

## Documentation Développeur

### Introduction

**BraceletStats** est une application d'analyse de données conçue pour une entreprise spécialisée dans la vente de bracelets. Elle permet de visualiser les performances des ventes, de gérer l'inventaire et de produire des analyses avancées pour faciliter les décisions stratégiques.

### Technologies Utilisées

- **Frontend :**
  - React (composants modulaires)
  - Vite (outil de build rapide)
  - TypeScript (typage statique pour plus de robustesse)
- **Styling :** Tailwind CSS pour une mise en page moderne et réactive.
- **Icônes :** Lucide React pour des icônes élégantes et adaptables.
- **Visualisation des données :** Recharts pour créer des graphiques interactifs.

### Structure du Projet

```
braceletstats/  
├── src/  
│   ├── components/    # Composants React (Tableaux, Graphiques, etc.)  
│   ├── data/          # Données statiques et exemples  
│   ├── pages/         # Pages principales de l'application  
│   ├── hooks/         # Hooks personnalisés pour les états et effets  
│   ├── types/         # Interfaces TypeScript  
│   └── public/        # Assets publics  
├── package.json       # Dépendances et scripts  
├── tailwind.config.js  # Configuration Tailwind CSS  
├── vite.config.ts     # Configuration Vite  
└── README.md          # Documentation principale
```

### Installation

1. Clonez le dépôt : git clone  
[git@github.com:Hodavia05/CreaTEK\\_Analyse\\_de\\_donnee.git](https://github.com/Hodavia05/CreaTEK_Analyse_de_donnee.git)
2. Accédez au répertoire du projet : cd project
3. Installez les dépendances : npm install

4. Lancez l'application en mode développement : `npm run dev`
5. Accédez à l'application sur <http://localhost:3000>.

### Fonctionnement

- Les données de test sont présentes dans le dossier `data/`.
- Les composants de visualisation utilisent Recharts pour afficher des courbes, des histogrammes et des camemberts interactifs.
- Les états locaux sont gérés avec des hooks React.

### Personnalisation

- **Graphiques** : Modifiez les données dans `data/salesData.js` pour ajuster les visualisations.
- **Styles** : Personnalisez le thème dans `tailwind.config.js`.
- **Types** : Ajoutez ou mettez à jour les types dans `types/` pour garantir la cohérence du typage TypeScript.