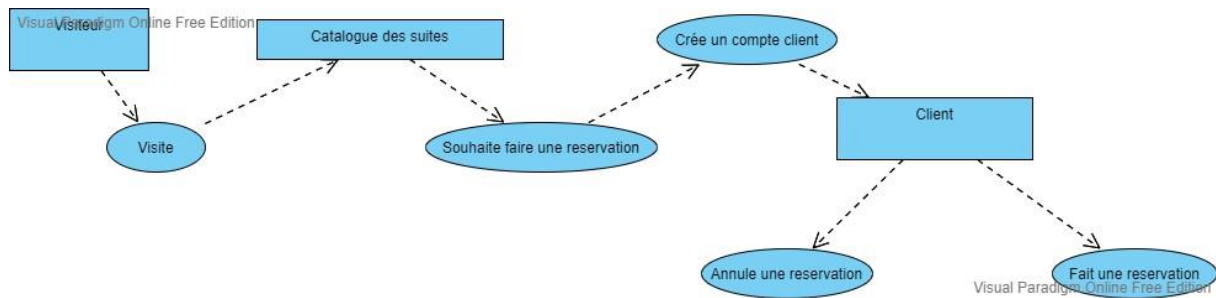
Et on constate qu'à ce moment-là, j'avais complètement omis le cas des formulaires de contacts.

La table réservation a été la plus complexe à mettre en place car elle est relié a tout le reste via des clefs étrangères.

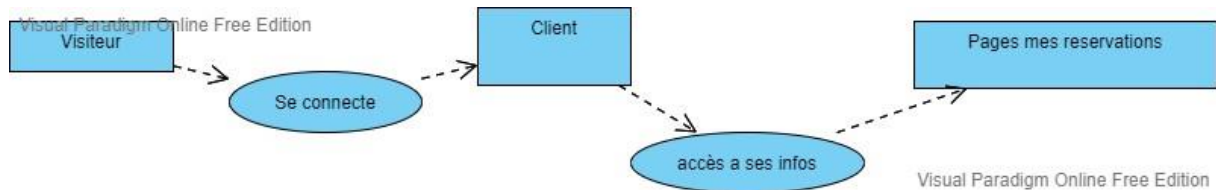
Enfin mon use case donne ceci :



La séquence pour passer de visiteur anonyme à client avec réservation :



Et enfin la séquence permettant a un client abonné de voir ses propres réservations et si nécessaire de les annuler :



J'ai opté pour MySQL car que c'est adaptable partout, et également pour la manipulation des données qui se fait de façon intuitive.

J'ai créé chaque table séparément de façon à pouvoir les remettre à zéro et les modifier si nécessaire.

Pour les photos, j'ai opté pour des liens url supposant que les photos serait hébergé sur un cloud type google drive