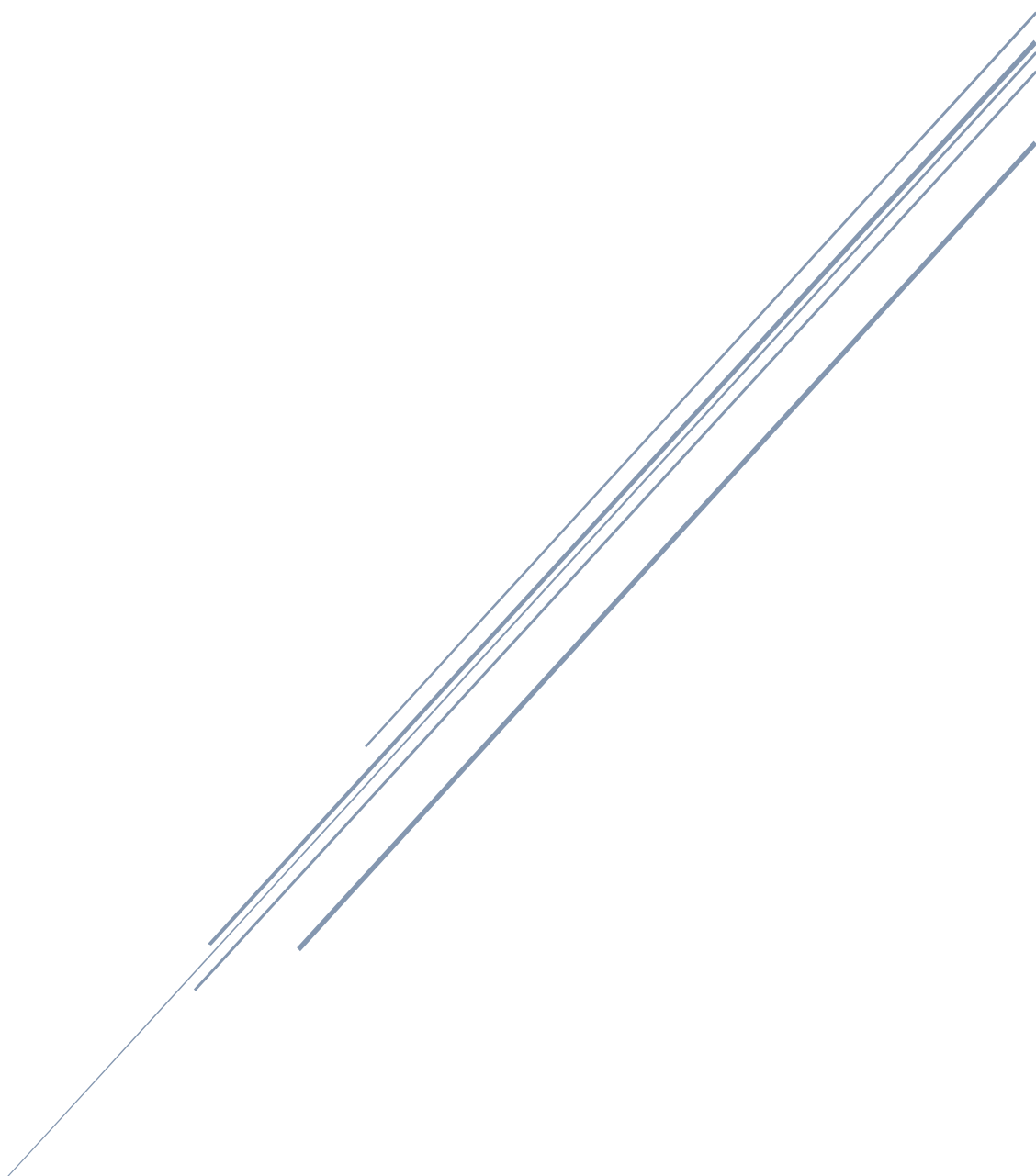


RAPPORT DE PROJET

CO'WORK



Sujet de rattrapage 2019 - 2020

Table des matières

Introduction	3
Analyse des applications	4
API REST	4
Auth.....	4
User	4
Building.....	4
Room	4
Subscription.....	4
Reservation	5
Application Front Office	5
Application Back Office	5
Modèle conceptuel des données	6
Lancement du projet	6
Bilan du projet.....	6

Introduction

Co'Work est une entreprise qui propose six espaces de travail collaboratif à Paris (Bastille, République, Odéon, Place d'Italie, Ternes et Beaubourg).

L'objectif est de développer des outils pour :

- Gérer les adhésions des clients (informations générales, identification, ...)
- Gérer le système de réservation des espaces privés, de prêts de matériels, de commandes de plateaux repas

Le projet s'articule :

- D'une API REST servant à l'adhésion des clients et du système de réservations.
- D'une application Web pour les clients
- D'une application BackOffice utilisé par Co'Work

Notre groupe pour ce projet se compose de :

- Théo Gouraud
- Jean-Luc Lys-Phorimavong

Analyse des applications

API REST

L'API est en NodeJS. La base de données a été générée avec le module Sequelize.

Voici la liste des différents points d'accès de l'API :

Auth

Name	Methods	Endpoints	Params
Inscription	POST	/api/auth/signup	email, password, firstName, lastName, buildingId, subscriptionId
Connexion	POST	/api/auth/signin	email, password

User

Name	Methods	Endpoints	Params
Tous les utilisateurs	GET	/api/users	
Utilisateur par email	GET	/api/users	email
Utilisateur par id	GET	/api/users/:id	Id passé dans l'URL

Building

Name	Methods	Endpoints	Params
Tous les bâtiments	GET	/api/buildings	
Un bâtiment par son ID	GET	/api/buildings/:id	Id passé dans l'URL
Un bâtiment	GET	/api/buildings	name
Id du bâtiment	GET	/api/buildings/id	name

Room

Name	Methods	Endpoints	Params
Ajout d'une salle	POST	/api/rooms	name, type, buildingId
Toutes les salles selon l'id d'un bâtiment	GET	/api/rooms	buildingId
Une salle d'un bâtiment	GET	/api/rooms	name, buildingId

Subscription

Name	Methods	Endpoints	Params
Tous les abonnements	GET	/api/subscriptions	
Abonnement par id	GET	/api/subscriptions/:id	Id passé dans l'URL

Reservation

Name	Methods	Endpoints	Params
Réserver une salle	POST	/api/reservations	startDate, endDate, participants, totalPrice, userEmail, roomName, buildingName
Toutes les réservations	GET	/api/reservations	
Réservations d'une salle	GET	/api/rooms/reservations	roomId
Réservations d'un utilisateur	GET	/api/users/reservations	userId
Réservation selon l'ID	GET	/api/reservations/:id	Id passé dans l'URL

Application Front Office

Application web (Angular) destinée aux clients :

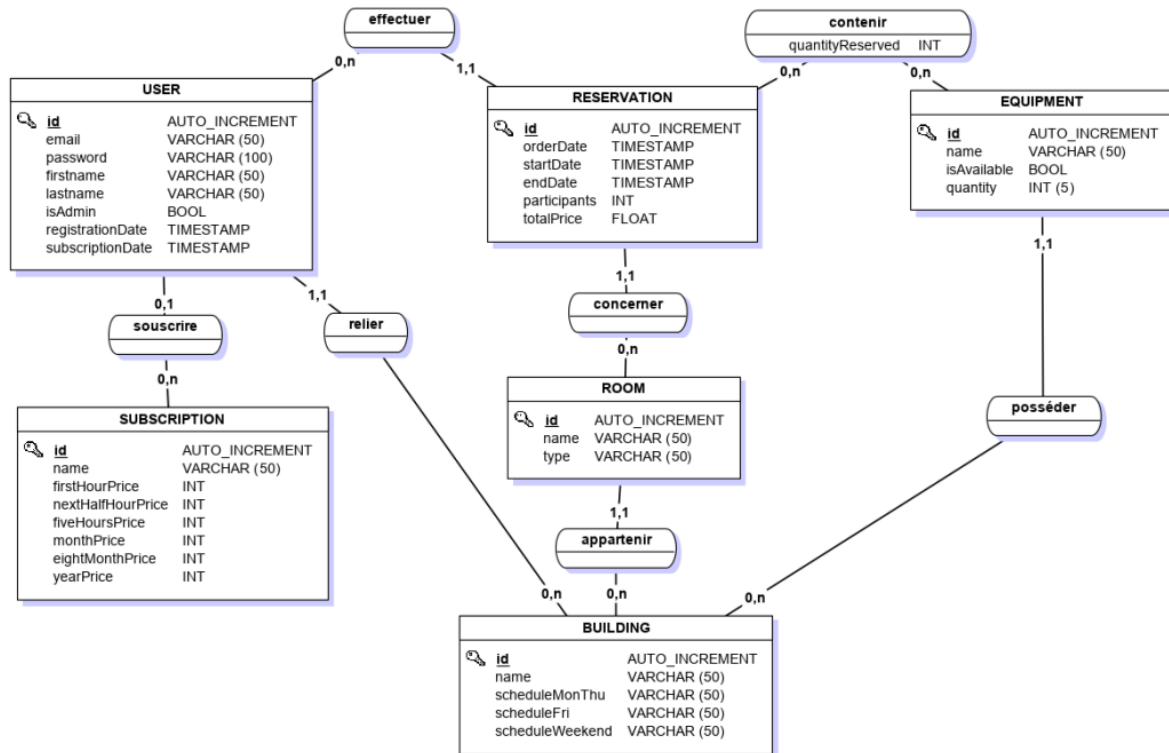
- Inscription d'un nouveau client
- Connexion d'un utilisateur
- Réservation d'une salle selon le bâtiment

Application Back Office

Application web destinée à Co'Work :

- Connexion administrateur
- Administration des données (réservations, services...)

Modèle conceptuel des données



Lancement du projet

Après avoir récupéré le projet :

1. Démarrer l'API avec la commande « nodemon »
2. Ouvrir la base de données pour entrer manuellement le « script.sql » dans la base de données.
3. Lancer l'application front office.

Bilan du projet

Ce projet nous a apporté techniquement. Il nous a permis de consolider nos connaissances concernant les API ou les différentes difficultés rencontrées lors du développement de l'application web. Nous avons mis du temps pour résoudre les différents problèmes tels que les problèmes d'accès à l'API via différentes origines ou bien le surplus d'informations.

Au niveau des difficultés humaines, il est très compliqué de faire un projet sans se voir. Nous avons dû parfois travailler de notre côté et travailler souvent en appel, ce qui n'est pas souvent pratique quand on veut expliquer une chose précise.

L'ensemble du projet n'a pas pu être livré en raison d'un manque de temps.