

DOCUMENTATION TECHNIQUE

A. SPECIFICATIONS TECHNIQUES

Pour le front-end :

Le choix s'est porté sur React Js afin de répondre au besoin du site en SPA (Single Page Application). Cela permet d'avoir un site où la navigation est très rapide.

Avec ce framework, j'ai utilisé les bibliothèques suivantes :

- react-router-dom : afin d'avoir un routing maîtrisé côté navigateur
- Axios : pour simplifier les requêtes ajax pour l'échange de données avec le serveur
- Material-ui : pour avoir des icônes reconnues du grand public (login, logout, supprimer...)
- Match-sorter : pour avoir une recherche dynamique de données
- React-modal : pour simplifier l'utilisation de modal

Pour le back-end :

Le choix s'est porté sur du Node.js avec l'utilisation du framework Express. Il y a un gros avantage d'utiliser le même langage de programmation côté client et côté serveur. Cela permet de simplifier les échanges sans jongler en deux langages différents. La rapidité de Node et son système non-bloquant semble tout à fait adapté à ce projet à fort trafic potentiel. Les bibliothèques suivantes ont été utilisées :

- sequelize : ORM permettant de simplifier les requêtes avec la base de données SQL
- bcryptjs : pour hasher les mots de passe et de ne pas les avoir en « clair » dans la base de données (sécurité first!!!)
- jsonwebtoken : pour simplifier la manipulation des tokens d'identification
- nodemailer : pour l'envoi de mail via le serveur.

Pour le déploiement :

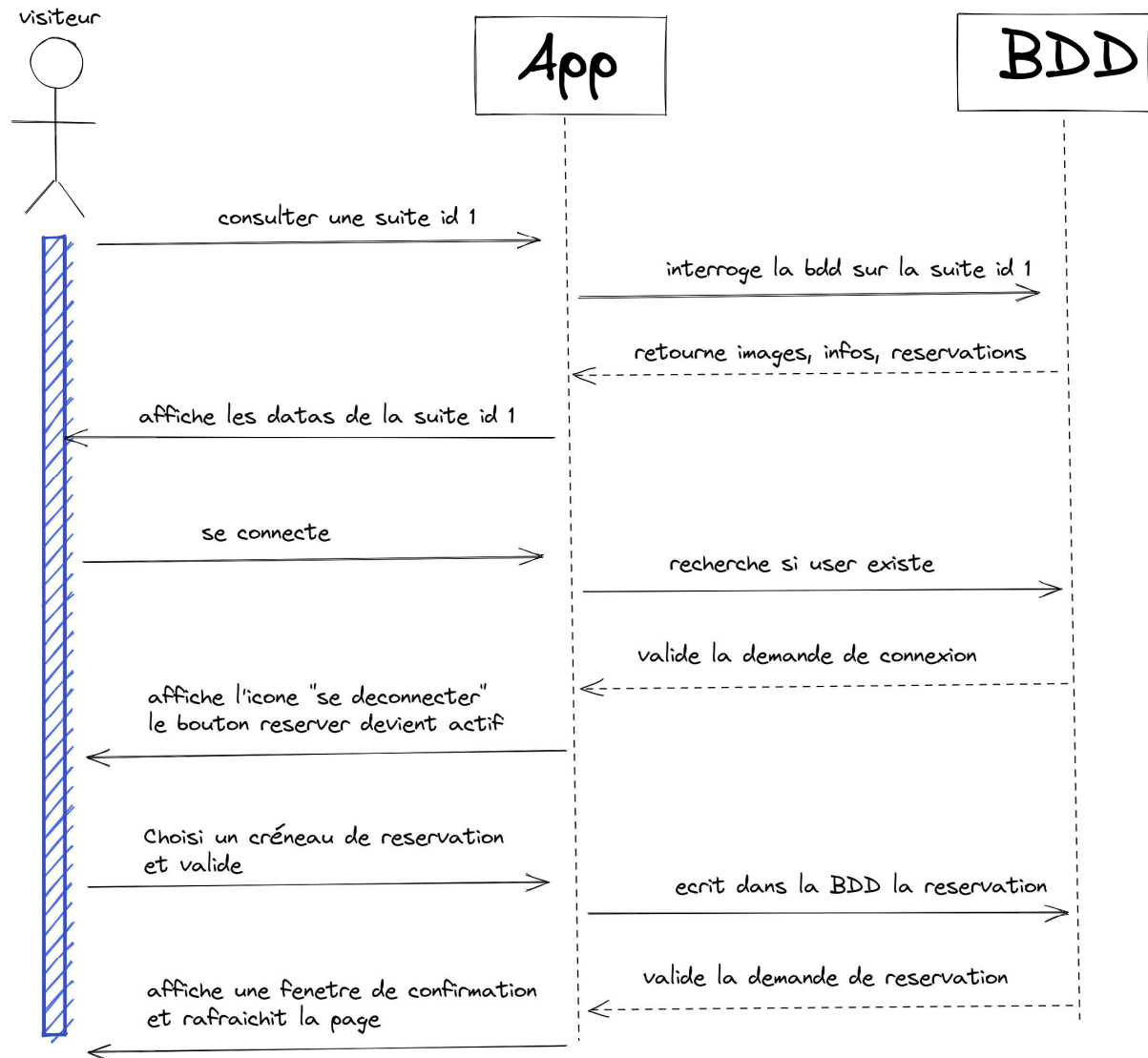
Possédant un serveur chez l'hébergeur PLANETHOSTER, j'en ai profité pour y héberger ce site.

B. DIAGRAMMES

CAS D'UTILISATIONS :

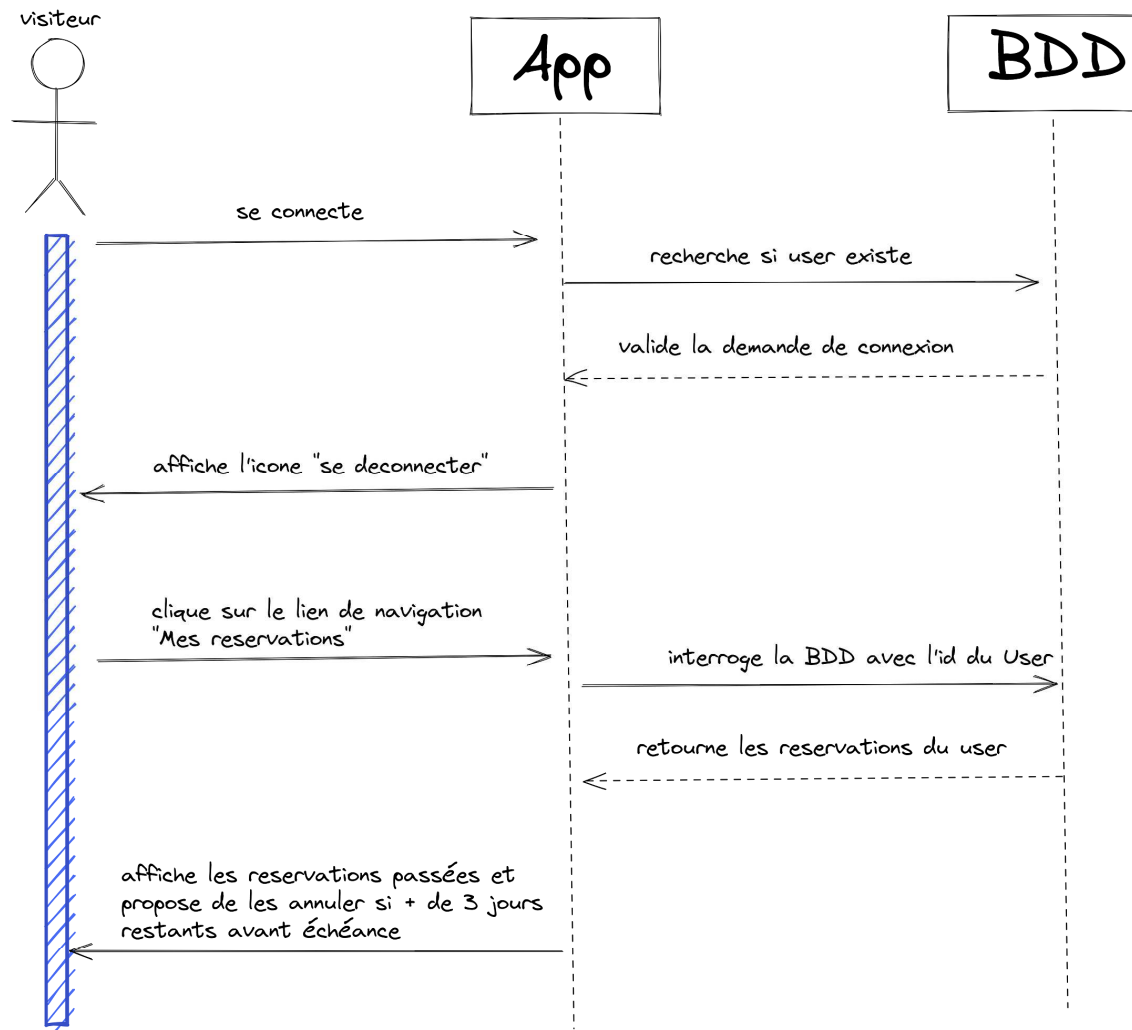


DIAGRAMMES DE SEQUENCES :
DS.1 :



RESERVER UNE SUITE

DS.2 :



VOIR SES RESERVATIONS

DIAGRAMME DE CLASSE :

