**Documentation**

**Lien git** : <https://github.com/Badrk7/projetb2>

**Prérequis** :

-npm install ou npm install --force

-npm run dev

**Technologies utilisées** : next js , mongo db et tailwind css (avec une template quasiment entièrement en tailwind)

**Structures des pages** :

*Home page* : liste des lieux créés et filtre en cochant une ou plusieurs catégories et/ou en écrivant un nom de pays et possibilité de modifier, supprimer et voir les détails

*Add-page* : sélection des informations obligatoires qui changent en fonction des catégories et l’affiche dans la home page avec une image en fonction de sa catégorie

*Single-place* : page de détail en cliquant sur un des lieux de la home page et affichage de toutes les informations du lieu

*Update-place* : la page de modification avec les informations remplies précédemment et l’image change si la catégorie change aussi

**Components** :

*404* : la page pour les liens du footer

*Card* : pour récupérer les images des catégories

*Dropdown* : pour la sélection des informations obligatoires dans add et edit

*Footer* : actif dans toutes les pages du site et les liens renvoient vers la page du component 404 et les icones vers les réseaux sociaux

*Header* : actif dans toutes les pages du site pour naviguer entre la home page et la add-page

*Layout* : utilisé dans chaque component c’est lui qui contient le header et le footer et est appelé dans toutes les pages

*Logo* : pour son paramétrage et affichage dans le header et le footer

*Modal* : pour la suppression d’un lieu avec l’alerte le bouton cancel et delete

*Place-detail* : pour afficher l’image et les informations complètes en fonction du lieu sélectionné

*Place-form* : le formulaire des informations des lieux utilisé dans add-place et update-place

*Side-bar-search-filter :* la barre de en fonction du pays et des catégories cochées (plusieurs possibles en même temps)

*Social* : les icones des réseaux sociaux et leur lien dans le footer

**Api**: Toutes les requêtes qui interagissent avec la bdd pour les actions du CRUD

**Services** : Les fonctions qui utilisent l’api mongo db pour envoyer et récupérer les informations de la bdd

**Problèmes rencontrés** :

Les images étaient au choix parmi une sélection mais inutile et pas de sens si le choix est limité, donc sélection de l’image forcé en fonction de sa catégorie.

Connexion avec la bdd sur mongo atlas qui n’arrivait pas à récupérer les données lors de l’ajout d’un lieu à cause de la configuration incomplète sur mongo atlas

**Axes d’amélioration** :

Page d’identification et ajout de droit pour certains comptes, pour avoir un délai d’attente pour que l’admin puisse valider ou non la publication du lieu et ses informations

Choix d’image par l’utilisateur par fichier ou par recherche dynamique si aucune image n’est importée par l’utilisateur

Connecter la bdd de mongo atlas (celle utilisée avec le local) avec [le site](https://badrok-web-app.netlify.app/) avec le host netifly pour que les changements soient actifs et visibles sur le site et en local

Améliorer a recherche pour pouvoir utiliser les noms des rue régions départements etc…

Utiliser différentes monnaies et les convertir en fonction du choix de l’utilisateur