

QuickRepair France

Digitalisation du Système d'Information

Partie 2 — Section B

Documentation de l'Application Retool

Réalisé par :	Ouafae Berhili & Amal Bouzaher
Formation :	CDA Bac+3 — ESIC
Groupe :	AMOU
Outil utilisé :	Retool (Low-code)
Base de données :	Supabase PostgreSQL (19 tables)
Date :	27 février 2026

Ce document présente les captures d'écran de l'application de gestion développée sur Retool, avec description détaillée de chaque module.

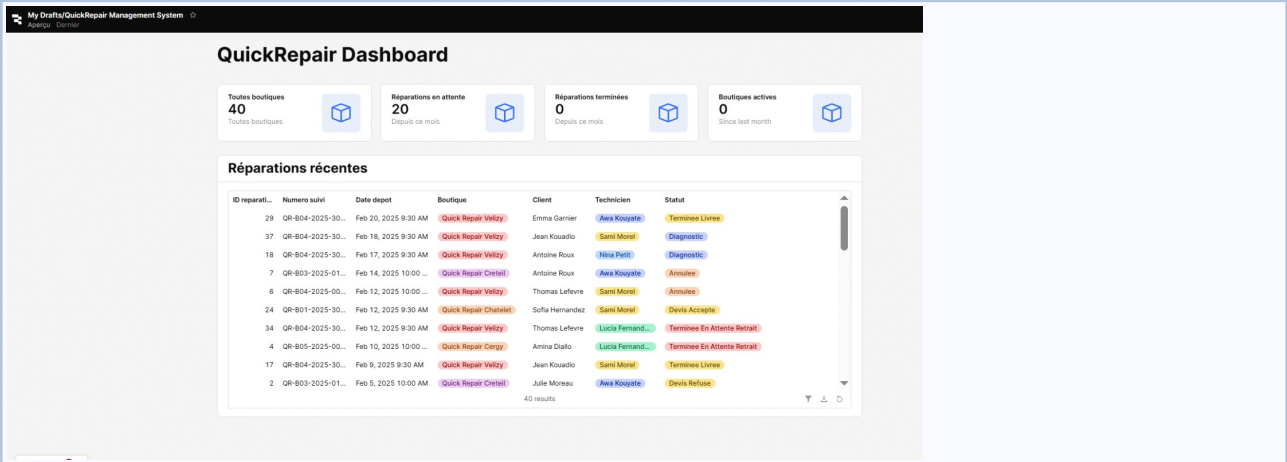
1. Présentation de l'Application

L'application QuickRepair a été développée sur **Retool** (low-code), connectée à une base **PostgreSQL Supabase** (19 tables, Session Pooler SSL, eu-west-1). Elle couvre 5 boutiques Île-de-France et propose 3 modules principaux.

Module	Fonctionnalités
Dashboard	KPIs temps réel, réparations récentes
Clients	Listing, recherche, création, fiche détail
Réparations	Listing filtré, détail, historique statuts, mise à jour

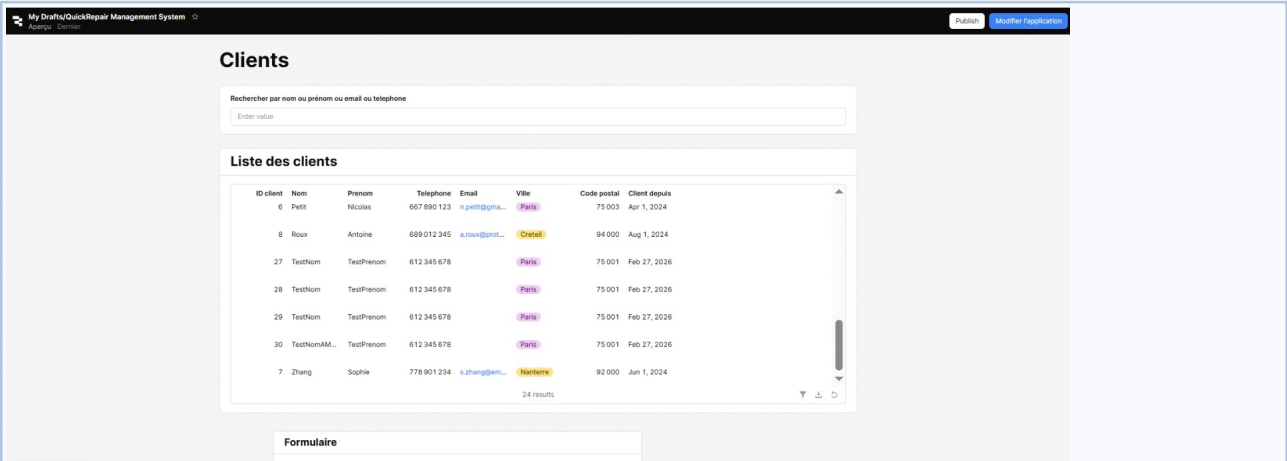
2. Captures d'écran de l'Application

Figure 1 — Dashboard principal (Vue d'ensemble)



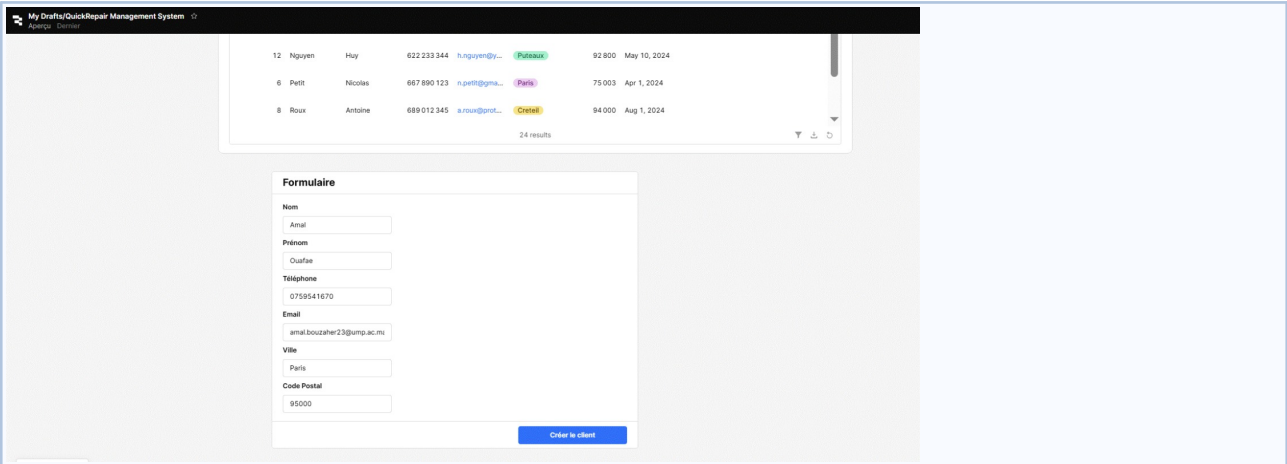
Le tableau de bord affiche les KPIs en temps réel : nombre total de réparations (40), réparations en attente ce mois (20), réparations terminées et boutiques actives. La liste des réparations récentes présente ID, numéro de suivi, date, boutique, client, technicien et statut, avec une coloration par statut pour une lecture rapide.

Figure 2 — Module Clients : liste et recherche



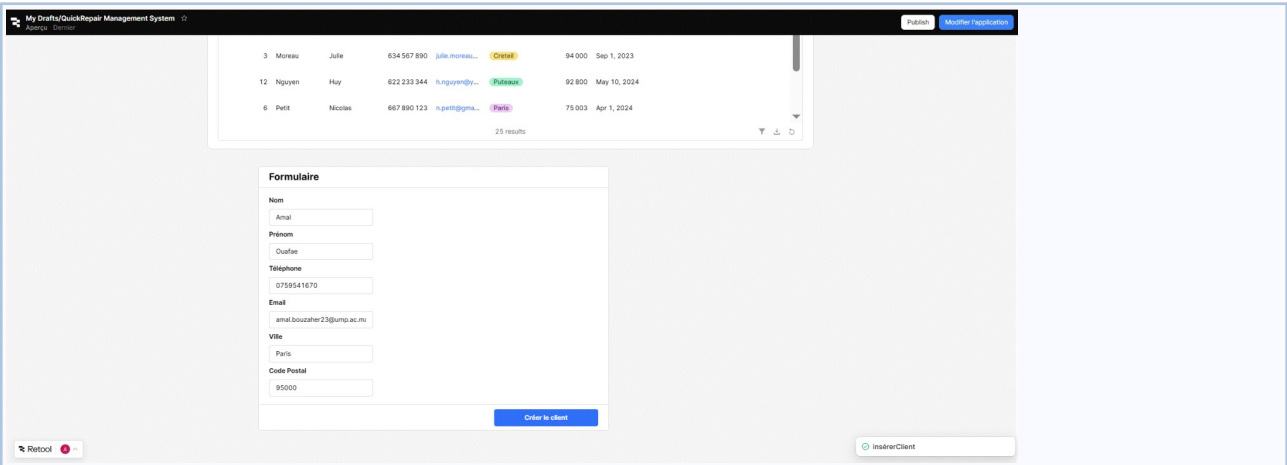
La page Clients propose un champ de recherche multicritères (nom, prénom, email, téléphone). La liste affiche 24 résultats avec ID, nom, prénom, téléphone, email, ville (colorisée) et date d'inscription. Des clients récemment ajoutés le 27/02/2026 sont visibles.

Figure 3 — Module Clients : formulaire de création



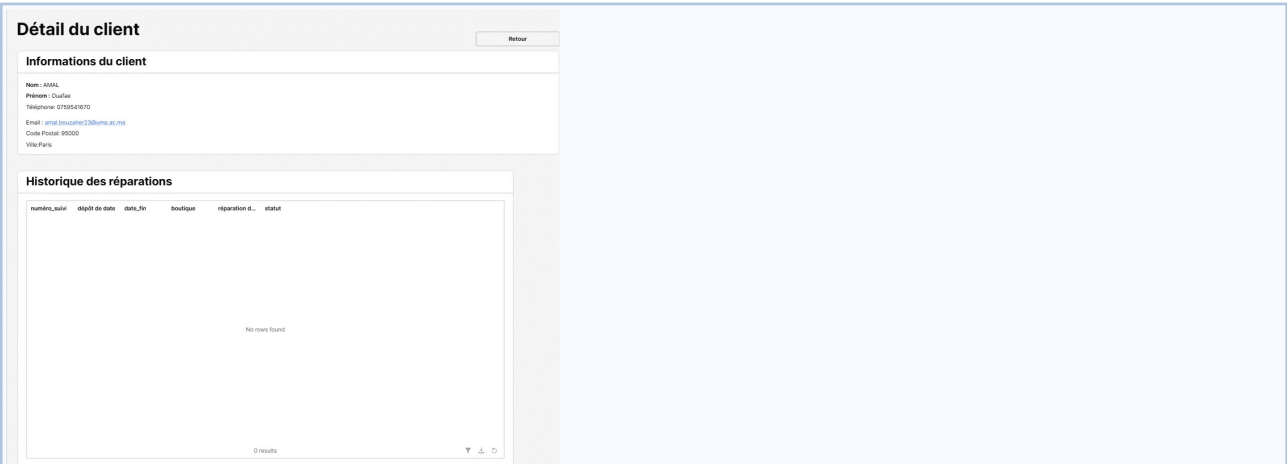
Le formulaire de création permet de saisir Nom, Prénom, Téléphone, Email, Ville et Code Postal. Le bouton 'Créer le client' déclenche un INSERT INTO en base Supabase. On voit ici les données de test saisies (Amal / Ouafae, Paris).

Figure 4 — Confirmation d'insertion client



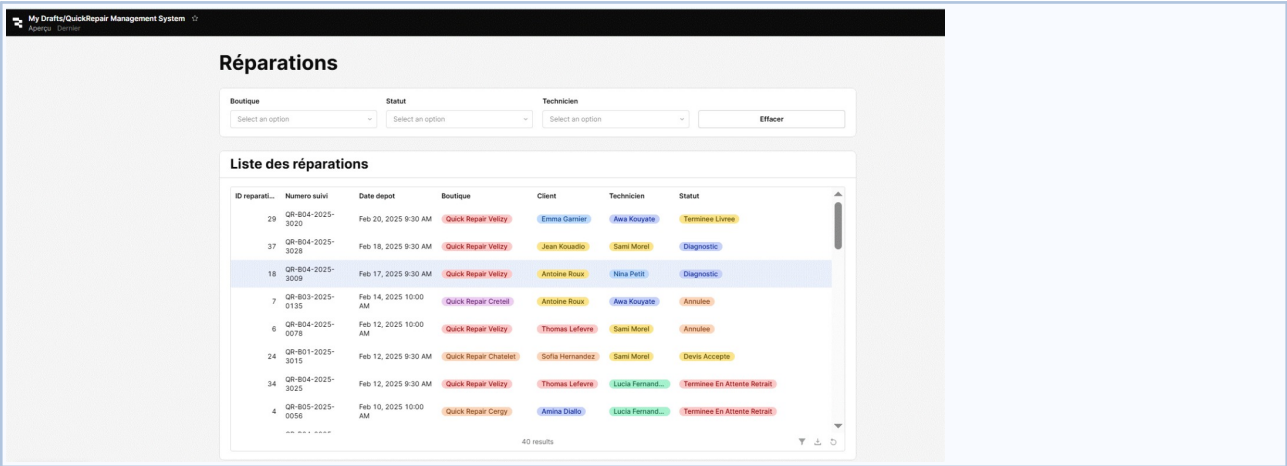
Après soumission, le compteur passe à 25 résultats, confirmant l'insertion en base. L'indicateur 'InsérerClient' visible en bas à droite confirme l'exécution de la requête SQL et assure la traçabilité des opérations CRUD.

Figure 5 — Fiche détail client



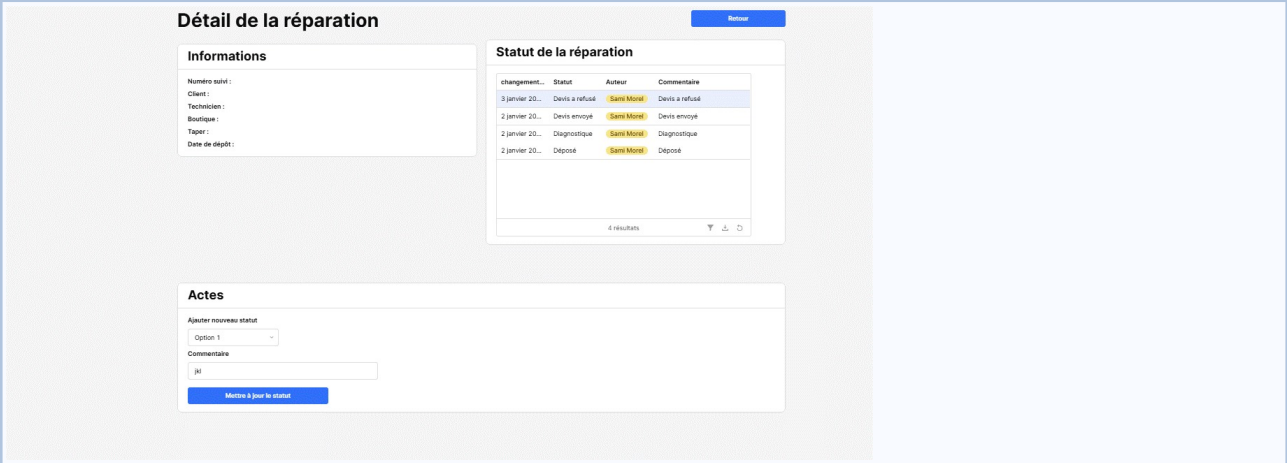
La fiche détail affiche les informations personnelles (nom, prénom, téléphone, email, code postal, ville) et un tableau 'Historique des réparations' avec numéro de suivi, dates, boutique, description et statut. Le client nouvellement créé n'a pas encore de réparation associée.

Figure 6 — Module Réparations : liste avec filtres



La page Réparations liste les 40 réparations filtrables par Boutique, Statut et Technicien via trois menus déroulants. Chaque ligne affiche ID, numéro de suivi (ex. QR-B04-2025-3009), date de dépôt, boutique, client, technicien et statut colorisé.

Figure 7 — Détail réparation et mise à jour du statut



La fiche réparation est organisée en 3 blocs : Informations (numéro, client, technicien, boutique, type, date), Historique des statuts (changements avec auteur et commentaire), et Actes permettant d'ajouter un nouveau statut via menu déroulant + commentaire. Le bouton 'Mettre à jour' insère dans historique_statut.

3. Bilan des Fonctionnalités Implémentées

Fonctionnalité	Statut	Détails
Dashboard KPIs	✓ Réalisé	4 indicateurs temps réel
Liste des réparations	✓ Réalisé	40 réparations, tri par date
Filtres réparations	✓ Réalisé	Par boutique, statut, technicien
Détail réparation	✓ Réalisé	Historique complet des statuts
Mise à jour statut	✓ Réalisé	Insert historique_statut
Liste des clients	✓ Réalisé	Recherche multicritères
Création client	✓ Réalisé	Formulaire + INSERT SQL
Fiche détail client	✓ Réalisé	Infos + historique réparations
Connexion Supabase	✓ Réalisé	Session Pooler SSL, eu-west-1

