**D'après le code et la configuration du projet, voici les langages et technologies utilisés :**

Langages de programmation

* **TypeScript** - Langage principal pour le frontend et backend
* **JavaScript** - Base du projet avec Node.js
* **SQL** - Pour la base de données PostgreSQL

Frontend

* **React 18** - Framework principal avec hooks
* **Tailwind CSS** - Système de styles utilitaires
* **Vite** - Outil de build et serveur de développement
* **Wouter** - Routage côté client
* **TanStack Query** - Gestion d'état serveur
* **React Hook Form** - Gestion des formulaires
* **Zod** - Validation des données

Backend

* **Node.js** - Environnement d'exécution
* **Express.js** - Framework serveur web
* **Drizzle ORM** - ORM pour base de données
* **Neon** - Driver PostgreSQL serverless

Base de données

* **PostgreSQL** - Base de données relationnelle
* **Drizzle Kit** - Migrations et gestion du schéma

Composants UI

* **Radix UI** - Composants primitifs accessibles
* **Shadcn/ui** - Système de composants
* **Lucide React** - Icônes

Outils de développement

* **ESBuild** - Bundler rapide
* **TypeScript Compiler** - Vérification de types
* **PostCSS** - Traitement CSS
* **Autoprefixer** - Préfixes CSS automatiques

Déploiement

* **Replit** - Plateforme d'hébergement et développement

Cette stack moderne privilégie TypeScript pour la sécurité des types, React pour l'interface utilisateur, et PostgreSQL pour la persistance des données, avec un focus sur la performance et l'expérience développeur.

**Voici les étapes complètes pour lancer ce projet en local :**

Prérequis à installer

1. Node.js

# Installer Node.js version 18 ou supérieure

# Télécharger depuis https://nodejs.org/

# Ou avec un gestionnaire de versions comme nvm :

nvm install 18

nvm use 18

2. PostgreSQL

# Sur macOS avec Homebrew :

brew install postgresql

brew services start postgresql

# Sur Ubuntu/Debian :

sudo apt-get install postgresql postgresql-contrib

sudo systemctl start postgresql

# Sur Windows :

# Télécharger depuis https://www.postgresql.org/download/windows/

3. Git (optionnel)

# Pour cloner le projet

git clone [URL\_DU\_PROJET]

cd richat-funding-tracker

Étapes de lancement

1. Installation des dépendances

# Installer toutes les dépendances npm

npm install

2. Configuration de la base de données

# Créer une base de données PostgreSQL

createdb richat\_funding\_tracker

# Ou via psql :

psql -U postgres

CREATE DATABASE richat\_funding\_tracker;

\q

3. Variables d'environnement

# Créer un fichier .env à la racine

touch .env

# Ajouter la configuration de base de données :

DATABASE\_URL=postgresql://username:password@localhost:5432/richat\_funding\_tracker

PGUSER=your\_username

PGPASSWORD=your\_password

PGHOST=localhost

PGPORT=5432

PGDATABASE=richat\_funding\_tracker

4. Initialisation du schéma

# Pousser le schéma vers la base de données

npm run db:push

5. Lancement du serveur de développement

# Démarrer le serveur (frontend + backend)

npm run dev

Vérification du fonctionnement

L'application sera accessible sur :

* **Frontend** : [http://localhost:5000](https://replit.com/@khalefami/MauriTender)
* **Backend API** : [http://localhost:5000/api](https://replit.com/@khalefami/MauriTender)

Scripts disponibles

npm run dev # Serveur de développement

npm run build # Build de production

npm run start # Serveur de production

npm run db:push # Mise à jour du schéma DB

Structure des ports

* **Port 5000** : Serveur Express (backend + frontend statique)
* **Vite dev server** : Intégré dans Express en mode développement

Dépannage courant

1. **Erreur de connexion DB** : Vérifier que PostgreSQL est démarré
2. **Port déjà utilisé** : Changer le port dans les variables d'environnement
3. **Dépendances manquantes** : Relancer npm install

Le projet utilise une configuration unifiée où Express sert à la fois l'API et le frontend, simplifiant le déploiement et évitant les problèmes de CORS.

Scroll to latest

**Installation de git et github**

Initialiser git :

1. Nom d’utilisateur : ***git config –global user.name «  Aminetou«***
2. Email : ***git config –global user.email “***[***khalefami@gmail.com”***](mailto:khalefami@gmail.com”%20)

On ouvre vscode et dans un terminal de vscode on tape ::

Version de git : ***git -v***

initialisé git : ***git – init***

pour voir les fichiers : ***git status***

Sélectionner tous les fichiers : ***git add .***

***git commit -m « sauvegarde du projet le 10/07/2025 »***

connecter repositori avec github : ***git remote add origin*** [***https://github.com/El-Khalef/Aminetou\_Richat.git***](https://github.com/El-Khalef/Aminetou_Richat.git)

transferer le dossier dans github : ***git push origin main***