

# Rapport de Projet : Application de Gestion de Projets

**Auteur :** Ayoub Majjid

**Portfolio :** [majjid.com](http://majjid.com)

**Email :** [ayoub@majjid.com](mailto:ayoub@majjid.com)

**Contexte :** Projet Flutter 5IIR 16 EMSI

**Date :** 21 Décembre 2025

## Table des Matières

1. Introduction
2. Architecture et Technologies
3. Description Détallée des Pages
4. Design et Expérience Utilisateur (UX)
5. Qualité Technique
6. Conclusion

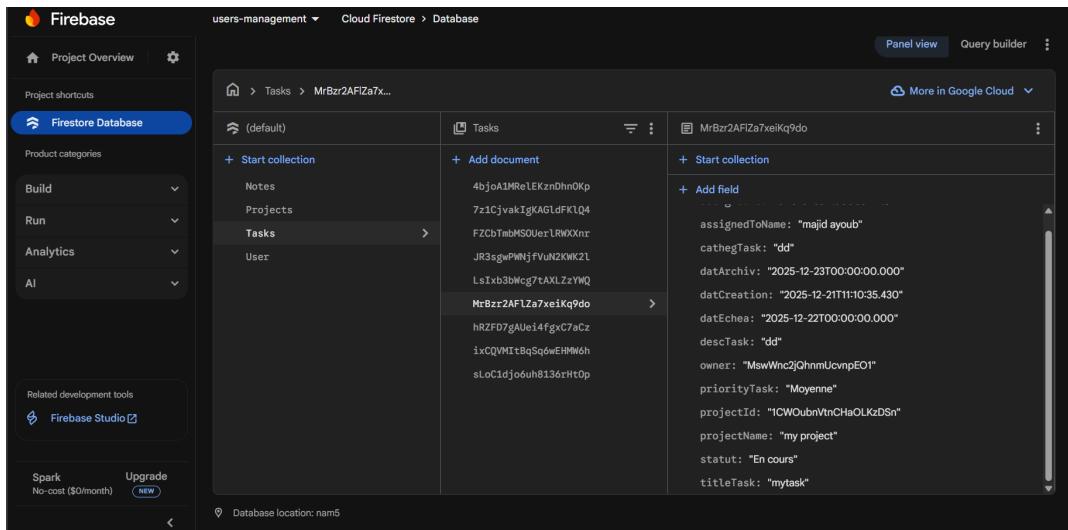
## 1. Introduction

Ce projet consiste en la réalisation d'une application mobile et web de **Gestion de Projets**, développée avec le framework **Flutter**. L'objectif principal est de fournir une solution complète pour la gestion des tâches, des projets et des équipes, avec une séparation claire des rôles (Administrateur et Membre). L'application se distingue par une interface utilisateur moderne, fluide et adaptative (Support des thèmes Clair et Sombre).

## 2. Architecture et Technologies

### 2.1 Technologies Utilisées

- **Frontend :** Flutter (Dart), offrant une compatibilité multiplateforme (Android, Windows, Web).
- **Backend :** Firebase.



The screenshot shows the Firebase Cloud Firestore interface. The left sidebar shows 'Project Overview' and 'Firestore Database' selected. Under 'Firestore Database', there are 'Product categories', 'Build', 'Run', 'Analytics', and 'AI' sections. The main area shows a collection named 'Tasks' with a document named 'MrBzr2AFIza7xeiKq9do'. The document details are as follows:

Field	Value
assignedToName	"majid ayoub"
category	"Tasks"
descTask	"dd"
dueArchiv	"2025-12-23T00:00:00.000"
dueDate	"2025-12-21T11:03:43.430"
dueTime	"00:00:00.000000"
owner	"MswWlnC2jQhnmUcvnpEo1"
priorityTask	"Moyenne"
projectId	"ICWOUbnVtnChsOLKzDSn"
projectName	"my project"
statut	"En cours"
titleTask	"mytask"

- **Authentication** : Gestion sécurisée des sessions utilisateurs via Email/Mot de passe. - **Cloud Firestore** : Base de données NoSQL temps réel pour le stockage performant des tâches, projets et utilisateurs. - **Outils de Développement** : VS Code, Git.

### 2.2 Structure du Code (Feature-First)

L'architecture du projet est modulaire, organisée par fonctionnalité pour faciliter la maintenance et l'évolutivité :

- **features/auth** : Gestion de la connexion et sécurité.
- **features/dashboard** : Les tableaux de bord principaux adaptés aux rôles.
- **features/tasks** : Gestion complète du cycle de vie des tâches.
- **features/projects** : Gestion des projets globaux.
- **features/users** : Administration des membres et des droits.
- **features/settings** : Configuration et préférences de l'application.

## 3. Description Détallée des Pages

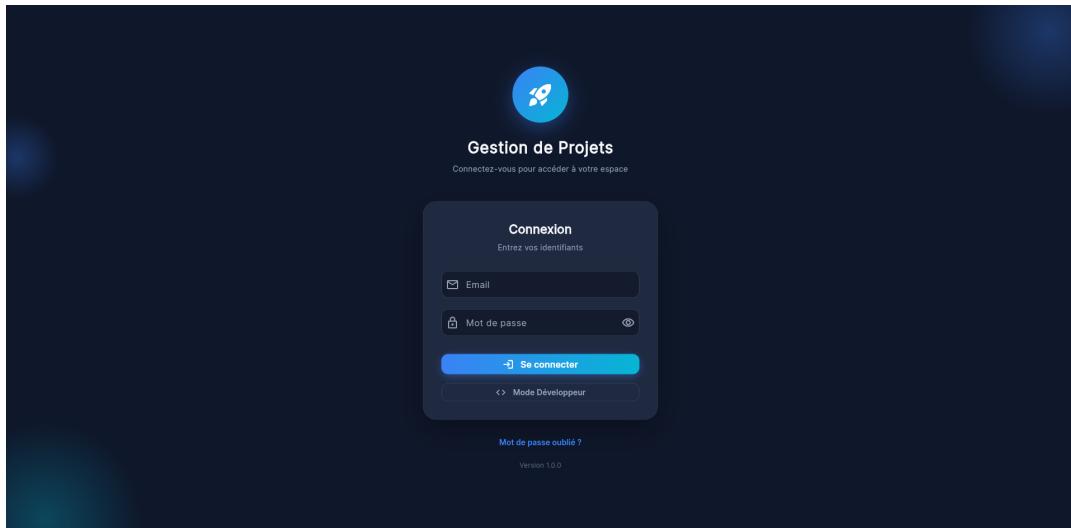
Cette section présente chaque écran de l'application, illustrant le parcours utilisateur complet.

### 3.1 Authentification

#### Page de Connexion (LoginPage)

Point d'entrée sécurisé de l'application.

- **Fonctionnalités** : Validation des champs en temps réel, gestion des erreurs de connexion, accès à la récupération de compte.
- **Design** : Interface épurée mettant l'accent sur l'accessibilité.



#### Page Mot de Passe Oublié (ForgotPasswordPage)

Permet à un utilisateur de réinitialiser son mot de passe de manière autonome.

- **Fonctionnalités** : Envoi d'email via Firebase Auth, feedback visuel instantané (succès/erreur).

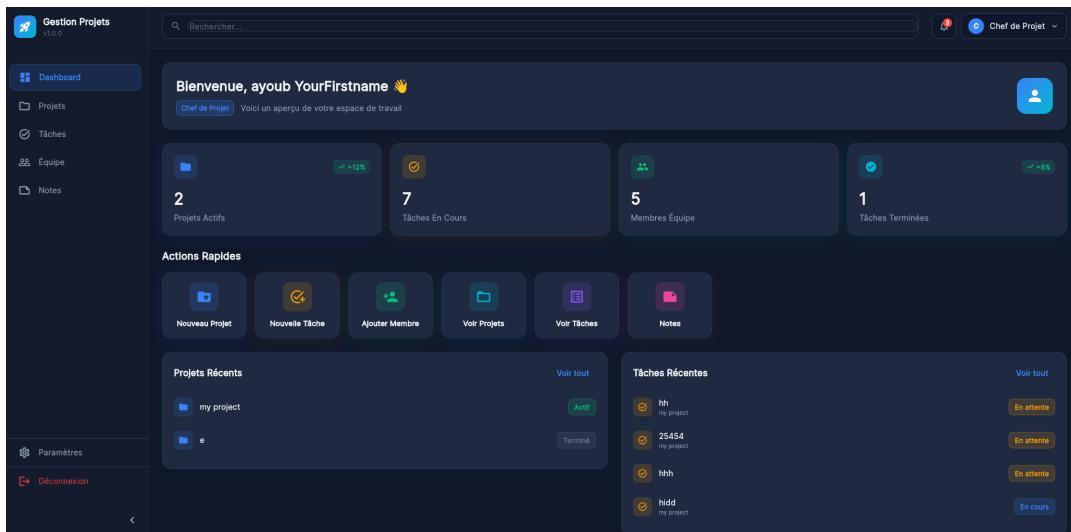


### 3.2 Tableaux de Bord (Dashboard)

#### Dashboard Administrateur (AdminPage)

Le centre de contrôle stratégique pour les gestionnaires.

- **Fonctionnalités** :
  - **Statistiques (KPIs)** : Vue synthétique des projets, tâches et membres.
  - **Tâches Récentes** : Flux d'activité en temps réel.
  - **Actions Rapides** : Raccourcis pour la création de ressources.
  - **Menu Latéral** : Navigation fluide vers toutes les sections.

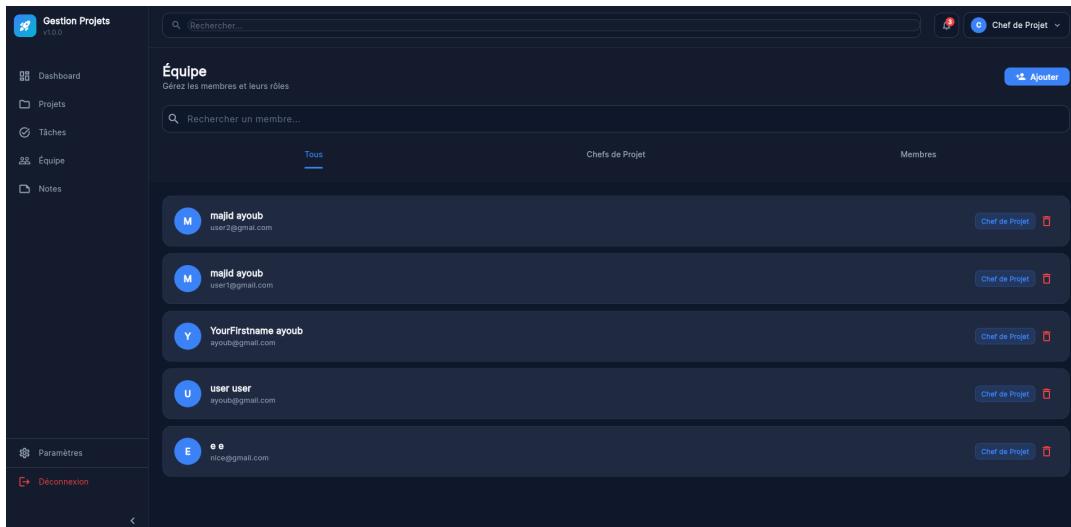


### Dashboard Membre (StudentPage)

L'espace personnel dédié à la productivité individuelle.

- **Fonctionnalités :**

- **Mes Tâches** : Filtrage automatique des tâches assignées à l'utilisateur.
- **KPIs Personnels** : Suivi de la performance individuelle.
- **Priorités** : Mise en évidence visuelle des urgences.



### 3.3 Gestion des Projets

#### Liste des Projets (ViewProjectsPage)

Vue d'ensemble du portefeuille de projets.

- **Fonctionnalités** : Présentation sous forme de cartes interactives, barre de recherche.

### Détails d'un Projet (ProjectDetailsPage)

Fiche complète d'un projet.

- **Fonctionnalités** : Indicateurs de progression, dates clés, et liste des tâches associées.

### Ajout/Modification de Projet

Formulaires dynamiques pour gérer le cycle de vie des projets.

### 3.4 Gestion des Tâches

#### Liste des Tâches (ViewTasksPage)

Le cœur opérationnel de l'application pour le suivi quotidien.

- **Fonctionnalités :**

- **Filtrage par Onglets :** (Toutes / Actives / Terminées).
- **Indicateurs Visuels :** Priorité (Drapeaux couleur) et Membre assigné.

### Détails d'une Tâche (TaskDetailsPage)

Informations complètes et contexte d'une tâche.

### Édition de Tâche (EditTaskPage)

Interface de mise à jour des tâches.

- **Fonctionnalités :**

- **Sélecteurs Intelligents :** Listes connectées à Firebase pour le Projet et l'Assignation.
- **Statut Dynamique :** Boutons d'action rapide pour changer l'état (En cours, Terminé).

## 3.5 Administration Utilisateurs

### Liste des Utilisateurs (ViewUsersPage)

Annuaire de l'équipe avec gestion des rôles.

- **Fonctionnalités :** Recherche instantanée et distinction visuelle Admin/Membre.

### Ajout d'Utilisateur (AddUserPage)

Formulaire de création de compte.

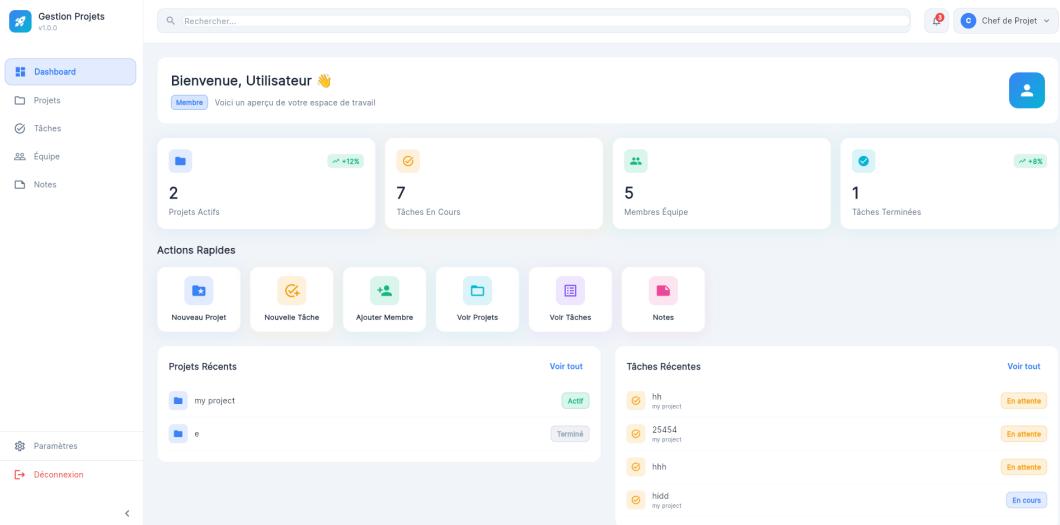
### 3.6 Paramètres (SettingsPage)

Configuration personnelle de l'interface.

- Fonctionnalités** : Basculement instantané entre le Thème Clair et le Thème Sombre.

## 4. Design et Expérience Utilisateur (UX)

L'application adopte un style visuel moderne, utilisant des effets de transparence et de flou ("Glassmorphism") pour les cartes et les panneaux, offrant une esthétique épurée et professionnelle.



## Thème Dynamique (Light & Dark Mode)

Une attention particulière a été portée à l'accessibilité et au confort visuel. L'application supporte nativement :

- **Mode Sombre** : Couleurs profondes (Bleu nuit/Violet) pour réduire la fatigue visuelle.
- **Mode Clair** : Design épuré et lisible pour les environnements lumineux.

## 5. Qualité Technique

Ce projet respecte les standards de développement professionnels :

- **State Management** : Utilisation optimisée de `StreamBuilder` pour une interface réactive sans latence.
- **Robustesse** : Gestion complète des exceptions, validation des formulaires et feedback utilisateur.
- **Clean Code** : Structure modulaire, nommage explicite et séparation des responsabilités.

## 6. Conclusion

Cette application de Gestion de Projets représente une solution aboutie, alliant la puissance du **Cloud temps réel** avec Firebase et la fluidité de **Flutter**. Elle répond efficacement aux besoins modernes de collaboration et de productivité, tout en offrant une base solide et maintenable pour de futures évolutions.