



# Document de recette

## PRESS AI

**Application mobile de transformation d'interviews  
audio en articles journalistiques par IA**

**Projet** : BUT Informatique 2e - Développement mobile avec FlutterFlow

**Auteur** : Grégoire MAROUSE BUT 2B1

**Date** : Janvier 2026

# TABLE DES MATIÈRES

## TABLE DES MATIÈRES

### 1. PRÉSENTATION

Contexte du projet

La solution

Caractéristiques principales

### 2. PAGES DE L'APPLICATION

2.1 Login

2.2 Signup

2.3 Forgot Password

2.4 Dashboard

2.5 Generation Form

2.6 Editor

2.7 Settings

### 3. FONCTIONNALITÉS

3.1 Authentification

3.2 Génération d'articles IA

3.3 Édition et export

3.4 Bibliothèque et statistiques

3.5 Gestion des données

3.6 Sécurité

### 4. CRITÈRES DE VALIDATION

Tableau de synthèse

Contraintes

### 5. PERSPECTIVES D'AMÉLIORATION

5.1 Stockage et lecture audio

5.2 Optimisation adaptative des prompts

5.3 Modèles IA plus performants

5.4 Autres améliorations envisagées

### CONCLUSION

# 1. PRÉSENTATION

## Contexte du projet

Les journalistes passent en moyenne **4 à 6 heures** pour transformer une interview d'une heure en article publié (retranscription 4h, rédaction 2h). Press AI automatise ce processus grâce à l'IA, **réduisant le temps de production de 80%** : de 4-6h à moins d'1h.

## La solution

**Workflow** : Audio d'interview MP3 → Transcription IA (Groq Whisper) → Génération article (Groq Llama) → Édition → Export Markdown

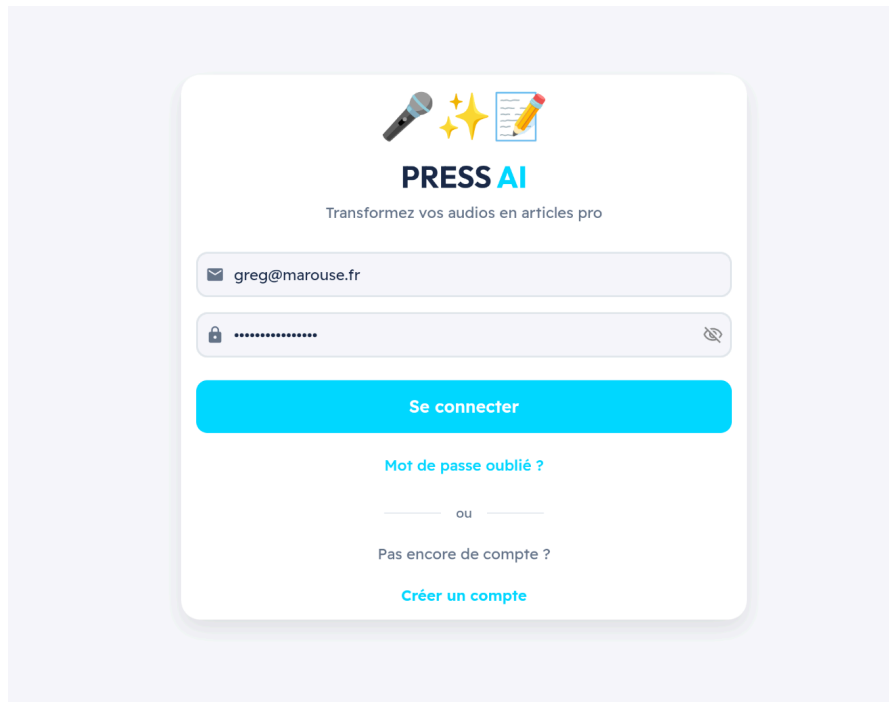
## Caractéristiques principales

Caractéristique	Valeur
Plateforme	Web (Responsive design)
Framework	FlutterFlow
Authentification	Firebase Authentication (Email/Password)
Base de données	Cloud Firestore (3 collections)
API IA	Groq Whisper Large v3 + Llama 3.3 70B
Custom Code	4 Custom Actions + 7 Custom Functions
Budget	0€ (100% free tier)

## 2. PAGES DE L'APPLICATION

### 2.1 Login

Connexion email/password avec Firebase Auth. Formulaire simple (email, password), liens vers réinitialisation et inscription. Redirection Dashboard après authentification réussie avec chargement articles et profil depuis Firestore.



Message d'erreur :

```
Error: The supplied auth credential is incorrect, malformed or has expired.
```

### 2.2 Signup

Inscription nouveau compte avec création automatique des documents Firestore **users** et **profiles** (préférences par défaut). Formulaire : nom complet, email, password (min 6 car), confirmation. Redirection Login après succès.

Messages d'erreur :

Error: The email is already in use by a different account

Passwords don't match!

Error: Password should be at least 6 characters

## 2.3 Forgot Password

Réinitialisation mot de passe via email. Saisie email, envoi lien temporaire (1h) par Firebase `sendPasswordResetEmail()`.

← Retour à la connexion

**Mot de passe oublié ?**

Entrez votre email et nous vous enverrons un lien

Adresse email  
gregoire@marouse.fr

Envoyer le lien --

💡 Vérifiez vos spams. Lien valide 10 min

Hello,

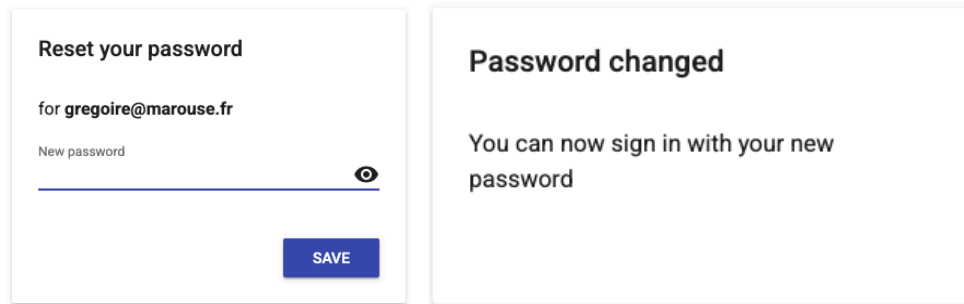
Follow this link to reset your presse-a-i-yjpx91 password for your [gregoire@marouse.fr](mailto:gregoire@marouse.fr) account.

[https://presse-a-i-yjpx91.firebaseio.com/\\_/auth/action?mode=resetPassword&oobCode=gUgV-FQPfMzeZSeiQm1jrGJCulh8WTVvQ53Yo2wmvMMAAAGbkljEmQ&apiKey=AlzaSyDfSP2QAseqdWw1yjCsGt0jaQuU4liRwYc&lang=en](https://presse-a-i-yjpx91.firebaseio.com/_/auth/action?mode=resetPassword&oobCode=gUgV-FQPfMzeZSeiQm1jrGJCulh8WTVvQ53Yo2wmvMMAAAGbkljEmQ&apiKey=AlzaSyDfSP2QAseqdWw1yjCsGt0jaQuU4liRwYc&lang=en)

If you didn't ask to reset your password, you can ignore this email.

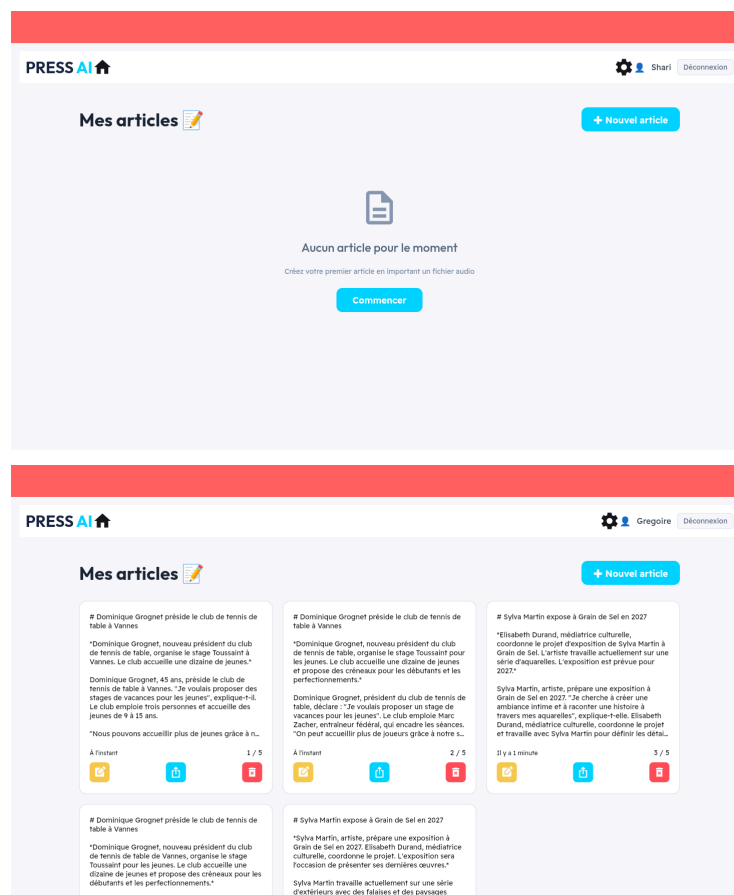
Thanks,

Your presse-a-i-yjpx91 team



## 2.4 Dashboard

Page d'accueil avec header permanent (logo, accueil, réglages, nom user, déconnexion). Affiche la grille responsive d'articles (2-3 colonnes). Cards avec titre, extrait, date formatée `smartDateDisplay`, boutons Voir/Modifier, Exporter et Supprimer. Tri par date décroissante. État vide si aucun article.



## 2.5 Generation Form

Configuration génération article. Section 1 : upload audio MP3 max 25MB. Section 2 : paramètres chargés depuis Firestore `profiles` (longueur slider 500-3000 synchronisé TextField via `AppState.targetLength`, ton, structure). Bouton génération avec loading. Navigation automatique vers l'Editor si succès.

The image shows two screenshots of the 'Nouveau projet' form in the PRESS AI application. Both screenshots show the same form with the following elements:

- Header:** PRESS AI logo and user profile (Gregoire) with a 'Déconnexion' button.
- Title:** Nouveau projet
- File Upload:** A box with a microphone icon and the text 'Glissez votre fichier audio ici' (MP3 - Max 25MB). The second screenshot shows a file named 'Fichier : Nouvel-enregistrement-4.mp3'.
- Longueur cible:** A slider between 500 and 3000 characters, currently set at 1200.
- Ton:** Radio buttons for Neutre, Dynamique, and Analytique (selected).
- Structure:** Radio buttons for Classique, Q&A (selected), and Reportage.
- Footer:** A blue button labeled 'Générer l'article'.

## 2.6 Editor

Édition article Markdown avec compteurs temps réel (caractères/mots via `getStats` et `AppState.counterWord / AppState.counterChar`). Trois boutons : "Nouvel article" (renvoie vers le Form sans enregistrement), "Enregistrer" (sauvegarde/update Firestore), "Exporter" (téléchargement article en .md). Distinction création/update via `AppState.refArticle`.

The image shows the 'Editor' interface in the PRESS AI application. It features a text area with Markdown content and three buttons at the bottom: 'Nouvel article', 'Enregistrer', and 'Exporter'.

**Header:** PRESS AI logo and user profile (Gregoire) with a 'Déconnexion' button.

**Character Counter:** 935 caractères / 106 mots

**Article Content:**

```
# Sylvie Martin expose à Grain de Sel en 2027

*Sylvie Martin, artiste, prépare une exposition à Grain de Sel en 2027. Elisabeth Durand, médiatrice culturelle, coordonne le projet. L'exposition sera l'occasion de présenter les dernières œuvres de Sylvie Martin.*

Sylvie Martin travaille sur une série d'œuvres avec des falaises et des paysages marins. "Je cherche à capturer l'essence de la Bretagne", déclare-t-elle. Elisabeth Durand, médiatrice culturelle, souligne l'importance de l'exposition : "Nous sommes ravis de présenter les œuvres de Sylvie Martin à Grain de Sel".

L'exposition est prévue pour 2027, mais la date exacte n'est pas encore fixée. Sylvie Martin explique : "J'ai besoin de temps pour préparer une exposition qui reflète mon univers". Elisabeth Durand précise : "Nous laisserons l'artiste libre de choisir les œuvres qu'elle souhaite présenter".

*Pratique*: Grain de Sel, lieu d'exposition à déterminer, 2027.
```

**Footer Buttons:** 'Nouvel article', 'Enregistrer', 'Exporter'.

## 2.7 Settings

Gestion profil, préférences et données. Cinq sections : **Mon compte** (modification nom), **Statistiques** de l'utilisateur, **Préférences par défaut** (longueur/ton/structure sauvegardés Firestore **profiles**), **Mes données** (export Markdown complet, suppression tous articles), **Gestion compte** (changement password Firebase, suppression compte avec séquence articles→profiles→users→Auth).

The screenshot shows the user settings page for 'Gregoire'. At the top left is the 'PRESS AI' logo with a home icon. At the top right is a settings gear icon, the name 'Gregoire', and a 'Déconnexion' button. The main content is organized into several sections:

- Profile Information:** A white box containing the name 'Gregoire', the email 'gregoire@marouse.fr', and an 'Enregistrer le profil' button.
- Statistics:** Three grey boxes showing: '5 Articles créés', '682 Mots écrits', and '3 Jours membre'.
- Default Preferences:** A white box with three dropdown menus: 'Longueur par défaut' (1200), 'Ton par défaut' (Analytique), and 'Structure par défaut' (Q&A). Below them is an 'Enregistrement automatique' status with a checkmark.
- Data Management:** A white box with two buttons: 'Exporter mes données' and 'Supprimer tous mes articles'.
- Account Management:** A white box with two password input fields ('Nouveau mot de passe' and 'Confirmer nouveau mot de passe'), a 'Changer mot de passe' button, and a 'Supprimer mon compte' button.

Suppression du compte :

Confirmation

Êtes-vous certain de vouloir supprimer votre compte ? Vos données seront perdues.

[Annuler](#) [Confirmer](#)

Too long since most recent sign in. Sign in again before deleting your account.

Redirection vers login si confirmé

Changer le mot de passe :

Confirmation

Êtes-vous sûr de vouloir changer votre mot de passe ?

[Annuler](#) [Confirmer](#)

Mot de passe

Mot de passe mis à jour.

[Ok](#)

Error: This operation is sensitive and requires recent authentication. Log in again before retrying this request.

Suppression des articles :

Confirmation

Êtes-vous certain de vouloir supprimer l'ensemble de vos articles ?

[Annuler](#) [Confirmer](#)

Redirection sur Dashboard vide si confirmé

Modification du mail :

Confirmation

Êtes-vous sûr de vouloir changer votre adresse e-mail ?

[Annuler](#) [Confirmer](#)

Too long since most recent sign in. Sign in again before updating your email.

## 3. FONCTIONNALITÉS

### 3.1 Authentification

Firebase Auth Email/Password avec méthodes `createUserWithEmailAndPassword` (inscription + création documents Firestore), `signInWithEmailAndPassword` (connexion + chargement données), `sendPasswordResetEmail` (reset), `updatePassword` (Settings), `deleteUser` (suppression), `signOut` (déconnexion). Protection pages privées : redirection automatique Login si non authentifié.

### 3.2 Génération d'articles IA

Upload audio MP3 (<25MB), Custom Action `transcribeAudioGroq` (multipart/form-data vers Whisper), Custom Action `generateArticleGroq` (prompt dynamique + appel Llama) et sortie d'un Markdown structuré. Configuration de la longueur, du ton, de la structure.

### 3.3 Édition et export

Custom Function `getStats` appelée à chaque frappe pour compteurs temps réel de mots et de caractères. Édition libre Markdown. Sauvegarde Firestore : création document `articles` si nouveau (`refArticle` null) ou update si existant. Export via Custom Function `createMarkdownFile` : génération FFUploadedFile `username_timestamp.md` encodé UTF-8, téléchargement automatique.

### 3.4 Bibliothèque et statistiques

Chaque utilisateur possède une bibliothèque d'articles qu'il a enregistrés grâce à une Query Firestore `articles.where('user_id', '=', currentUser.uid).orderBy('created_at', 'desc')` au login et après sauvegarde. Affichage grille cards avec Custom Functions : `getTitle` (extraction titre Markdown), `smartDateDisplay` (date relative/absolue), Suppression article export ou modification possible depuis la bibliothèque.

### 3.5 Gestion des données

Settings : modification longueur/ton/structure, update Firestore `profiles`. Export complet : Custom Action `exportUserDataAsMarkdown` (queries users+profiles+articles, compilation Markdown) → `createMarkdownFile`. Suppressions : boucle delete tous articles, ou séquence complète suppression compte (articles→profiles→users→Auth).

### 3.6 Sécurité

Isolation utilisateurs : queries filtrées par `user_id`. Test User A (3 articles) ≠ User B (2 articles). Firestore Security Rules : lecture/écriture uniquement si `request.auth.uid` correspond à propriétaire document.

## 4. CRITÈRES DE VALIDATION

### Tableau de synthèse

#	Critère imposé	Détail implémentation	Justification technique	Validé
1	<b>Stockage AppState</b>	7 AppState Variables FlutterFlow	<code>targetLength</code> , <code>wordCount</code> , <code>charCount</code> , <code>error</code> , <code>article</code> , <code>refArticle</code> , <code>articlesList</code> . Utilisées pour synchronisation slider↔textfield, compteurs temps réel, transit données inter-pages, stockage référence article sauvegardé.	✓
2	<b>Base de données</b>	Cloud Firestore (3 collections)	<b>users</b> (user_id, email, user_name, created_time), <b>profiles</b> (user reference, length, tone, structure), <b>articles</b> (user_id, title, body, target_length, tone, structure, transcription_text, created_at). Relations : profiles.user→users, articles.user_id→users. Firestore Security Rules déployées (isolation par user_id).	✓
3	<b>APIs externes</b>	2 APIs Groq	<b>Whisper Large v3</b> (transcription audio→texte, endpoint <code>/audio/transcriptions</code> , multipart/form-data, 3-8s) et <b>Llama 3.3 70B</b> (génération article, endpoint <code>/chat/completions</code> , JSON, temp=0, max_tokens=4000, 5-12s). Workflow total <30s. Choix Groq : gratuit (budget 0€), ultra-rapide, qualité suffisante.	✓
4	<b>Custom Actions</b>	4 actions Dart	<code>transcribeAudioGroq</code> (upload multipart/form-data Whisper), <code>generateArticleGroq</code> (construction prompt dynamique + appel Llama + parsing JSON), <code>exportUserDataAsMarkdown</code> (queries multiples Firestore + compilation Markdown), <code>createMarkdownFile</code> (génération FFUploadedFile encodé UTF-8). Nécessaires car API builder FlutterFlow limité (pas FFUploadedFile, pas logic complexe).	✓
5	<b>Custom Functions</b>	7 fonctions utilitaires Dart	<code>smartDateDisplay</code> (date relative/absolue), <code>getTitle</code> (extraction titre Markdown), <code>countAllWord</code> (comptage mots avec nettoyage Markdown), <code>validateLength</code> (validation 500-3000), <code>getStats</code> (retourne [chars, words]), <code>isOK</code> + <code>getStringFromJson</code> (parsing JSON dynamic, contournement limitations typage FlutterFlow).	✓
6	<b>Système login</b>	Firebase Authentication Email/Password	Méthodes : <code>createUserWithEmailAndPassword()</code> , <code>signInWithEmailAndPassword()</code> , <code>sendPasswordResetEmail()</code> , <code>updatePassword()</code> , <code>deleteUser()</code> , <code>signOut()</code> . Protection pages privées : redirection auto Login si non auth. Messages erreur natifs Firebase. Création simultanée documents users+profiles à l'inscription.	✓
7	<b>Page configuration</b>	Settings avec 5 sections	<b>Mon compte</b> (modification nom), <b>Statistiques</b> utilisateur, <b>Préférences</b> (longueur/ton/structure sauvegardés Firestore <code>profiles</code> , chargés auto prochain article), <b>Données</b> (export Markdown complet via <code>exportUserDataAsMarkdown</code> + suppression articles), <b>Compte</b> (changement password <code>updatePassword()</code> + suppression séquence articles→profiles→users→Auth).	✓

## Contraintes

Contrainte	Description
<b>Budget 0€</b>	Groq APIs free tier (6000/min Whisper, 30/min Llama), Firebase free tier (Auth illimité, Firestore 1GB), Flutter Flow plan étudiant
<b>Login obligatoire</b>	Firebase Auth Email/Password (voir critère 6)
<b>APIs externes</b>	Dépendant des 2 APIs Groq (voir critère 3)
<b>Format audio</b>	MP3 max 25MB
<b>Langue</b>	Français uniquement

## 5. PERSPECTIVES D'AMÉLIORATION

### 5.1 Stockage et lecture audio

**Problématique actuelle** : Les fichiers audio ne sont pas conservés après traitement, empêchant leur réécoute en lien avec l'article généré.

**Amélioration proposée** : Implémenter le stockage des fichiers audio dans Firebase Storage avec référence dans Firestore `articles.audio_url`. Ajouter un lecteur audio dans l'éditeur permettant de réécouter l'interview originale pendant l'édition. Cette fonctionnalité faciliterait la vérification de citations et la correction d'éventuelles erreurs de transcription.

**Contrainte** : Firebase Storage free tier limité (5GB), nécessiterait compression audio optimale ou passage plan payant pour usage intensif.

**Problème lié** : L'API de transcription ne prend les audio que jusqu'à 25MB idéalement il faudrait les compresser et/ou avoir un algorithme de segmentation intelligente. Cette limite est assez contraignante et ne permet pas d'avoir des audio d'une heure ou plus. Hors ces pour ces audio long que le traitement automatique devient le plus intéressant en gain de temps.

### 5.2 Optimisation adaptative des prompts

**Amélioration proposée** : Mettre en place un système d'apprentissage des préférences utilisateur. Analyser les modifications apportées par l'utilisateur aux articles générés (suppressions, ajouts, restructurations) pour affiner progressivement les prompts. Stocker des "profils éditoriaux" dans Firestore incluant des exemples d'articles validés par l'utilisateur, utilisés comme références dans les prompts ("Génère un article dans le style des exemples suivants...").

**Bénéfice attendu** : Réduction du temps d'édition manuelle de 30-50% après 10-15 articles générés.

## 5.3 Modèles IA plus performants

**Constat** : Llama 3.3 70B (Groq) produit des articles de qualité acceptable mais l'utilisation de modèles plus avancés améliorerait significativement la qualité rédactionnelle.

**Alternatives identifiées** :

- **Gemini 2.0 Flash** (Google) : Meilleure compréhension du contexte français, style plus naturel
- **Claude 3.5 Sonnet** (Anthropic) : Excellence en rédaction longue, respect strict des consignes
- **GPT-4o** (OpenAI) : Polyvalence maximale, créativité contrôlée

**Idée** : Implémenter un système hybride avec Llama 3.3 gratuit par défaut, et option premium permettant de sélectionner Claude 3.5 Sonnet ou GPT-4o pour les articles les plus importants. Modèle freemium : l'utilisateur paie uniquement les articles premium générés.

**Aspect économique** : Pour un journaliste produisant 20 articles/mois avec modèle premium, le coût reste largement inférieur au temps gagné (80h/mois × taux horaire).

## 5.4 Autres améliorations envisagées

- **Mode collaboratif** : Partage articles entre utilisateurs pour relecture
- **Export formats avancés** : PDF, Word avec styles, etc.
- **Dictaphone in-app** : Capturer interviews directement depuis l'application
- **Support multi-langues** : Transcription et génération en anglais, espagnol, allemand
- **Intégration CMS** : Publication directe WordPress, Medium, etc.
- **Amélioration de l'éditeur** avec des options de mise en forme
- **Ajout d'un Agent IA** : Copilot Rédactionnel

# CONCLUSION

L'application Press AI respecte les critères imposés et atteint son objectif principal : réduire le temps de production journalistique. Le workflow audio→article est efficace, l'architecture technique est robuste (Firebase + Groq APIs + Custom Code), et le budget 0€ est respecté pour cette V1 avec des possibilités d'évolution.

Les perspectives d'amélioration identifiées (stockage audio, optimisation prompts, modèles IA avancés) positionnent Press AI comme une solution évolutive pouvant être un produit commercial viable.