

NOM	BELLENGER
Prénom	PAUL
Date de naissance	01/07/1997

Copie à rendre

Graduate Développeur

(Android, Angular, Flutter, Front End, Full Stack, IOS, PHP/Symfony)

Documents à compléter et à rendre

Objectif & Livrables

Livrables :

Le lien du (ou des) dépôt(s) Github public, où sera présent le code de vos applications :

https://github.com/Johnny-fullstack/ECF_sept23_P.Bellenger.git

Le lien de la version en ligne de l'application web : <https://ecfquaientique.herokuapp.com/>

Le lien vers votre logiciel de gestion de projet pour cet ECF (Trello, Jira, etc...) :

<https://trello.com/invite/b/JM3eHwtu/>

ATTI4d58323dc8fcb836bc3b7f1b740f9bf4AC5FAF80/gestion-de-projet-ecf

Devra aussi se trouver:

- Un fichier readme.md contenant la démarche à suivre pour l'exécution en local
 - Ne pas oublier dans ce readme l'explication de la création d'un administrateur pour le back-office de l'application web
- Les fichiers de création et d'alimentation de la base de données de votre application web (migrations, fixtures ou scripts SQL à la main)
 - **Attention:** Générer du SQL par le biais d'une interface graphique (ex: PHPMYAdmin), et donc sans avoir à coder, conduira à la non-validation d'une des compétences du titre professionnel!
- Une documentation technique au format .pdf avec vos réflexions initiales sur le projet (technologies choisies, configuration de l'environnement de travail, diagramme de classe ou Méthode MERISE, diagramme de cas d'utilisation, diagramme de séquence...)
 - **Attention:** Chaque diagramme précisé dans l'annexe doit être obligatoirement présent! Ils sont en lien direct avec une des compétences du référentiel.
- Une charte graphique au format .pdf regroupant:
 - La palette de couleurs
 - La palette des polices d'écriture choisies
 - L'export des maquettes attendues
 - wireframe et/ou mockup pour la version mobile ET desktop

Attention! Pour respecter les contraintes de temps et valider toutes les compétences du référentiel, il est vivement conseillé de ne pas passer trop de temps sur le design de vos applications. S'il est minimaliste, mais cohérent, alors c'est suffisant.

Réflexions :

Pour commencer, j'ai créé les fichiers de bases (charte graphique, fiche technique) et le Trello, pour m'organiser correctement. Après avoir réalisé les wireframes et les mockups ainsi que les diagrammes d'après les indications de l'énoncé j'ai commencé le front. Entre les maquettes figma et le front j'ai un peu modifier le styles des éléments du main (les photos, la carte et les formulaires) car j'ai trouvé que ça rendait mieux comme cela. J'ai mis environ 2 semaines à réaliser le front-end du site.

Après cela, j'ai installé mon application symfony dans XAMPP, installé les extensions nécessaires, crée un répertoire git et est importé mon front en le convertissant en Twig et en créant les controllers et les routes. Ensuite j'ai fait la base de données avec Doctrine ainsi que les entités. Puis, j'ai fait mes formulaires en php avant de créer leur controller. J'ai eu beaucoup de problème à cette étape, car j'ai fait beaucoup d'erreur de syntaxe et je n'ai pas compris tout de suite comment associer ma base de données à mon projet.

Une fois tous cela terminé, j'ai pu m'occuper du reste des fonctionnalités (Ajax des places disponibles, profil User, profil Admin) et corriger les erreurs restantes. J'ai réussi à créer un profil User avec le pré-remplissage du formulaire de réservation, mais je n'ai pas réussi à réaliser les autres fonctionnalités :

- Lorsque je remplis le formulaire de réservation j'ai un message d'erreur me disant que les champs des heures attendent un type Dateinterface alors qu'ils renvoient un string. Je voulais qu'ils renvoient un string et j'ai bien déclaré dans l'entité qu'il devait recevoir un string et malgré tout l'erreur persiste.
- Pour le formulaire de connexion j'ai donc créé un profil avec le formulaire d'inscription dont je me suis servi pour tester la connexion. J'ai beau rentrer le bon mail et le bon mot de passe, ça me renvoyait constamment identifiants incorrects.
- Pour la requête Ajax du nombre de place restante pour le formulaire de réservation, j'ai tout juste intégré React.js dans l'application mais comme j'avais perdu trop de temps sur les formulaires il me restait 1 jour avant le rendu final donc j'ai préféré me concentrer sur la mise en ligne du projet.
- Cependant, je n'ai pas non plus réussi la mise en ligne sur Heroku, j'ai pourtant suivi les instructions du site qui indique comment mettre en ligne une application symfony mais au moment du push, j'ai eu un premier message d'erreur m'indiquant que le bundle-twig était introuvable. Il n'apparaissait pas dans composer.json bien que lorsque je vérifiais via le shell il était installé. J'ai donc réinstallé bundle-twig mais après cela j'ai eu le même message mais pour MonologBundle, puis de nombreuses autres erreurs jusqu'à ce que je réussisse à afficher la page d'accueil. Mais les routes vers les autres pages ne fonctionnent pas alors qu'en local ça fonctionne.

Méthode :

- Comprendre le projet
- Faire la charte graphique
- Faire les maquettes (mockups et diagrammes)
- Commencer le front en le pensant pour la version portable. D'abord faire l'html. Ensuite, réalisation du header et du footer qui seront copier/coller sur les autres pages une fois leur

css fait. Ensuite, réalisation du main de chaque page, et du css dans des fichiers différents de celui du header et du footer.

- Pour le Back-end, j'ai fait du général au particulier. J'ai commencé par le plus indispensable pour aller vers les fonctions les plus spécifiques du site. J'ai donc commencé par créer les routes et la base de données, ensuite les formulaires et les entités, pour finir avec le profil utilisateur et Admin et la requête asynchrone vérifiant le nombre de places disponibles.
- Le Trello ainsi que le git était mise à jour à chaque fin de journée ou à chaque passage d'étapes importantes.

Environnement de travail :

- Trello
- Figma

Outils :

Serveur :

- XAMPP en local
- Heroku en public

Front :

- HTML5
- CSS3
- Javascript
- JQuery.js
- React.js (prévu mais pas utilisé)

Back :

- PHP 8.1
- Symfony 5.4
- Apache
- Phpmyadmin/MySQL

UML

Diagramme de classes :

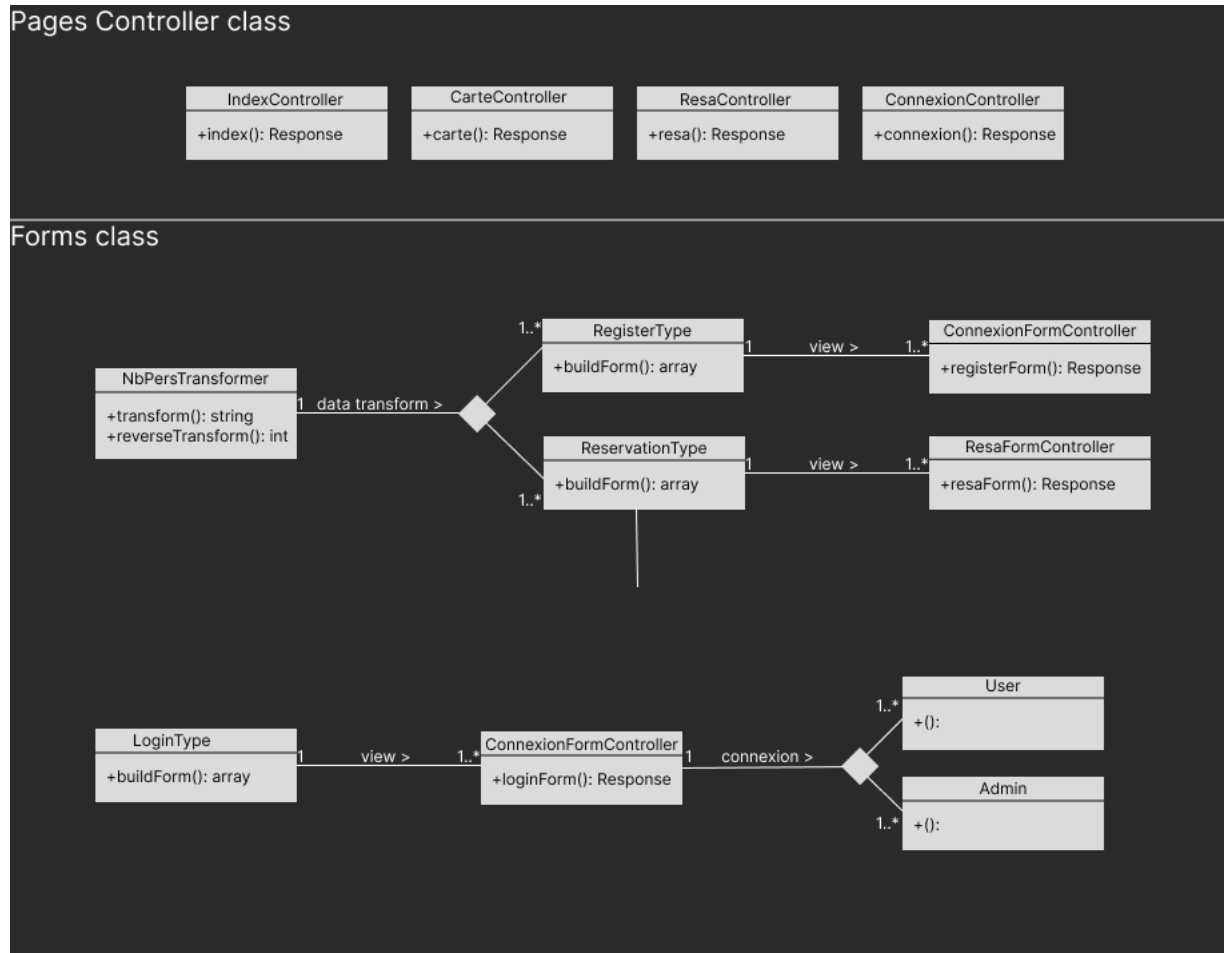
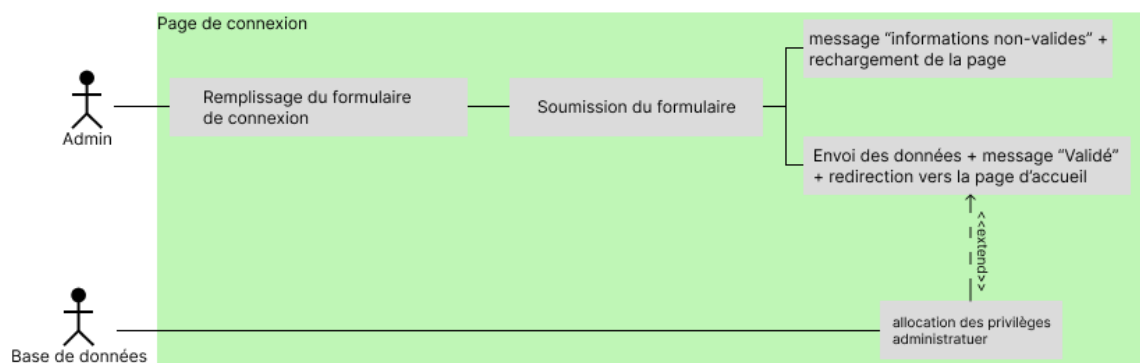
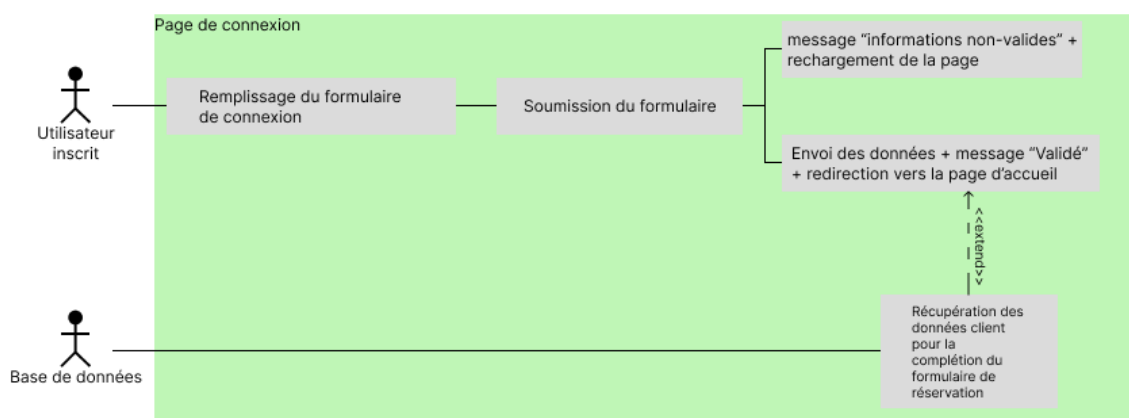
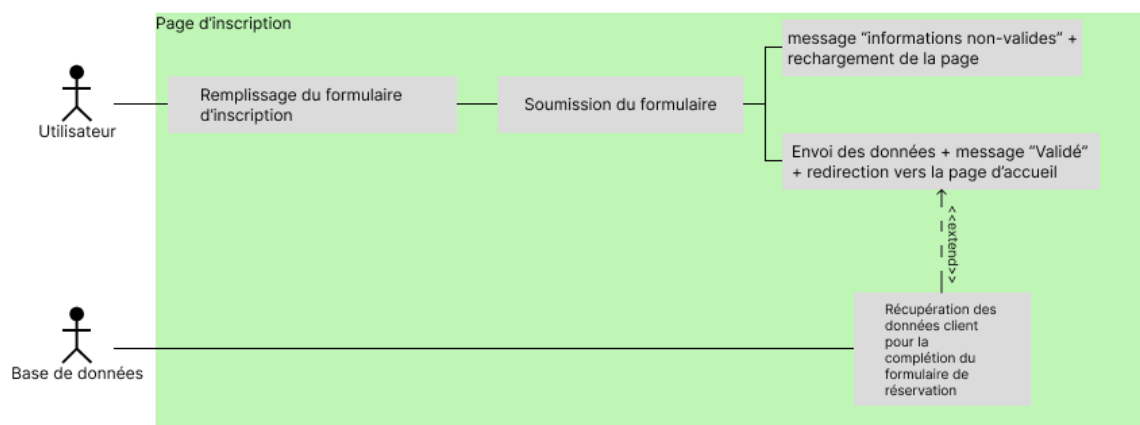


Diagramme de cas d'utilisation :



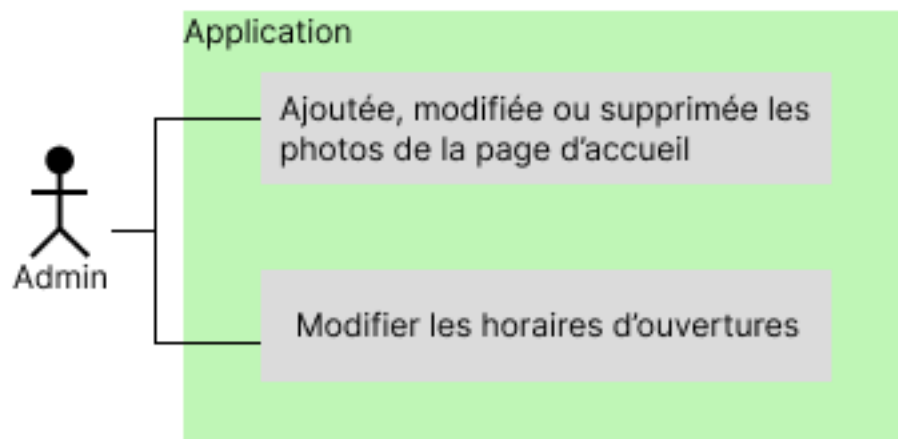
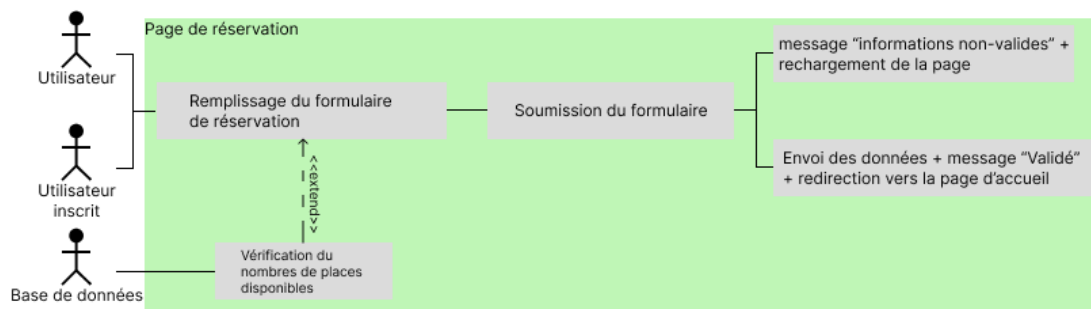
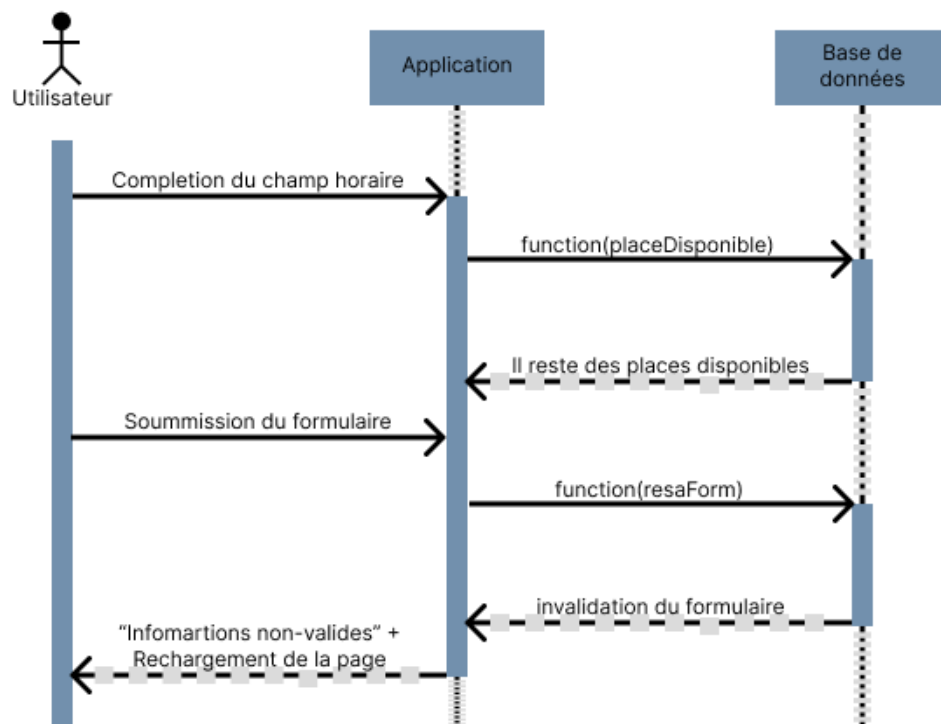
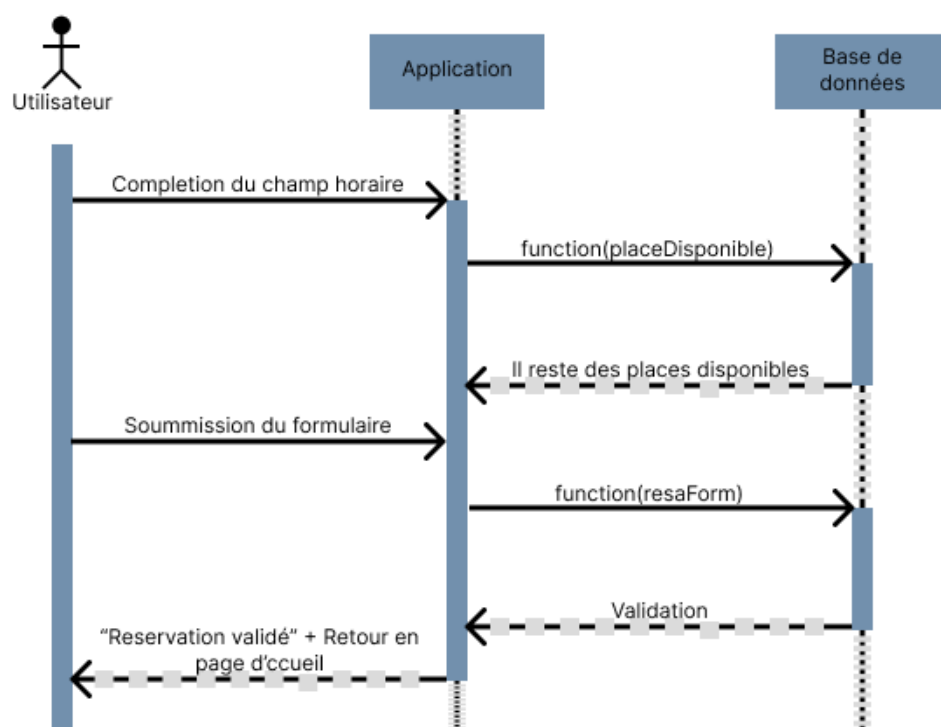


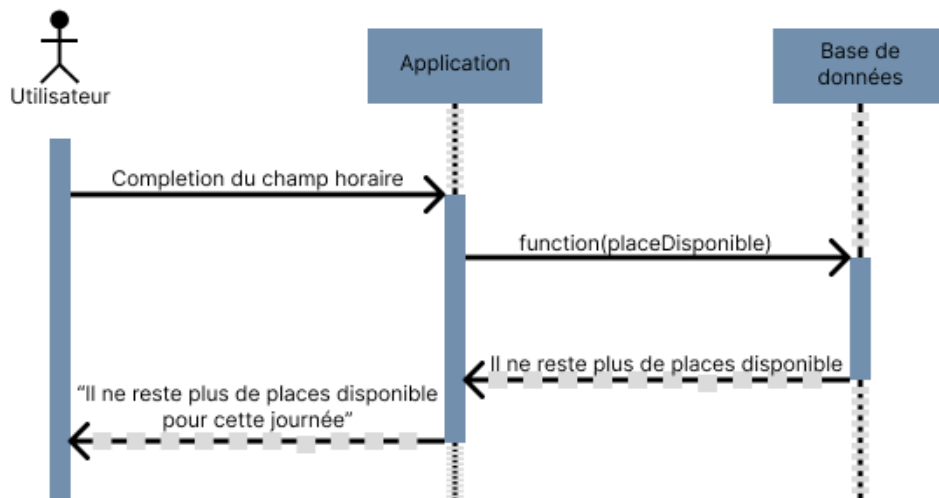
Diagramme de séquence :

Séquence de réservation non-validé



Séquence de réservation validé





CHARTRE GRAPHIQUE

Inspiration :

<https://www.alixetmika.com/>

<https://www.iletaitunsquare.com/accueil.html>

<https://instinct-paris.com/>

<https://www.paris-restaurant-pur.fr/>

<https://www.legrandcorona.fr/>

<https://www.gosier-paris.fr/fr/>

<https://www.mamie-lise.com/restaurant-annecy/#ambiance>

Logo :



Palette de couleurs :

Header/Footer : #392C1E

Nav bar/Footer title : #906427

Elément background : #B6AC97

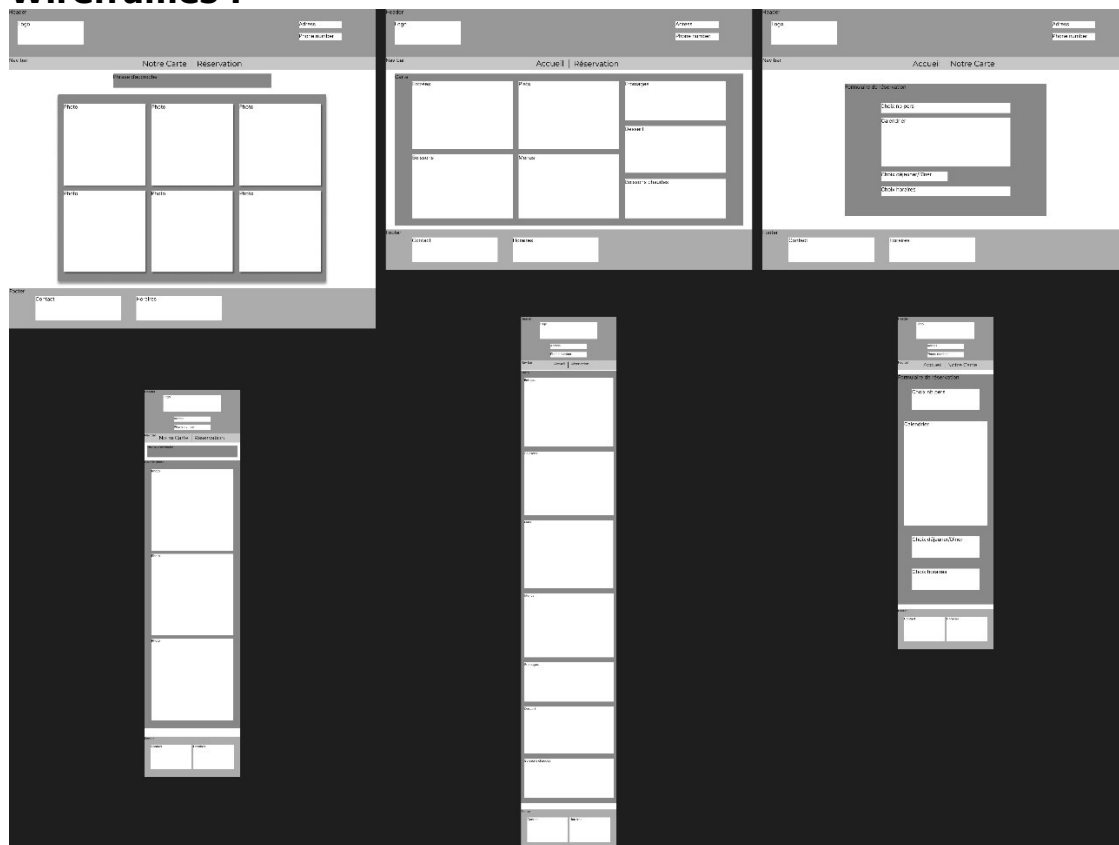
Bordures et couleurs de boutons : #faebd7

Polices d'écriture :

Montserrat: Text d'information (contact, horaires, carte, formulaire, navigation)

Playfair Display SC: Text de logo et titres/Slogan

Wireframes :



Mockups :

