

# Mini CRM - Challenge Technique AriMayi

---

Ce projet est une application front-end d'un mini CRM développée avec Next.js. Il a été réalisé dans le cadre d'un challenge technique pour une opportunité de stage chez AriMayi. L'objectif est de démontrer la maîtrise des technologies front-end modernes, la structuration de projet, et le sens du design et de l'ergonomie.

## Aperçu du Projet

L'application permet de gérer une liste de clients, de consulter leurs informations détaillées et d'en ajouter de nouveaux via un formulaire. Toutes les données sont simulées côté client, sans nécessiter de backend.

**[Lien vers le déploiement]** (Ajoutez le lien si vous déployez l'application)

## ✦ Fonctionnalités

### 1. Page de Connexion (Mock)

- Une interface de connexion simple et épurée.
- Entièrement responsive.
- Aucune logique de validation (simulation).

### 2. Dashboard

- Une barre de navigation permettant d'accéder aux différentes sections : **Dashboard**, **Clients**, **Ajouter un client**.
- Un composant de layout principal qui structure l'ensemble des pages.

### 3. Liste des Clients

- Affichage des clients dans un tableau clair et lisible.
- Données mockées (générées statiquement ou avec Faker.js).
- **Colonnes** : Nom, Email, Téléphone, Date de création.
- **Tri** : Possibilité de trier les clients par nom.
- **Recherche** : Une barre de recherche pour filtrer les clients en temps réel.

### 4. Fiche Client

- Page détaillée accessible en cliquant sur un client dans la liste.
- Affiche toutes les informations du client sélectionné.
- Inclut un historique d'activités (simulé) pour le client.

### 5. Formulaire d'Ajout de Client

- Un formulaire pour ajouter un nouveau client.
- **Champs obligatoires** : Nom, Prénom, Email, Téléphone.
- **Validation** : Contrôle du format de l'email et du téléphone.

- Affichage d'un message de succès après soumission.

## Stack Technique

- **Framework** : [Next.js](#) (avec App Router)
- **Librairie** : [React](#) (v18+)
- **Styling** : [Tailwind CSS](#)
- **Gestion de Formulaires** : [React Hook Form](#) (ou gestion manuelle)
- **Icônes** : [Font Awesome](#) (ou autre)
- **Typage** : [TypeScript](#) (en bonus)

## Installation et Lancement

Pour exécuter ce projet en local, suivez ces étapes :

### 1. Clonez le dépôt :

```
git clone https://github.com/votre-username/AriMayi-assignment-nextjs-mini-crm.git
cd AriMayi-assignment-nextjs-mini-crm
```

### 2. Installez les dépendances :

```
npm install
# ou
yarn install
```

### 3. Lancez le serveur de développement :

```
npm run dev
# ou
yarn dev
```

### 4. Ouvrez votre navigateur et allez sur <http://localhost:3000>.

## Structure du Projet

Le projet suit une structure organisée pour faciliter la maintenance et la scalabilité :

```
/
├── app/                # App Router : pages, layouts, etc.
│   ├── (auth)/login/  # Route pour la connexion
│   ├── (dashboard)/   # Routes protégées après connexion
│   └── layout.tsx      # Layout du dashboard
```

```
| | | page.tsx      # Page du dashboard
| | | clients/     # Pages liées aux clients
| | | ...
| components/     # Composants réutilisables (UI, formulaires, etc.)
| | ui/           # Petits composants d'interface (Button, Input...)
| | ...
| data/           # Données mockées (ex: clients.json)
| hooks/          # Hooks personnalisés
| lib/            # Fonctions utilitaires
| public/         # Fichiers statiques (images, polices)
| ...
```

## 🌟 Bonus

Cette section peut être remplie si des fonctionnalités bonus ont été implémentées :

- ☐ Système de "tags" par client.
- ☐ Pagination pour la liste des clients.
- ☐ Responsive mobile complet et soigné.
- ☐ Gestion de state avec Zustand ou Redux Toolkit.
- ☐ Tests unitaires ou d'intégration (Jest, Testing Library).

---

Merci d'avoir pris le temps d'évaluer ce challenge.