



Rapport Tâche 5 : Projet 2 - App JSON Share avec Prisma et Clerk

Firas Ben Hmida
Encadré par NextGen Coding

01/08/2025

Table des matières

1	App JSON Share	2
1.1	Lien YouTube du projet	2
1.2	Auteur	2
1.3	Technologies utilisées	2
1.3.1	Next.js	2
1.3.2	Shadcn UI	2
1.3.3	Clerk (authentification)	3
1.3.4	Prisma ORM	3
1.3.5	PostgreSQL	3
1.4	Description du projet	3
1.5	Démo - Captures d'écran	4
1.6	Lien GitHub du projet	5
2	Déploiement sur Vercel avec Neon	6
2.1	Présentation de Vercel	6
2.2	Base de données Neon	6
2.3	Étapes de déploiement	6
2.4	Démo - Captures d'écran	6
2.5	Lien Vercel du projet	7

Chapitre 1

App JSON Share

1.1 Lien YouTube du projet

Lien : <https://www.youtube.com/watch?v=H0GUJV5a8Wg>

1.2 Auteur

Cand Dev

1.3 Technologies utilisées

1.3.1 Next.js

Next.js est un framework React qui permet de créer des applications web performantes avec génération statique et dynamique des pages. Il permet de créer des pages web côté client et côté serveur. Dans ce projet, Next.js a permis :

- la création d’une interface utilisateur réactive,
- la gestion des routes et des pages,
- le déploiement rapide et efficace sur Vercel.

1.3.2 Shadcn UI

Shadcn UI est une bibliothèque de composants React basée sur Tailwind CSS, conçue pour créer des interfaces utilisateur modernes, accessibles et personnalisables. Contrairement à d’autres bibliothèques prêtes à l’emploi, Shadcn UI offre un contrôle total sur les styles, tout en fournissant une structure cohérente et professionnelle.

Dans ce projet, Shadcn UI a permis :

- de créer rapidement une interface cohérente et esthétique,
- d’utiliser des composants réutilisables (boutons, formulaires, boîtes de dialogue, etc.),
- de maintenir un style homogène grâce à Tailwind CSS,
- d’améliorer l’expérience utilisateur tout en conservant une grande flexibilité sur le design.

1.3.3 Clerk (authentication)

Clerk fournit une solution complète d'authentification et de gestion des utilisateurs. Il propose une interface intégrée et sécurisée pour gérer les connexions. Dans ce projet, Clerk permet :

- la création de comptes utilisateurs,
- la connexion sécurisée,
- la gestion des sessions (authentification persistante),
- la gestion complète des profils utilisateurs (création, suppression, mise à jour),
- l'intégration facile avec l'interface Next.js.

1.3.4 Prisma ORM

Prisma est un ORM (Object Relational Mapper) pour TypeScript qui simplifie l'interaction avec une base de données relationnelle. Dans ce projet :

- il facilite les opérations CRUD (Create, Read, Update, Delete),
- il génère automatiquement les schémas de base de données,
- il intègre facilement avec PostgreSQL.

1.3.5 PostgreSQL

PostgreSQL est un système de gestion de base de données relationnelle open-source. Il permet :

- un stockage fiable des données utilisateurs et JSON,
- une compatibilité directe avec Prisma ORM,
- une performance optimale dans les requêtes complexes.

1.4 Description du projet

Création d'une application web permettant aux utilisateurs de se connecter, créer, sauvegarder et partager des données au format JSON. Introduction à Prisma comme ORM, avec intégration complète de Clerk pour la gestion des comptes utilisateurs.

1.5 Démo - Captures d'écran

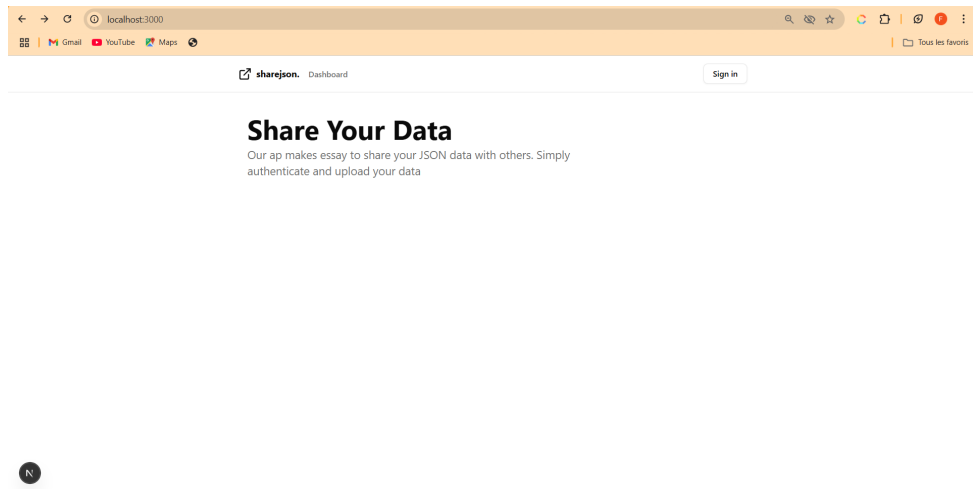


FIGURE 1.1 – Interface principale de l'application JSON Share

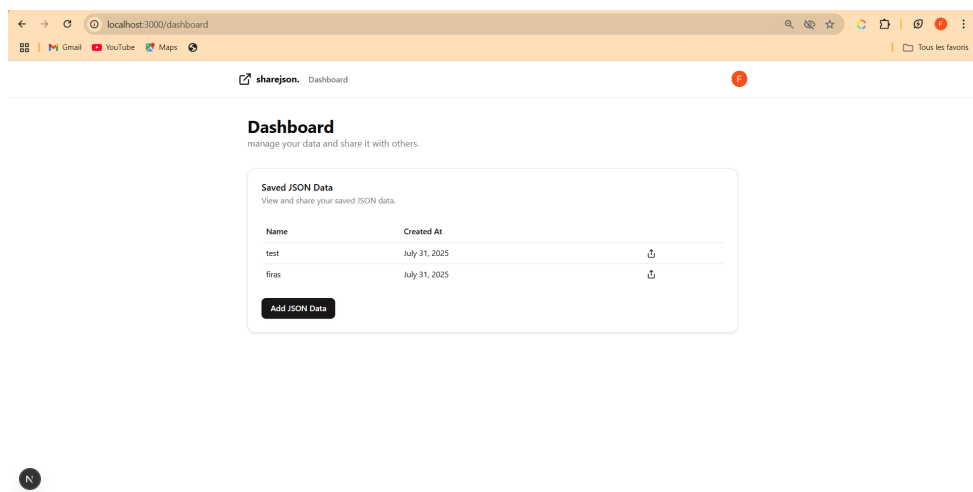


FIGURE 1.2 – Dashboard

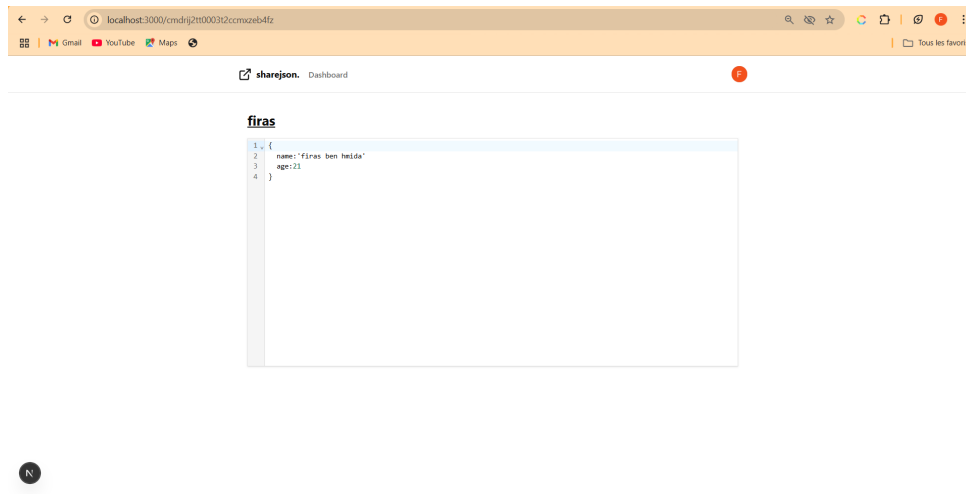


FIGURE 1.3 – partage de données JSON

1.6 Lien GitHub du projet

GitHub : <https://github.com/firas-ben-hmida/projet-json-share>

Chapitre 2

Déploiement sur Vercel avec Neon

2.1 Présentation de Vercel

Vercel est une plateforme de déploiement optimisée pour les projets Next.js.

2.2 Base de données Neon

Neon est une base PostgreSQL hébergée dans le cloud, idéale pour les projets modernes. Elle offre des fonctionnalités comme le scaling automatique, les branches de base de données, et une intégration simple avec Prisma.

2.3 Étapes de déploiement

- Connexion de Neon à Prisma via l'URL d'environnement.
- Déploiement automatique sur Vercel via GitHub.
- Configuration des variables d'environnement (clé Clerk, URL Neon...).

2.4 Démo - Captures d'écran

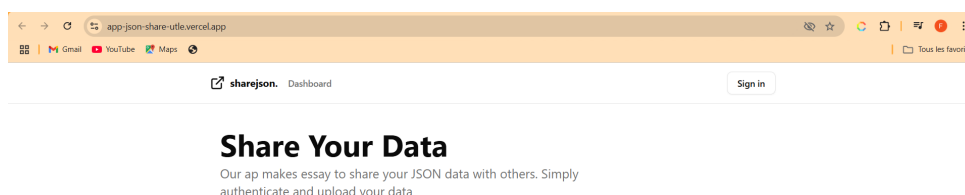


FIGURE 2.1 – Interface principale de l'application JSON Share

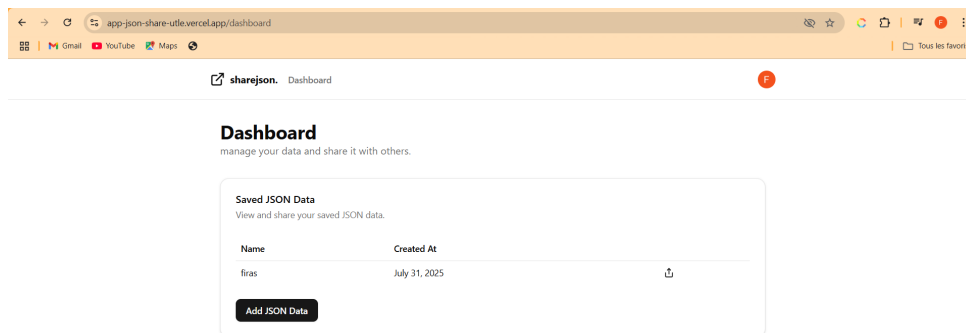


FIGURE 2.2 – Dashboard

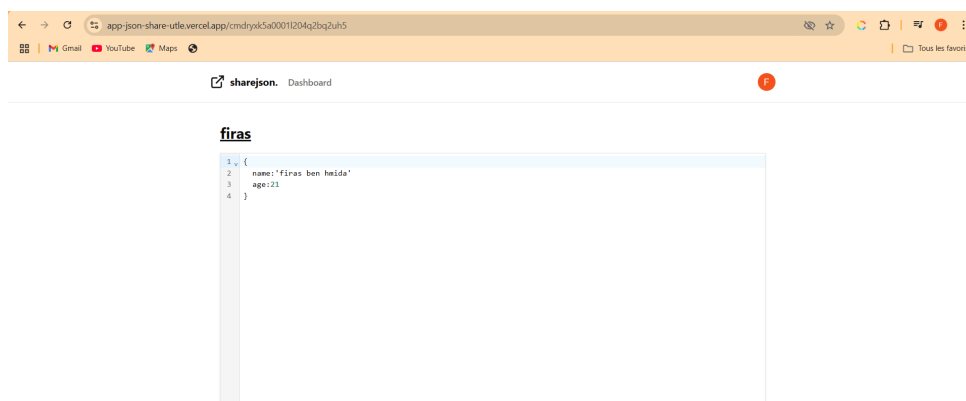


FIGURE 2.3 – partage de données JSON

2.5 Lien Vercel du projet

Démo en ligne : <https://app-json-share.vercel.app/>