

L'Institut International de Technologie IIT

Département Technologies de l'Informatique



Compte Rendu

Atelier 6

Pages Galerie & Galerie-Détails

Objectifs

- Créer une page de recherche d'images avec formulaire
- Implémenter une page de détails affichant les résultats Pixabay
- Intégrer l'API Pixabay pour récupérer les images gratuites
- Gérer les requêtes HTTP asynchrones avec le package http
- Afficher les images avec pagination infinie (scroll)
- Présenter les métadonnées détaillées (tags, dimensions, statistiques)
- Implémenter le système de pagination avec ScrollController
- Afficher les informations utilisateur et statistiques d'engagement

Matière : Programmation Mobile

Enseignant : M. Sami Hadhri

Réalisé par :

Ahmed Khelif

Classe : GLID2

Année Universitaire : 2025/2026

Table des matières

1	Introduction.....	2
1.1	Contexte du TP.....	2
1.2	Objectifs	2
1.3	Technologies utilisées.....	2
2	Architecture de l'application.....	3
2.1	Structure du projet.....	3
2.2	API Pixabay.....	3
2.3	Flux de navigation.....	3
3	Annexes - Captures d'écran.....	4
3.1	Figure 1 : Page Galerie - Recherche.....	4
3.2	Figure 2 : Page Galerie-Détails - Résultats "nature".....	4
4	Tests et validation.....	5
4.1	Cas de test.....	5
4.2	Performances.....	5
5	Références	5
5.1	Documentation et ressources.....	5
5.2	API Pixabay - Exemple de réponse.....	5

1 Introduction

1.1 Contexte du TP

Ce travail pratique fait suite aux ateliers précédents et vise à développer une fonctionnalité de recherche d'images dans l'application "Voyage" en utilisant l'API Pixabay pour accéder à une vaste bibliothèque d'images gratuites.

1.2 Objectifs

L'objectif principal de cet atelier est de créer un système de recherche et d'affichage d'images comprenant :

- Une page de recherche avec champ de saisie de mots-clés
- Une page de détails affichant les résultats de recherche d'images
- L'intégration de l'API Pixabay pour données d'images gratuites
- La gestion des requêtes HTTP et du parsing JSON
- L'affichage paginé des images avec scroll infini
- La présentation détaillée des métadonnées (dimensions, vues, téléchargements, likes)
- L'affichage des informations utilisateur et statistiques
- Des indicateurs visuels (chargement, erreurs, états vides)

1.3 Technologies utilisées

- **Framework :** Flutter
- **Langage :** Dart
- **API externe :** Pixabay API (images gratuites)
- **Packages :**
 - http : Requetes HTTP
 - dart:convert : Parsing JSON
- **Widgets :** ListView.builder, Card, CircularProgressIndicator, RefreshIndicator
- **Concepts :** Pagination infinie, ScrollController, State management

2 Architecture de l'application

2.1 Structure du projet

Le projet Flutter "voyage" intègre maintenant les pages gallerie :

```
voyage/
lib/
  main.dart
  config/
    global.params.dart
  menu/
    drawer.widget.dart
  pages/
    gallerie.page.dart
    gallerie-details.page.dart
    [autres pages...] images/
    [autres ressources...]
pubspec.yaml
```

2.2 API Pixabay

L'application utilise l'API de recherche d'images Pixabay :

- **Endpoint** : <https://pixabay.com/api/>
- **Paramètres** : `key={api_key}&q={keyword}&page={page}&per_page={size}`
- **Format** : JSON
- **Données** : Images gratuites avec métadonnées complètes
- **Contenu** : URL images, tags, dimensions, statistiques, informations utilisateur

2.3 Flux de navigation

1. L'utilisateur saisit un mot-clé dans la page Galerie
2. Au clic sur "Chercher", navigation vers la page Galerie-Détails
3. Affichage d'un indicateur de chargement circulaire
4. Requête HTTP vers l'API Pixabay
5. Parsing des données JSON reçues
6. Affichage des images dans un ListView avec pagination infinie
7. Pull-to-refresh pour actualiser les résultats

3 Annexes - Captures d'écran

3.1 Figure 1 : Page Galerie - Recherche

Interface de recherche sur fond vert dégradé avec carte blanche contenant un formulaire de recherche, champ de texte avec icône de recherche, bouton "Rechercher", et informations sur la disponibilité des images Pixabay.

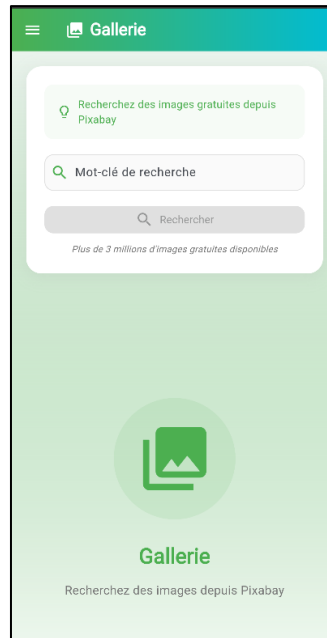


figure 1 – Page Galerie - Recherche

3.2 Figure 2 : Page Galerie-Détails - Résultats "nature"

Page de détails avec en-tête vert affichant "nature, Page 1/50", suivie de cartes d'images empilées montrant des photos de nature avec tags colorés, informations de dimensions et type, statistiques détaillées (vues, téléchargements, likes, commentaires), et informations utilisateur avec avatars.

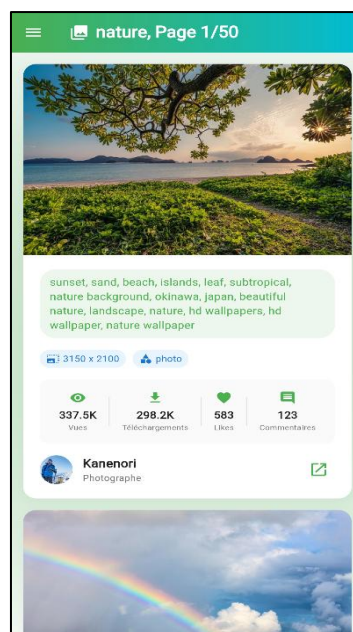


figure 2 – Page Galerie-Détails

4 Tests et validation

4.1 Cas de test

- Recherche avec mot-clé valide ("fleurs", "nature", "ville")
- Recherche avec mot-clé vide (validation)
- Pagination infinie (scroll vers le bas)
- Pull-to-refresh (glisser vers le bas)
- Gestion des erreurs réseau
- Images non disponibles (errorBuilder)
- Navigation retour depuis page détails

4.2 Performances

- Chargement progressif des images
- Mise en cache automatique des images
- Pagination par blocs de 10 images
- Gestion mémoire du ScrollController

5 Références

5.1 Documentation et ressources

- Documentation officielle Flutter : <https://docs.flutter.dev>
- Package HTTP : <https://pub.dev/packages/http>
- Pixabay API : <https://pixabay.com/api/docs/Flutter ListView : https://api.flutter.dev/flutter/widgets/ListView-class.html>
- Flutter RefreshIndicator : <https://api.flutter.dev/flutter/material/RefreshIndicator.html>
- Dart JSON : <https://dart.dev/guides/libraries/library-tour#dartconvert>
- Flutter ScrollController : <https://api.flutter.dev/flutter/widgets/ScrollController-class.html>
- Flutter Image.network : <https://api.flutter.dev/flutter/widgets/Image/Image.network.html>

5.2 API Pixabay - Exemple de réponse

```

1 {
2   "total": 326426,
3   "totalHits": 500,
4   "hits": [
5     {
6       "id": 7120167,
7       "webformatURL": "https://pixabay.com/get/...",
8       "tags": "fleur-de-lis, sphere, ball, floral design",
9       "imageWidth": 1920,
10      "imageHeight": 1920,
11      "views": 4794,
12      "downloads": 4305,
13      "likes": 20,
14      "comments": 7,
15      "user": "GDJ",
16      "userImageURL": "https://cdn.pixabay.com/user/..."
17    }
18  ]
19 }
```

Listing 1 – Structure JSON Pixabay