

# Cahier des charges - Application mobile intelligente Multi Plateforme

## 1. Titre du projet

**SmartFruit** : Application mobile intelligente pour la classification de fruits et assistance vocale intégrée

## 2. Contexte et objectif

L'objectif du projet est de concevoir et développer une **application mobile intelligente** Multi Plateforme capable de :

- Classifier des fruits à partir d'images capturées par la caméra ou sélectionnées depuis la galerie,
- Utiliser un modèle de Deep Learning (ANN / CNN) que l'étudiant aura entraîné et converti en TFLite,
- Intégrer un assistant vocal permettant d'interagir avec l'application via la voix : reconnaissance vocale, envoi des requêtes à une API d'IA (GPT ou Gemini) et restitution des réponses en voix, texte et image,
- Gérer l'accès utilisateur via un système d'authentification Firebase pour sécuriser l'usage.

Ainsi, ce projet vise à combiner vision par ordinateur, intelligence artificielle conversationnelle, et expérience utilisateur interactive dans une application mobile éducative et démonstrative.

## 3. Objectifs pédagogiques

À travers ce projet, l'étudiant devra :

1. Concevoir et entraîner un modèle de **classification d'images (fruits)** en utilisant les architectures **ANN** et **CNN** (TensorFlow / Keras).
2. Convertir le modèle entraîné en **TensorFlow Lite (TFLite)** pour une exécution sur mobile.
3. Intégrer le modèle dans une application Flutter pour l'inférence en temps réel.
4. Implémenter un **assistant vocal** combinant :
  - Reconnaissance vocale (SpeechToText),
  - Appel d'un modèle de langage (GPT ou Gemini),
  - Restitution de la réponse en texte, voix, et image.
5. Mettre en place une authentification Firebase et assurer la gestion sécurisée des utilisateurs.

## 4. Périmètre fonctionnel

### 4.1 Authentification Firebase

- Inscription / connexion via email et mot de passe.
- Récupération du mot de passe oublié.
- Redirection automatique vers l'écran principal après connexion.

### 4.2 Classification d'images (Fruit Recognition)

- Prendre une photo via la caméra ou importer depuis la galerie.
- Exécuter la classification sur l'image choisie en utilisant le modèle TFLite embarqué.
- Afficher :
  - Le nom du fruit prédit,
  - Le score de confiance (%),
  - Une image d'illustration du fruit

### 4.3 Assistant vocal intelligent

- Activation du micro ( commande vocale).
- Conversion voix → texte via Speech-to-Text.
- Envoi du texte à une API (GPT / Gemini).
- Réception de la réponse et restitution :
  - En **texte affiché**,
  - En **voix (Text-to-Speech)**,
  - En **image** (si la réponse en contient une ou si l'API le propose).

### 4.4. Menu principal

- **Accueil** (interface de classification de fruits).
- **Assistant vocal** (chat vocal / texte).
- **Historique des classifications** (optionnel).
- **Profil utilisateur**.
- **À propos / aide**.

## 5. Architecture technique proposée

### Frontend

- Framework : Flutter (Android & iOS).
- UI réactive avec un design simple et intuitif.
- Plugins :

- **image\_picker** (caméra / galerie),
- **tflite\_flutter** pour l'inférence du modèle TFLite,
- **speech\_to\_text** pour la reconnaissance vocale,
- **flutter\_tts** pour la synthèse vocale,
- **firebase\_auth** et **cloud\_firestore** pour l'authentification et les données utilisateurs.

## Backend

- Firebase Authentication : gestion des comptes utilisateurs.
- Firestore (optionnel) : stockage des historiques ou préférences.
- API externe (OpenAI GPT / Google Gemini) : réponses intelligentes de l'assistant vocal.

## Modèle de Deep Learning

- Modèle ANN ou CNN entraîné par l'étudiant en Python (TensorFlow / Keras).
- Conversion en TFLite.
- Intégration dans l'application via **assets/**.
- Chargement et inférence sur device avec **tflite\_flutter**.

## 6. Flux utilisateur

1. L'utilisateur crée un compte ou se connecte.
2. Il accède au menu principal et choisit :
  - Classifier un fruit :
    - Capture ou sélection d'une image,
    - L'application affiche la prédiction
  - Assistant vocal :
    - L'utilisateur parle,
    - L'assistant affiche et lit la réponse (et image si applicable).
3. (Optionnel) L'utilisateur consulte son historique ou modifie son profil.