

Software Development :

PictsManager



Introduction

PICTSMANAGER : une plateforme conviviale et sécurisée pour les utilisateurs afin de capturer, stocker et partager leurs précieux souvenirs photographiques.



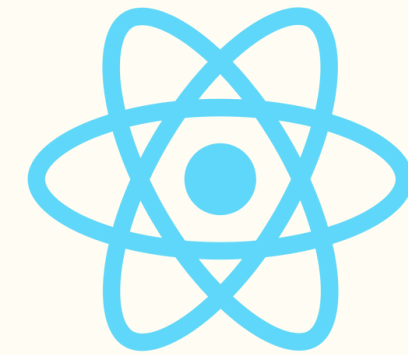
Technologies utilisées



To-do List



Plateforme de
communication



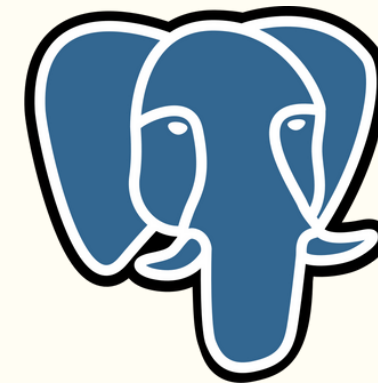
React Native



Plateforme
collaborative



Testeur d'API

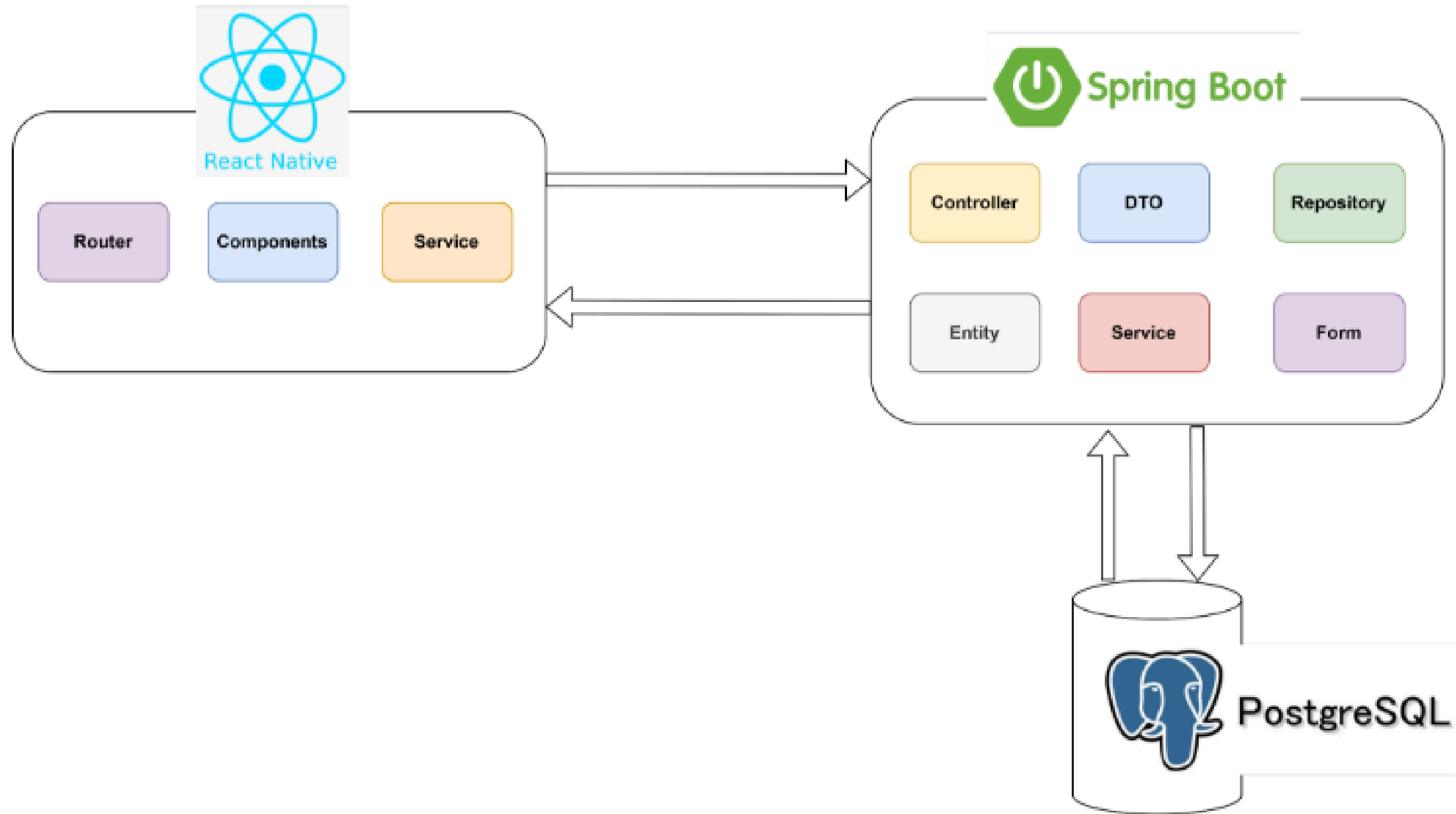


PostgreSQL

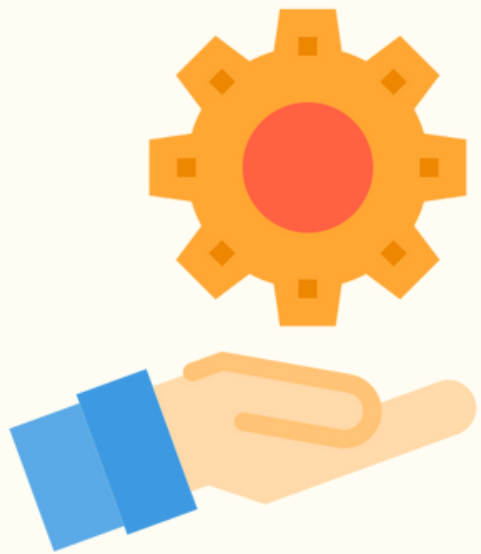


Docker

Architecture de l'application



Fonctionnalités clés



- Prise de photos
- Stockage dans des albums
- Gestion des autorisations (visibilité des photos)
- Compression d'images
- Suppressions de photos et d'albums
- Edition des données des photos et des albums

Sécurité et Gestion des Utilisateurs

- **Authentication** : Les utilisateurs se connectent avec leurs identifiants, recevant ensuite un jeton JWT qui est utilisé pour authentifier les requêtes ultérieures.
- **Modèle de propriété** : Les albums et les photos sont liés à leur créateur, garantissant que seul le propriétaire peut les modifier ou les supprimer.



Conception UX et Accessibilité

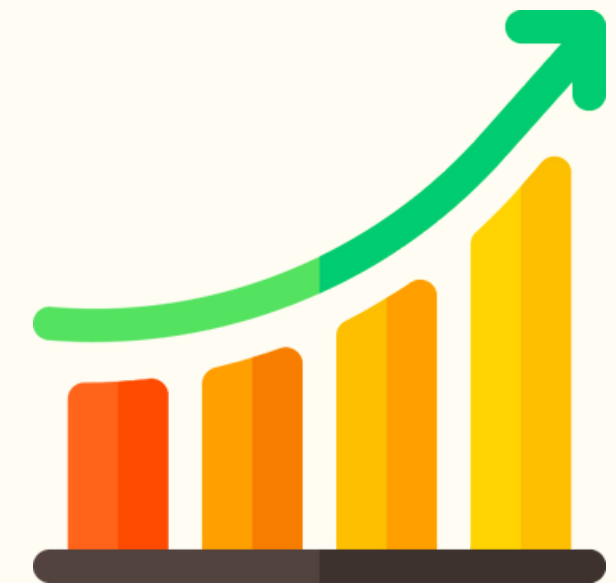
L'interface utilisateur est conçue pour être conviviale, avec une navigation simple et des éléments familiers.



The image shows a login interface for 'PictsManager' on a dark background. At the top is a colorful circular logo. Below it, the text 'Sign in to PictsManager' is displayed, with 'PictsManager' in orange. There are two input fields: 'Email address' and 'Password', both with light gray borders. Below these is a large orange button labeled 'Sign up'. Under the button, the text 'Don't have an account? [Sign up](#)' is shown, with 'Sign up' as a link. At the bottom, the text 'Disconnect' is visible.

Scalabilité et Performance

- **Architecture évolutive** : Notre architecture est conçue pour être scalable, capable de gérer une grande quantité de données et de trafic.
- **Optimisation des performances** : Nous utilisons des algorithmes efficaces et des techniques de compression pour assurer une réactivité maximale de l'application.



Tests et Qualité

- **Politique de test** : Nous avons mis en place une politique de test complète, comprenant des tests unitaires, des tests d'intégration et des tests de performance.
- **Documentation et Qualité du code** : Notre code est soigneusement documenté, suivant les normes de JavaDoc, et respecte les meilleures pratiques de qualité du code. Nous avons aussi mis en place Swagger.



Améliorations possibles

- Authentification Multifacteur (MFA)
- Cryptage de Bout en Bout
- Audit de Sécurité
- Journalisation des Activités
- Gestion des Sessions
- Chiffrement de Données Sensibles
- Alertes de Sécurité
- Analyse des Vulnérabilités
- Formation en Sécurité
- Surveillance en Temps Réel

Merci