



Mini CV + App Flutter pilotée par API

Rapport de projet

Université : Université Mundiapolis

Encadrant : Prof. Ayad

Étudiant : *Mohamed AASSOU*

Filière : *2ANCI - Génie Informatiquee*

Année : 2026

Casablanca, Maroc

1^{er} janvier 2026

Table des matières

1	Introduction	2
2	Aperçu de l'application	3
2.1	Onglets et fonctionnalités	3
3	Architecture et technologies	4
3.1	Organisation du projet	4
3.2	Dépendances principales	4
4	Configuration des APIs	5
4.1	Pixabay Video	5
4.2	GitHub	5
5	Interfaces et captures d'écran	6
6	Installation et exécution	12
6.1	Prérequis	12
6.2	Démarrage	12
7	Tests et qualité	13
8	Conclusion	14

Chapitre 1

Introduction

Ce rapport présente une application Flutter démonstrative combinant un *mini CV* et des fonctionnalités connectées à des APIs publiques. L'application offre une navigation par onglets (*bottom navigation*) pour explorer :

- **Accueil** : page de portfolio avec animations légères.
- **Recherche Vidéo** : recherche et lecture de vidéos depuis l'API *Pixabay Video*.
- **Dépôts GitHub** : liste des dépôts publics de l'utilisateur `mohammedaassou`.
- **Profil** : page de profil à onglets (*About / Settings / Links*).

Les objectifs pédagogiques incluent la manipulation de widgets Flutter, l'appel à des services REST, la structuration d'un projet, et l'intégration de médias (vidéo) et animations.

Chapitre 2

Aperçu de l'application

L'entrée de l'application se situe dans `lib/main.dart`. La navigation principale (*CurvedNavigationBar*) est gérée dans `lib/principale_page.dart`. Les différentes vues sont placées dans `lib/pages/` et les modèles/clients HTTP dans `lib/models/` et `lib/services/`.

2.1 Onglets et fonctionnalités

- **Accueil** (`home_page.dart`) : mini CV (compétences, projets) avec transitions `animations`.
- **Recherche Vidéo** (`video_search_page.dart`) : champ de recherche, liste des résultats, navigation vers `video_detail_page.dart` pour la lecture via `video_player`.
- **Dépôts GitHub** (`repos_page.dart`) : récupération des dépôts publics via `github_api.dart` et affichage (nom, description, langue, etc.).
- **Profil** (`profile_page.dart`) : onglets *About*, *Settings*, *Links*.

Chapitre 3

Architecture et technologies

3.1 Organisation du projet

- `lib/main.dart` : point d'entrée ; configuration de thème et routes.
- `lib/principale_page.dart` : barre de navigation et routing par onglets.
- `lib/services/` : clients HTTP (GitHub, Pixabay). Utilise le package `http`.
- `lib/models/` : modèles de données pour la désérialisation JSON.
- `lib/widgets/` : composants réutilisables (drawer, animations `staggered_fade_scale`).

3.2 Dépendances principales

Extrait depuis `pubspec.yaml` :

- `http` : appels REST.
- `video_player` : lecture vidéo native.
- `animations` : transitions Material motion.
- `curved_navigation_bar` : barre de navigation inférieure.

Chapitre 4

Configuration des APIs

4.1 Pixabay Video

- Base : <https://pixabay.com/api/videos>
- Clé API : stockée dans `lib/env.dart` (champ `pexelsApiKey` ; certains libellés UI mentionnent encore *Pexels*, mais l'implémentation est bien sur *Pixabay*).

4.2 GitHub

- Endpoint : <https://api.github.com/users/mohammedaassou/repos>
- Authentification : non requise pour les requêtes publiques de base.

Chapitre 5

Interfaces et captures d'écran

Les captures suivantes illustrent les écrans clefs de l'application.

21:20



Mohamed Aassou's Portfolio



Mohamed Aassou

Flutter developer focused on clean UI, smooth animations, and API-driven apps.

Skills

Flutter

Dart

Provider

REST APIs

Animations

Projects



Video Search App

Flutter + Provider + Pexels Video API



Portfolio Mini CV

Simple sections with clean UI + animations



Animated Cards

Smooth transitions with the

21:20



Video Search



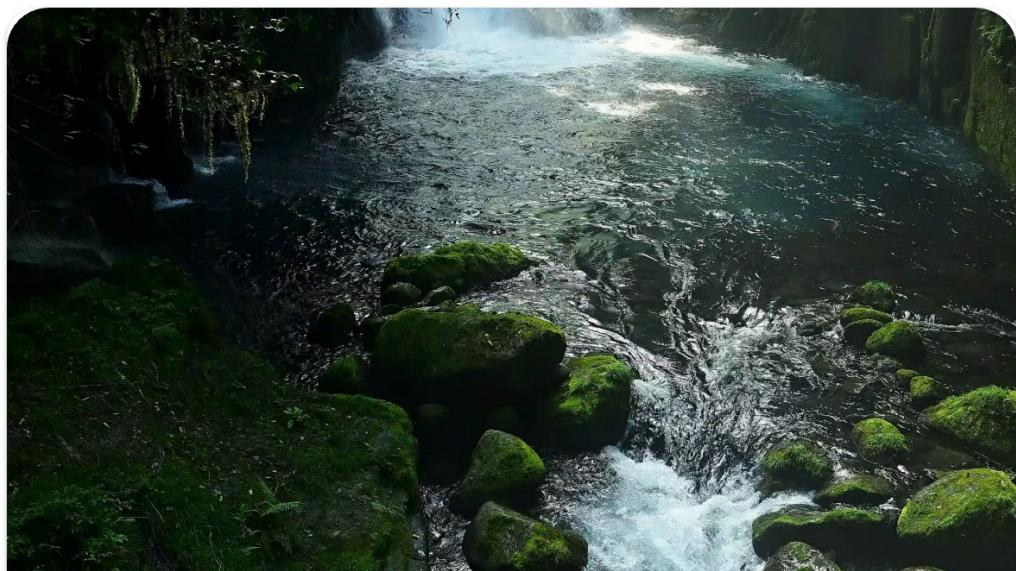
Search free videos on Pexels...



flutie8211

10 seconds

10s

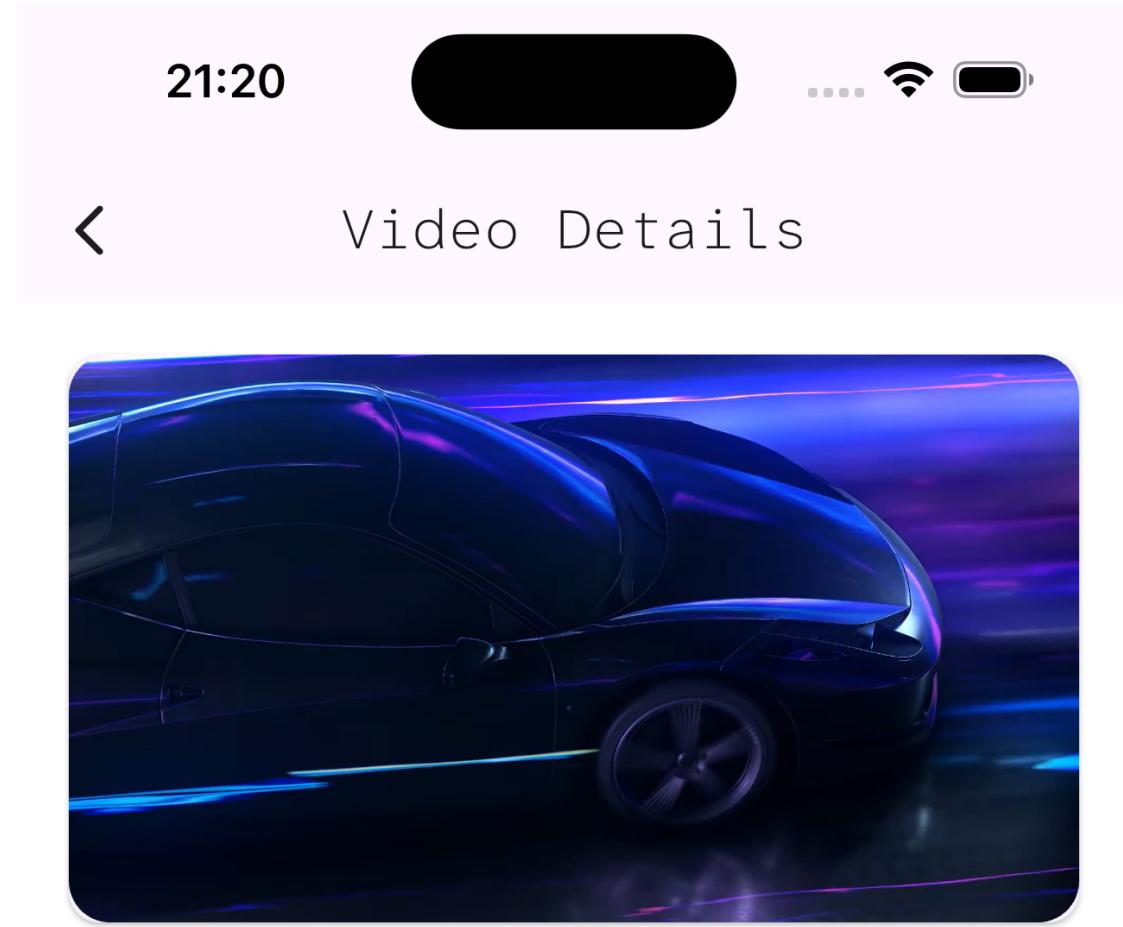


Kanenori

8

20 seconds

20s



The screenshot shows a mobile application interface. At the top, there is a black navigation bar with the time "21:20" on the left and signal, Wi-Fi, and battery icons on the right. Below the navigation bar is a white header bar with a back arrow icon on the left and the text "Video Details" in the center. The main content area features a large, rounded rectangular image of a dark, futuristic car with blue and purple glowing highlights, set against a dark background with motion blur streaks. Below this image is a light gray card with rounded corners containing the text "flutie8211" in bold. Underneath it, the text "Duration: 10s" is displayed. To the left of the card, there is a section titled "About" which includes the text "Source: Pexels • Duration: 10s", "Creator: flutie8211", and "Likes: 21".

21:21

.... WiFi

My GitHub Repos

**angular_app**

No description provided.

☆ 0 ⚡ 0

TypeScript

Updated: 2025-12-31 11:05:57

**chat**

No description provided.

☆ 0 ⚡ 0

JavaScript

Updated: 2024-08-09 19:35:32

**devOps-pipelines**

No description provided.

☆ 0 ⚡ 0

JavaScript

Updated: 2025-12-11 17:23:31

**digital-banking**

No description provided.

☆ 0 ⚡ 0

Java

Updated: 2025-12-30 17:58:22

21:21



Profile

[About](#)[Settings](#)[Links](#)

Mohamed Aassou

Flutter Developer • UI •
Animations • REST APIs

[Flutter](#)[Dart](#)[REST](#)[GitHub](#)[Repos](#)[Public](#)[Videos](#)[Search](#)

About

I build clean, modern Flutter apps with smooth animations and real APIs. This app includes a video search and GitHub repos page.

Chapitre 6

Installation et exécution

6.1 Prérequis

- Flutter SDK (Dart géré par Flutter).
- Un appareil ou émulateur (Android Studio / Xcode / iOS Simulator).

6.2 Démarrage

Depuis la racine du projet :

1. Installation des dépendances : `flutter pub get`
2. Lancement de l'application : `flutter run`

Chapitre 7

Tests et qualité

Des tests de widget de base sont fournis (`test/widget_test.dart`). L’application met l’accent sur une architecture claire, la séparation des responsabilités (services, modèles, vues) et l’utilisation de composants réutilisables.

Chapitre 8

Conclusion

Ce projet illustre une application Flutter moderne intégrant des APIs externes, des animations et des médias. Il constitue une base pédagogique pour approfondir la gestion d'état, l'architecture modulaire, l'optimisation des performances et le déploiement multiplateforme.

Travaux futurs

- Internationalisation complète (i18n) et uniformisation des libellés.
- Ajout de pagination et de mise en cache pour les listes.
- Intégration de thèmes (clair/sombre) et accessibilité.