

REPUBLIQUE DU SENEGAL
UNIVERSITE CHEIKH ANTA DIOP DE DAKAR



FACULTE DES SCIENCES ET TECHNIQUES
DEPARTEMENT MATHÉMATIQUES ET INFORMATIQUE

MASTER SYSTEMES D'INFORMATION REPARTIS

Développement Mobile
Projet Flutter- Application Todo

Année universitaire 2024-2025

REALISÉ PAR

Harouna MBAYE

SOUS LA DIRECTION DE

Dr. Moustapha DIOUF FALL

Liste des acronymes

Acronyme	Signification
API	Application Programming Interface – Interface de programmation permettant la communication entre l’application Flutter et le backend PHP/MySQL
CRUD	Create, Read, Update, Delete – Opérations de base sur les données (utilisées pour la gestion des tâches)
DB / BDD	Database / Base de Données – Ensemble structuré de données, ici MySQL
GPS	Global Positioning System – Système de positionnement par satellites utilisé pour la géolocalisation
HTTP	HyperText Transfer Protocol – Protocole de communication entre client et serveur
JSON	JavaScript Object Notation – Format d’échange de données entre Flutter et l’API
PHP	Hypertext Preprocessor – Langage côté serveur utilisé pour développer l’API backend
SQL	Structured Query Language – Langage de requête pour gérer la base de données relationnelle
UI	User Interface – Interface utilisateur de l’application
UX	User Experience – Expérience vécue par l’utilisateur lors de l’utilisation de l’application
VS Code	Visual Studio Code – Éditeur de code utilisé pour développer l’application Flutter
XAMPP	Cross-Platform, Apache, MariaDB/MySQL, PHP, Perl – Environnement de développement local pour exécuter l’API

Liste des Figures

Figure 1: Création du projet Flutter	5
Figure 2: Inscription d'un utilisateur	6
Figure 3: Authentification du user	6
Figure 4: Accès à la page d'accueil de l'API via le navigateur.....	7
Figure 5: Page d'inscription d'un utilisateur.....	8
Figure 6: Connexion de l'utilisateur	9
Figure 7: Déconnexion de l'utilisateur	10
Figure 8: Menu latéral.....	11
Figure 9: Page d'accueil de l'app.....	11
Figure 10: Création d'une tâche en ligne	12
Figure 11: Création d'une tâche hors ligne	13
Figure 12: Vérification dans la base de données	14
Figure 13: Affichage des tâches créées	15
Figure 14: Modification d'une tâche	16
Figure 15: Affichage de la modification	17
Figure 16: Tâche modifiée avec succès	17
Figure 17: Accomplir une tâche	18
Figure 18: Tâche accomplie avec succès	18
Figure 19: Supprimer une tâche	19
Figure 20: Avant suppression	19
Figure 21: Après suppression	20
Figure 22: Champ de recherche.....	20
Figure 23: Historique des tâches accomplies	21
Figure 24: Choisir un photo de profil	22
Figure 25: Géolocalisation.....	23
Figure 26: Météo	24

Table des matières

Liste des Figures.....	3
Introduction.....	5
Chapitre 1 - Mise en place et Création du projet	5
1.1 Création du projet Flutter (terminal).....	5
1.2 Test avec Postman.....	6
1.3 Accès à la page d'accueil de l'API via le navigateur.....	7
Chapitre 2 - Conception Générale	7
2.1 Objectifs	7
2.2 Technologies utilisées.....	7
Chapitre 3 - Gestion des Comptes	7
3.1 Inscription	7
3.2 Connexion	8
3.3 Déconnexion	9
Chapitre 4 - Interface Utilisateur (Page d'Accueil).....	10
Chapitre 5 - Gestion des Tâches	11
5.1 Création	11
5.2 Affichage	14
5.3 Modification	15
5.4 Accomplir une tâche.....	17
5.5 Suppression.....	19
5.6 Recherche	20
5.7 Historique.....	21
Chapitre 6 - Fonctionnalités Additionnelles	21
6.1 Photo de Profil.....	21
6.2 Géolocalisation.....	22
6.3 Météo	23
Conclusion	24

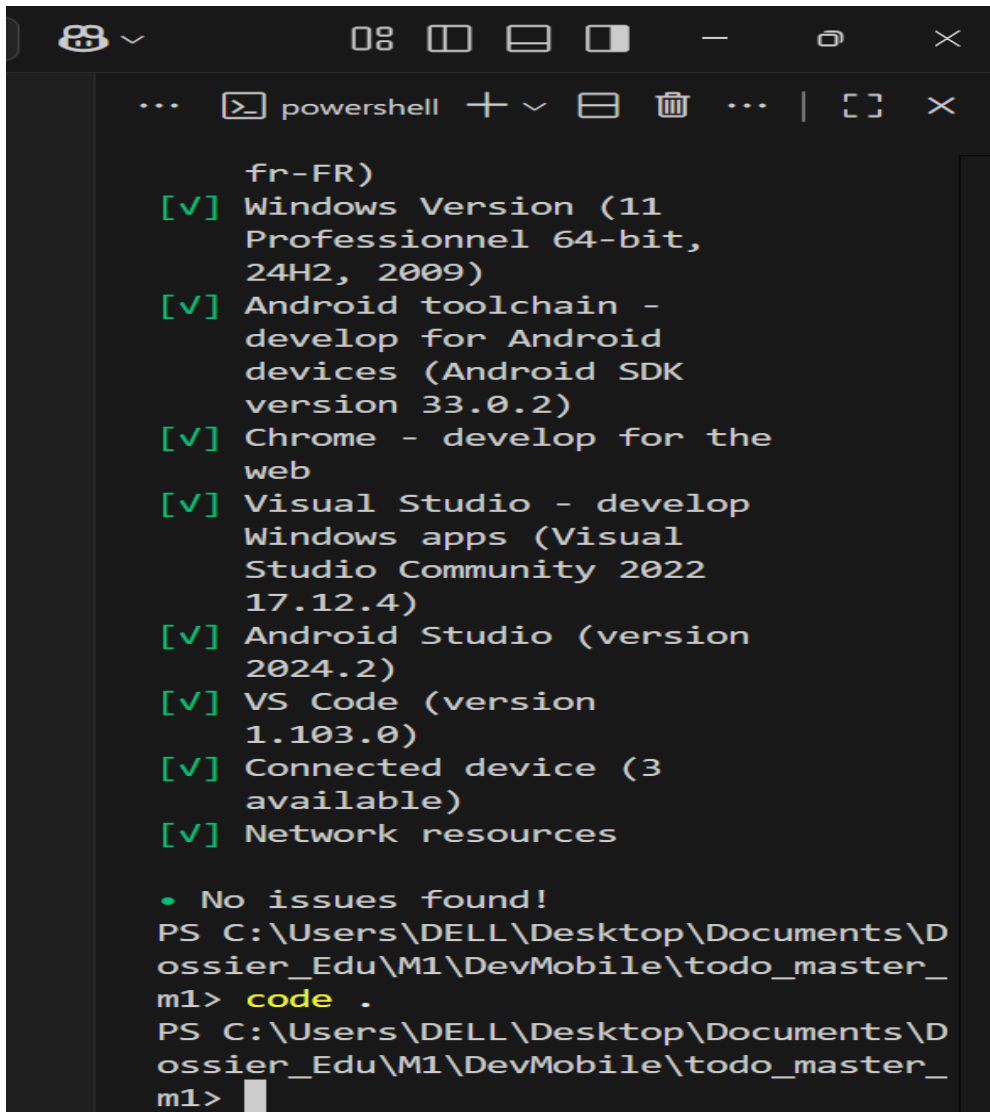
Introduction

Ce projet a pour objectif de concevoir et développer une application mobile de gestion des tâches en utilisant Flutter et le design pattern Provider. L'application permet à chaque utilisateur de gérer son quotidien via un compte personnel, avec des fonctionnalités de gestion des tâches, de géolocalisation et de météo.

Chapitre 1 - Mise en place et Création du projet

1.1 Création du projet Flutter (terminal)

```
flutter create todo_master_m1
cd todo_master_m1
```



```
fr-FR)
[✓] Windows Version (11
    Professionnel 64-bit,
    24H2, 2009)
[✓] Android toolchain -
    develop for Android
    devices (Android SDK
    version 33.0.2)
[✓] Chrome - develop for the
    web
[✓] Visual Studio - develop
    Windows apps (Visual
    Studio Community 2022
    17.12.4)
[✓] Android Studio (version
    2024.2)
[✓] VS Code (version
    1.103.0)
[✓] Connected device (3
    available)
[✓] Network resources

• No issues found!
PS C:\Users\DELL\Desktop\Documents\DevMobile\todo_master_
m1> code .
PS C:\Users\DELL\Desktop\Documents\DevMobile\todo_master_
m1>
```

Figure 1: Création du projet Flutter

1.2 Test avec Postman

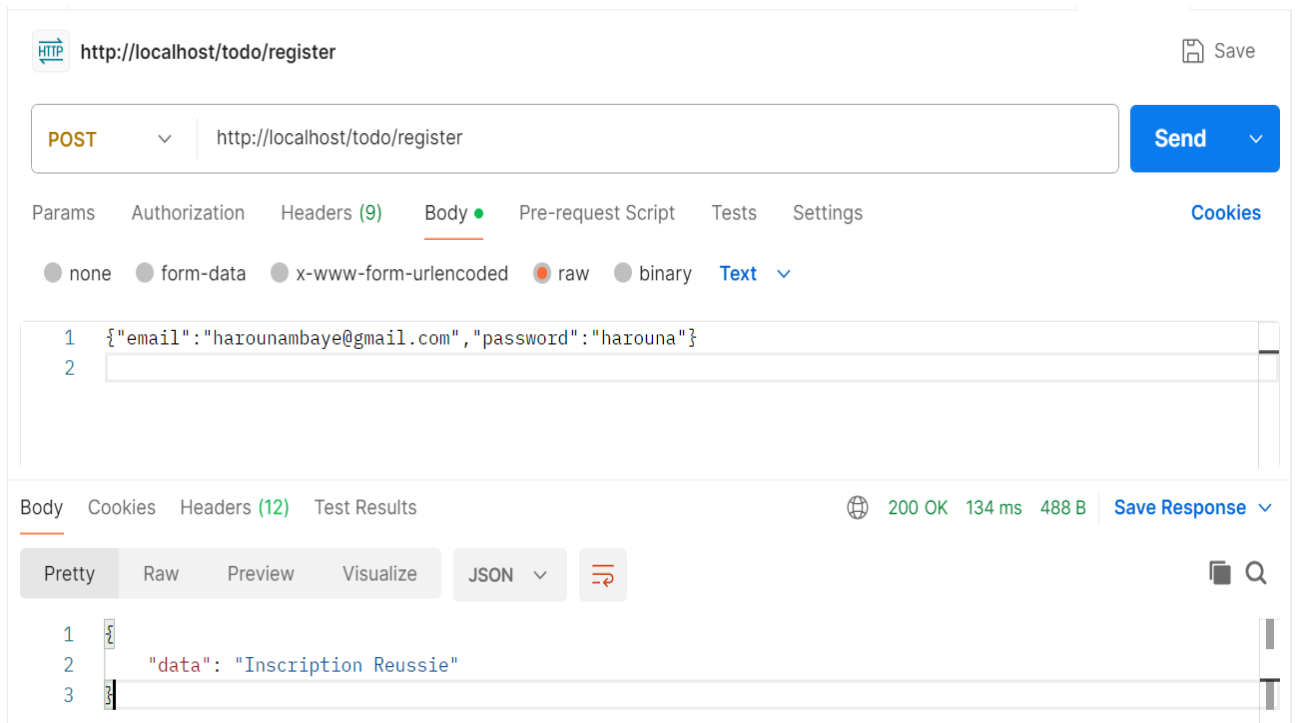


Figure 2: Inscription d'un utilisateur

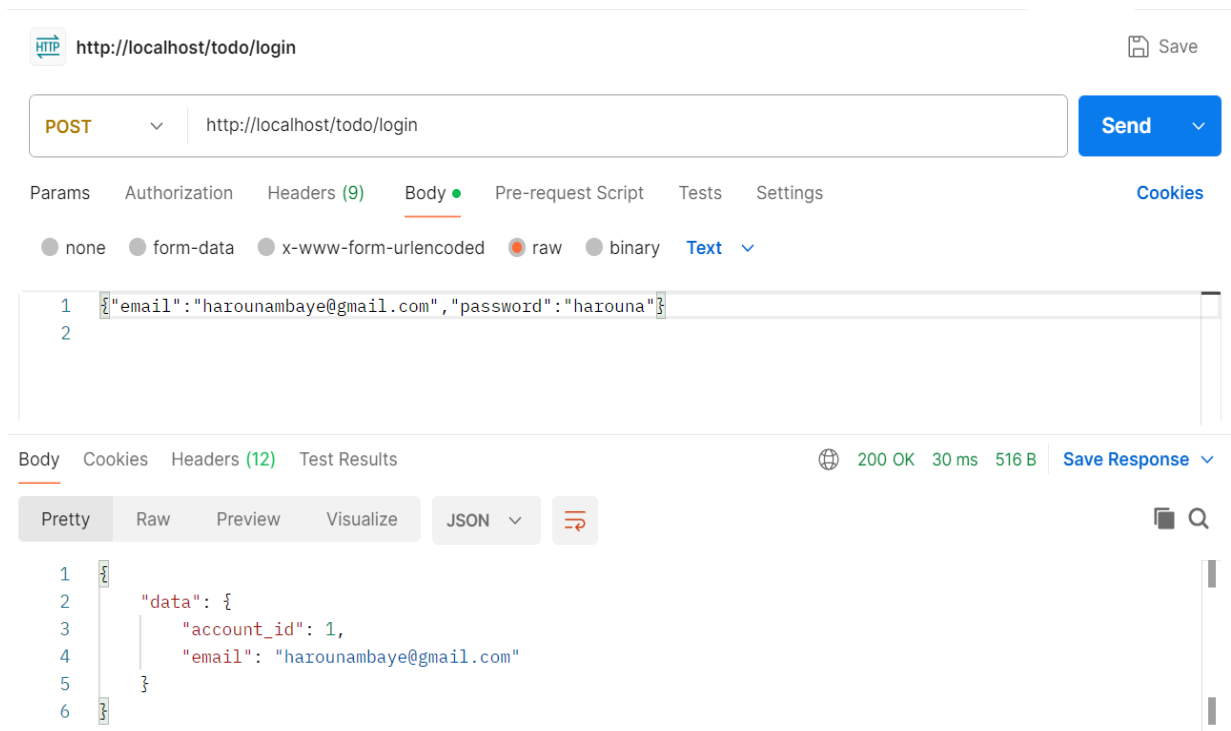


Figure 3: Authentification du user

1.3 Accès à la page d'accueil de l'API via le navigateur

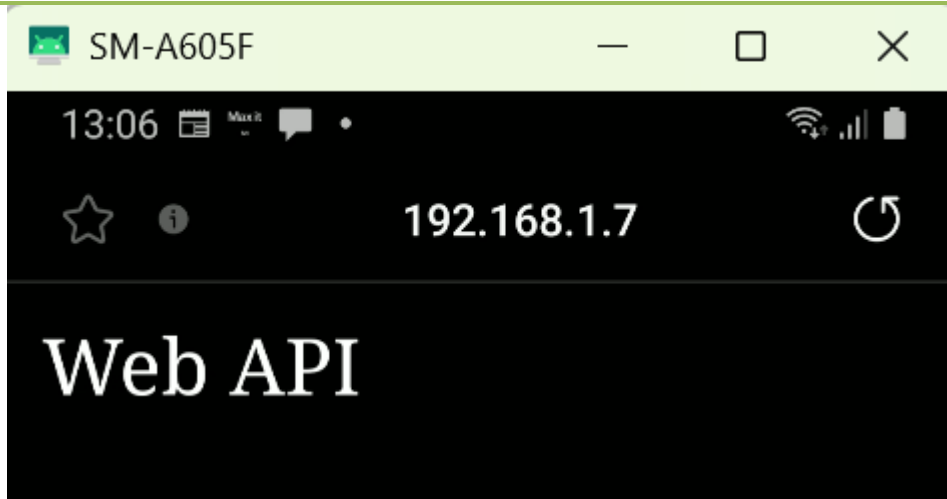


Figure 4: Accès à la page d'accueil de l'API via le navigateur

Chapitre 2 - Conception Générale

2.1 Objectifs

L'objectif principal est de fournir à l'utilisateur un outil moderne, ergonomique et complet pour:

- Créer et gérer ses tâches quotidiennes
- Suivre l'avancement (accomplissement / historique)
- Visualiser sa position géographique
- Consulter la météo locale
- Gérer son profil (photo, déconnexion sécurisée).

2.2 Technologies utilisées

- Flutter (Framework UI multiplateforme)
- Provider (Gestion d'état)
- PHP/MySQL (API backend et base de données)
- XAMPP (hébergement API locale)
- Postman (tests API)
- Google Fonts (UI moderne)

Chapitre 3 - Gestion des Comptes

3.1 Inscription

L'inscription permet à un nouvel utilisateur de créer un compte en fournissant un email et un mot de passe. Le backend PHP vérifie si l'email existe déjà, puis insère le nouvel utilisateur dans la base de données.

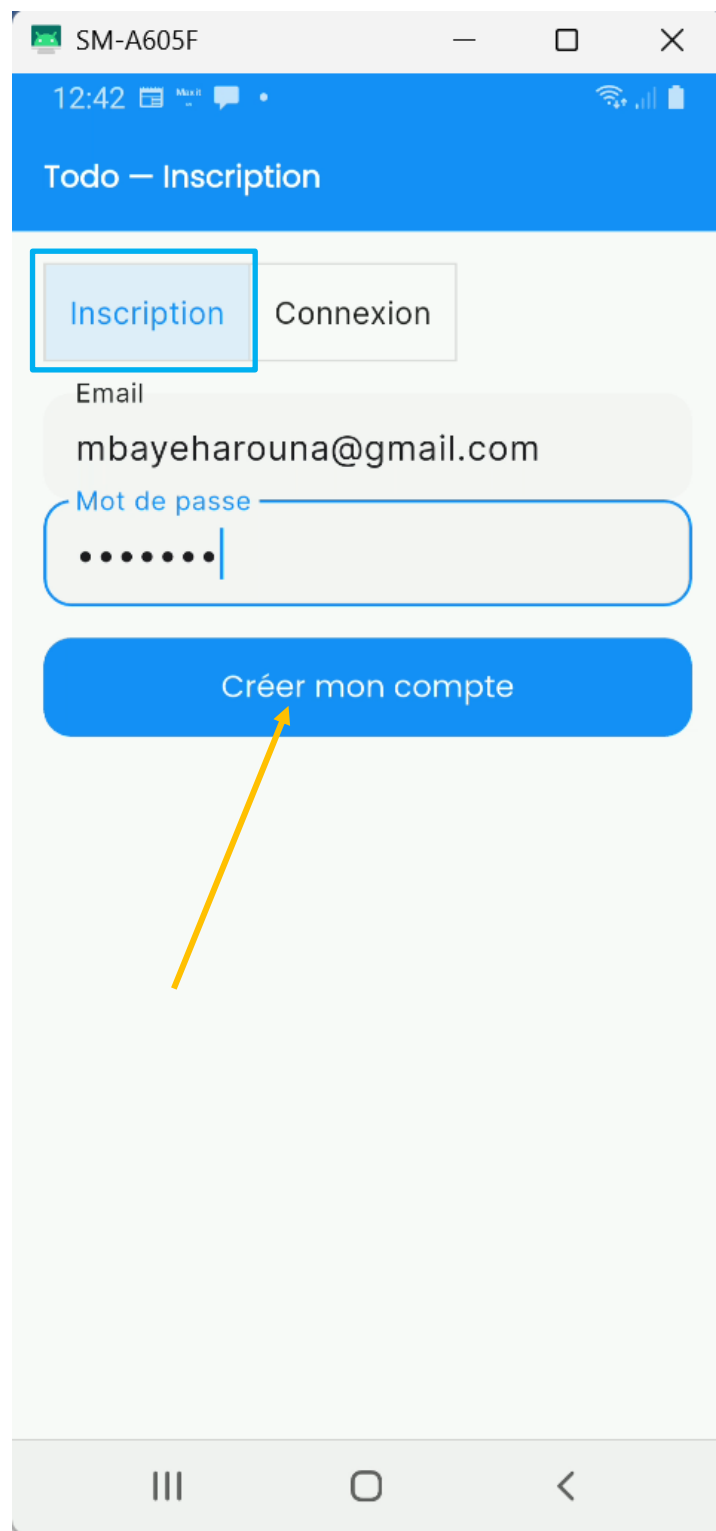


Figure 5: Page d'inscription d'un utilisateur

3.2 Connexion

La connexion permet à un utilisateur existant de s'authentifier en saisissant son email et mot de passe. L'API retourne l'identifiant unique de l'utilisateur, stocké localement afin de maintenir la session.

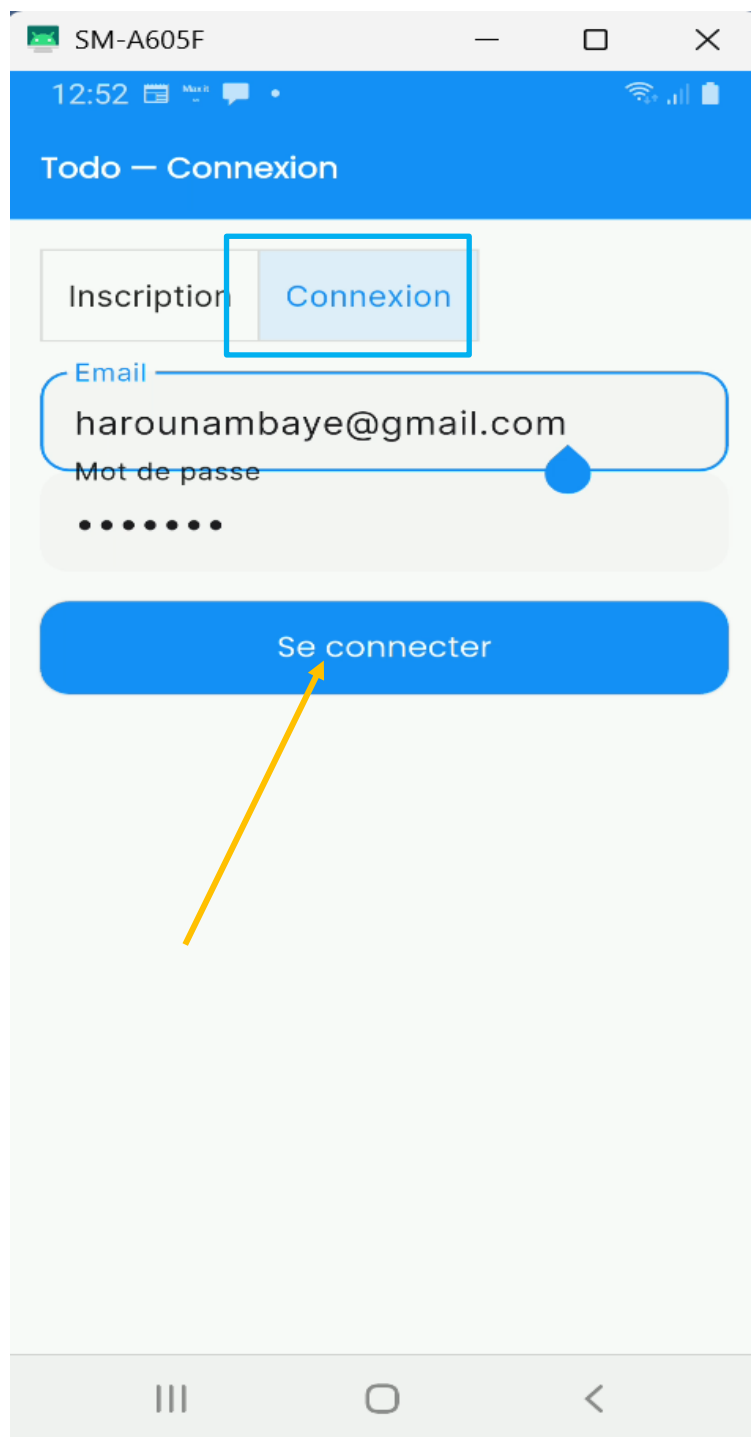


Figure 6: Connexion de l'utilisateur

3.3 Déconnexion

La déconnexion supprime les informations de session stockées localement afin de sécuriser l'accès. L'utilisateur est redirigé vers l'écran de connexion.

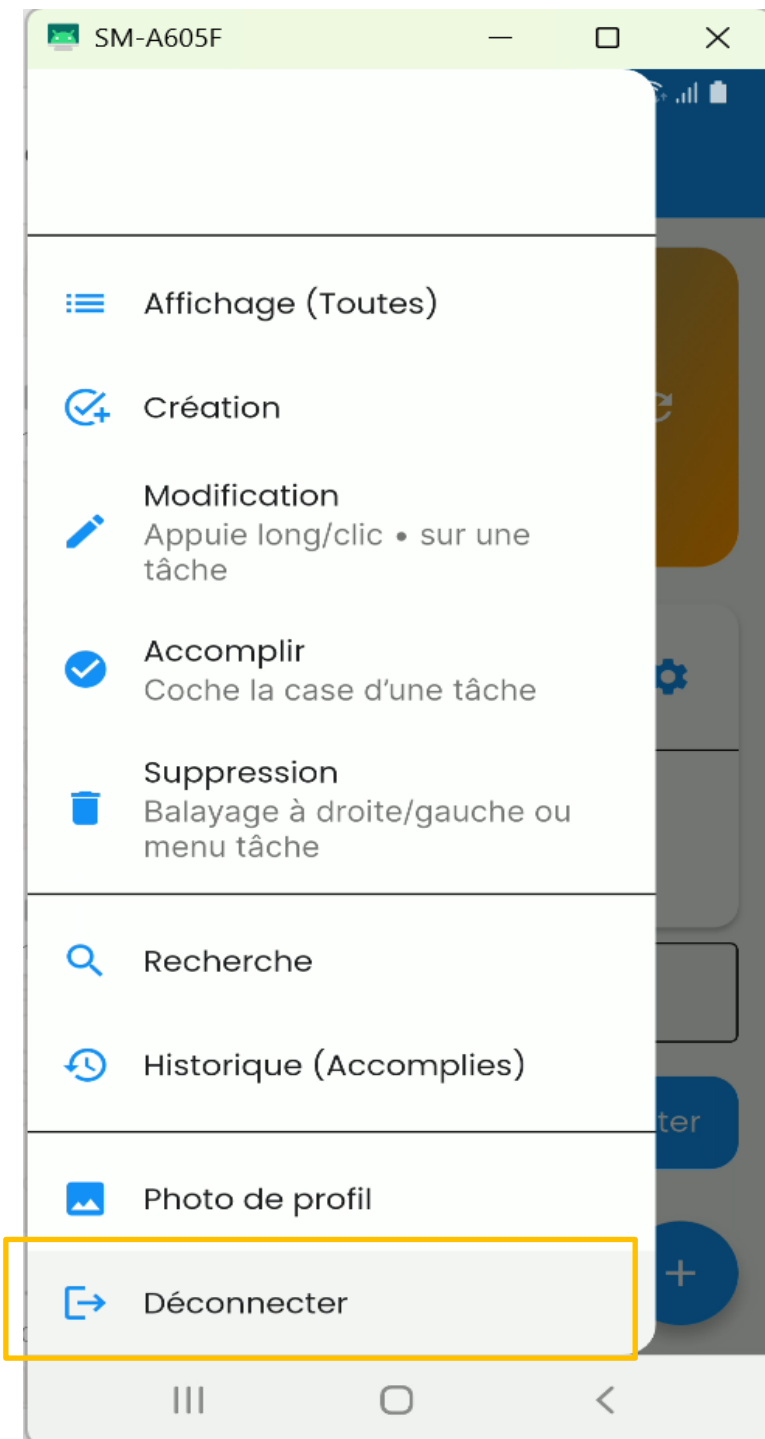


Figure 7: Déconnexion de l'utilisateur

Chapitre 4 - Interface Utilisateur (Page d'Accueil)

L'interface suit un design moderne basé sur Material 3 et Google Fonts. Les principales pages sont:

- Page de connexion/inscription
- Page d'accueil (tâches + météo + profil)
- Menu latéral (Drawer) pour accéder aux différentes fonctionnalités
- Dialogues pour création et modification de tâches.



Figure 9: Page d'accueil de l'app

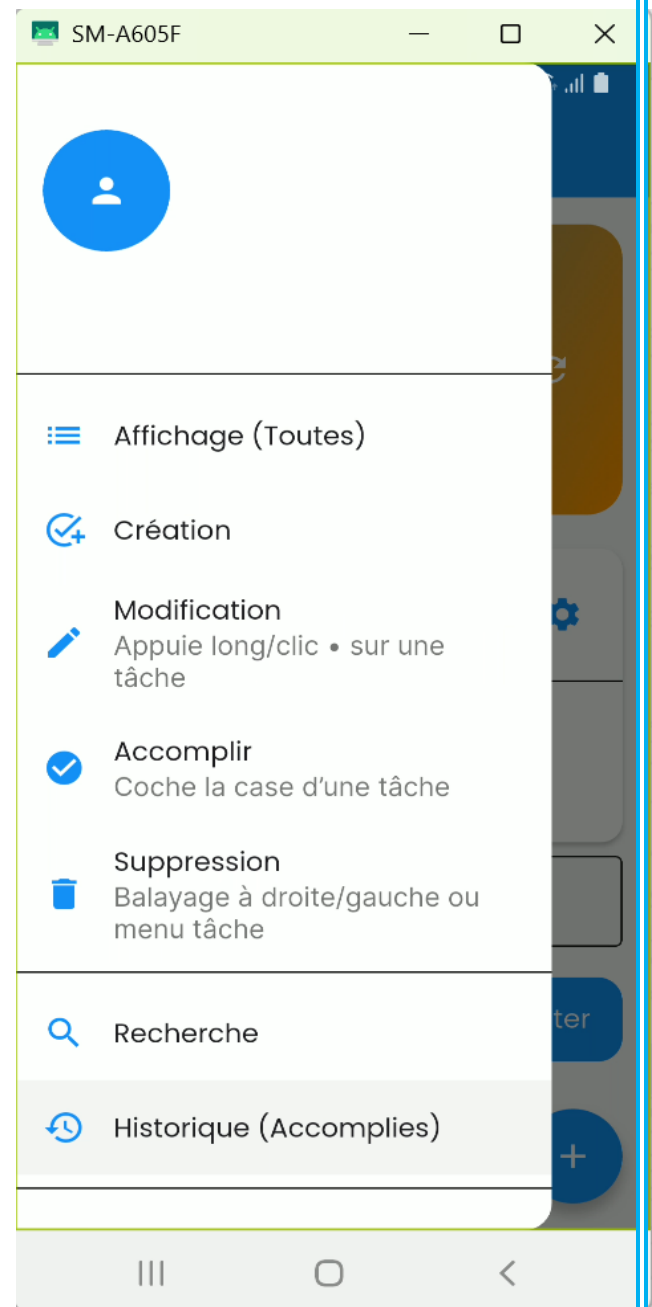


Figure 8: Menu latéral

Chapitre 5 - Gestion des Tâches

5.1 Création

Un utilisateur peut créer une tâche, même hors ligne. La tâche est stockée en local avec un indicateur de synchronisation, puis envoyée au serveur lorsque la connexion est disponible.



Figure 10: Création d'une tâche en ligne



Figure 11: Création d'une tâche hors ligne

localhost/phpmyadmin/index.php?route=/sql&db=todo_db&table=todo_tables&pos=0

Server: localhost:3306 » Database: todo_db » Table: todo_tables

Showing rows 0 - 2 (3 total, Query took 0.0005 seconds.)

SELECT * FROM `todo_tables`

Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

UPDATE `todo_tables` SET `todo` = 'Tache créée ade Talla NDIAYE' WHERE `todo_tables`.`todo_id` = 29;

[Edit inline] [Edit] [Create PHP code]

Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

Extra options

		todo_id	account_id	date	todo	done
<input type="checkbox"/>	Edit Copy Delete	31	1	2025-08-23	Acheter du café et du sucre	0
<input type="checkbox"/>	Edit Copy Delete	32	1	2025-08-23	Réviser Flutter et tester l'API	0

Check all | With selected: Edit Copy Delete Export

Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

Figure 12: Vérification dans la base de données

5.2 Affichage

L'écran principal affiche la liste des tâches en fonction du mode choisi: toutes les tâches ou seulement les accomplies (historique).



Figure 13: Affichage des tâches créées

5.3 Modification

L'utilisateur peut modifier le texte ou la date d'une tâche. La modification est appliquée localement et envoyée au backend pour mise à jour dans la base de données.

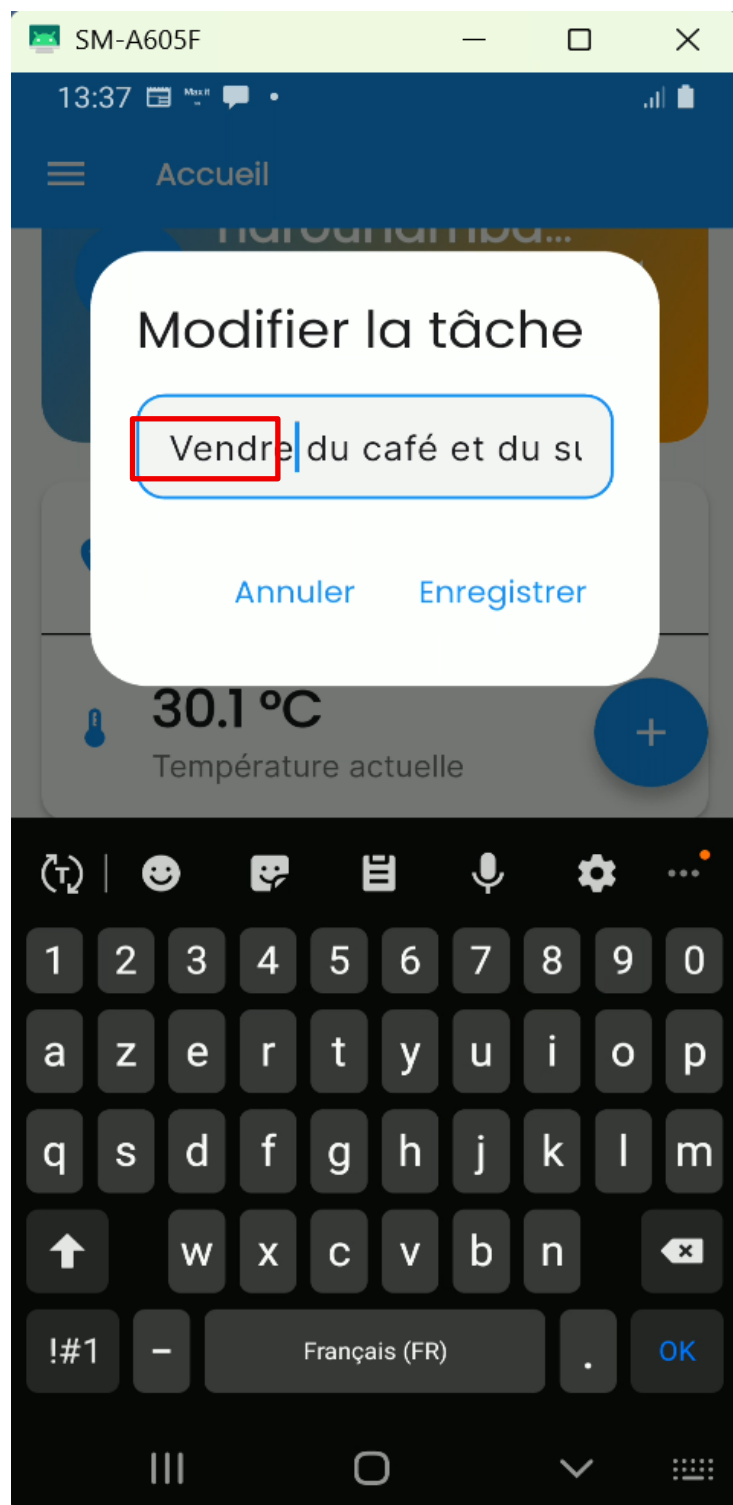


Figure 14: Modification d'une tâche



Figure 15: Affichage de la modification

				todo_id	account_id	date	todo	done
<input type="checkbox"/>	Edit	Copy	Delete	31	1	2025-08-23	Vendre du café et du sucre	0
<input type="checkbox"/>	Edit	Copy	Delete	32	1	2025-08-23	Réviser Flutter et tester l'API	0

Figure 16: Tâche modifiée avec succès

5.4 Accomplir une tâche

Chaque tâche peut être cochée comme accomplie via une case à cocher. L'état (fait / non fait) est synchronisé avec le serveur.



Figure 17: Accomplir une tâche

	todo_id	account_id	date	todo	done
<input type="checkbox"/>	31	1	2025-08-23	Vendre du café et du sucre	0
<input checked="" type="checkbox"/>	32	1	2025-08-23	Réviser Flutter et tester l'API	1

Click the drop-down arrow to toggle column's visibility.

↑ Check all With selected: Edit Copy Delete Export

Figure 18: Tâche accomplie avec succès

5.5 Suppression

Une tâche peut être supprimée via un balayage gauche/droite ou via un menu contextuel. La suppression est appliquée localement et côté serveur.

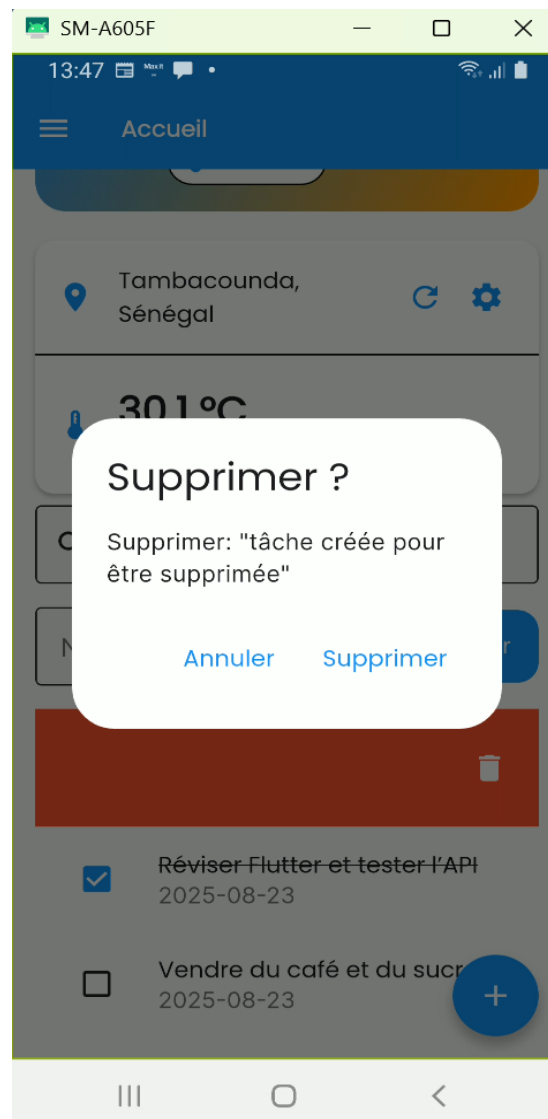


Figure 19: Supprimer une tâche

		todo_id	account_id	date	todo	done
<input type="checkbox"/>	Edit Copy Delete	31	1	2025-08-23	Vendre du café et du sucre	0
<input type="checkbox"/>	Edit Copy Delete	32	1	2025-08-23	Réviser Flutter et tester l'API	1
<input type="checkbox"/>	Edit Copy Delete	33	1	2025-08-23	tâche créée pour être supprimée	0

Figure 20: Avant suppression

←T→		todo_id	account_id	date	todo	done
<input type="checkbox"/>	Edit Copy Delete	31	1	2025-08-23	Vendre du café et du sucre	0
<input type="checkbox"/>	Edit Copy Delete	32	1	2025-08-23	Réviser Flutter et tester l'API	1

☐ Check all With selected: Edit Copy Delete Export

Figure 21: Après suppression

5.6 Recherche

Un champ de recherche permet de filtrer les tâches par mot-clé en temps réel.

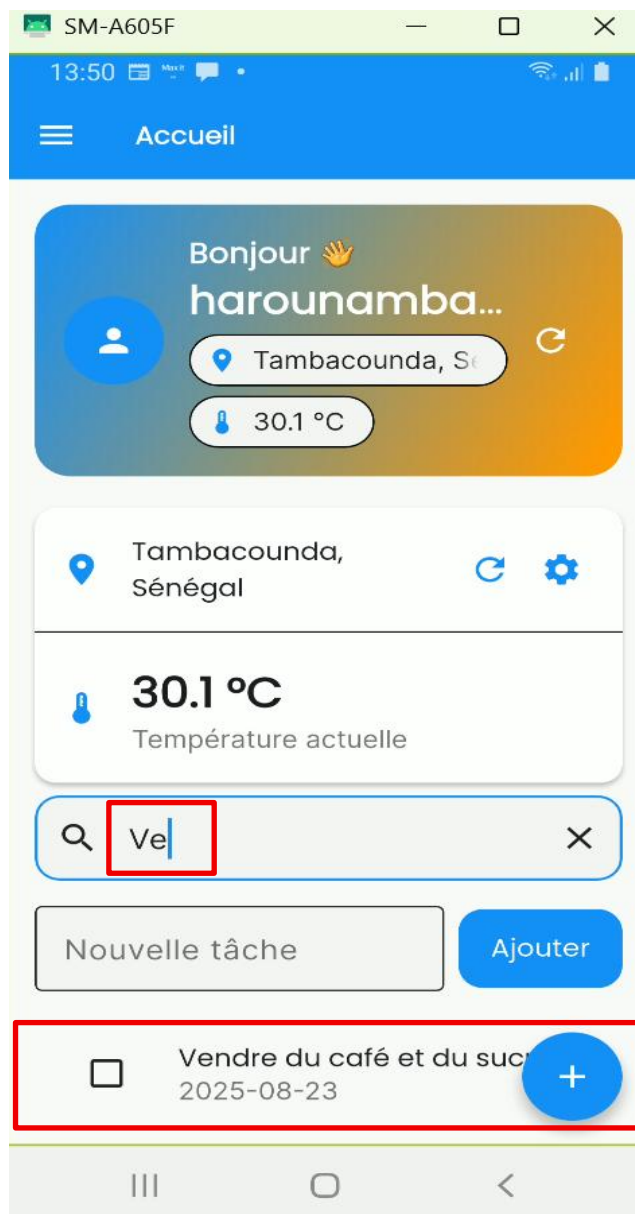


Figure 22: Champ de recherche

5.7 Historique

L'utilisateur peut accéder à l'historique des tâches accomplies, stockées localement pour consultation même sans connexion internet.

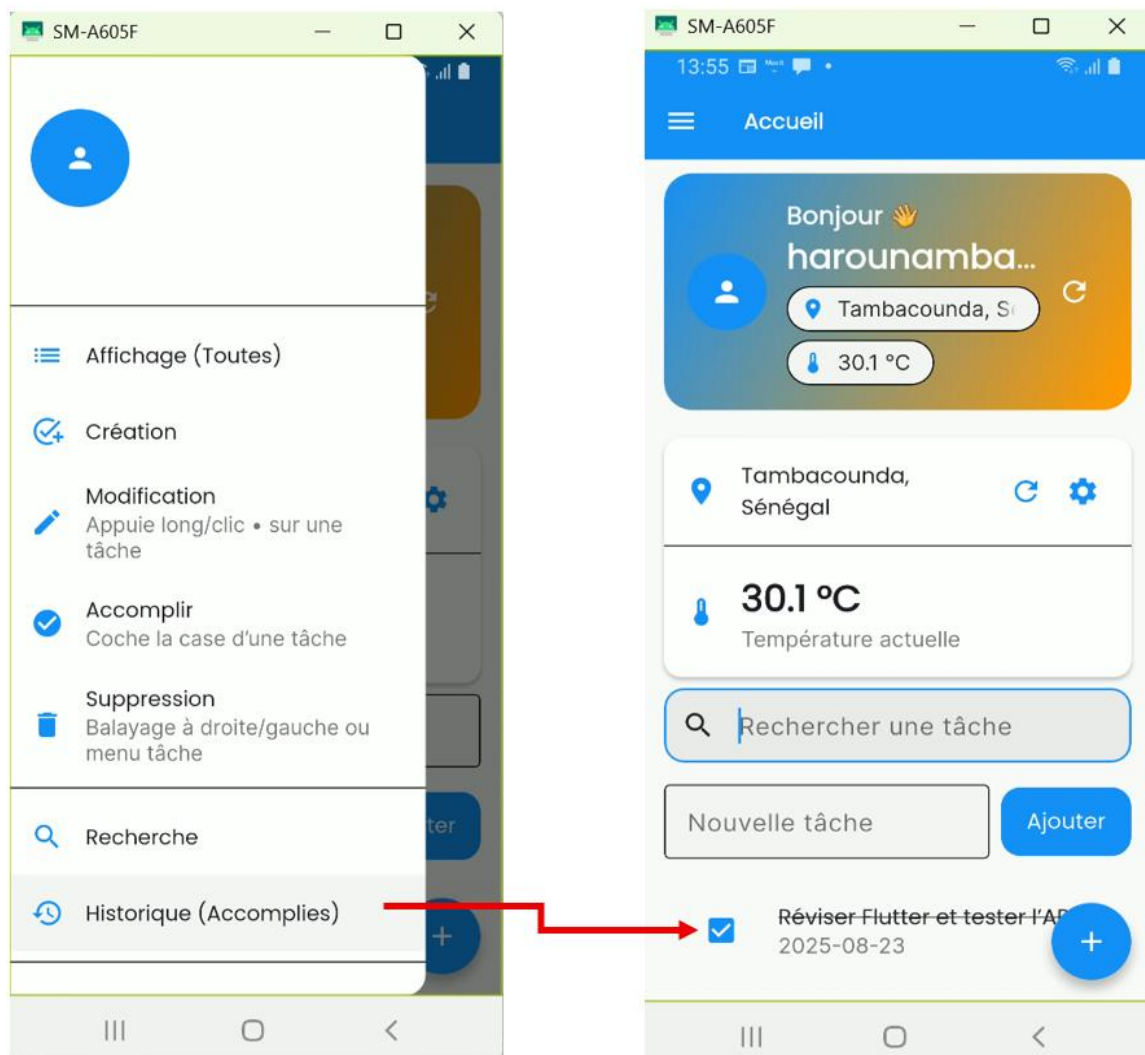


Figure 23: Historique des tâches accomplies

Chapitre 6 - Fonctionnalités Additionnelles

6.1 Photo de Profil

L'utilisateur peut sélectionner une photo depuis son appareil et l'associer à son profil. La photo est persistée localement même après déconnexion.

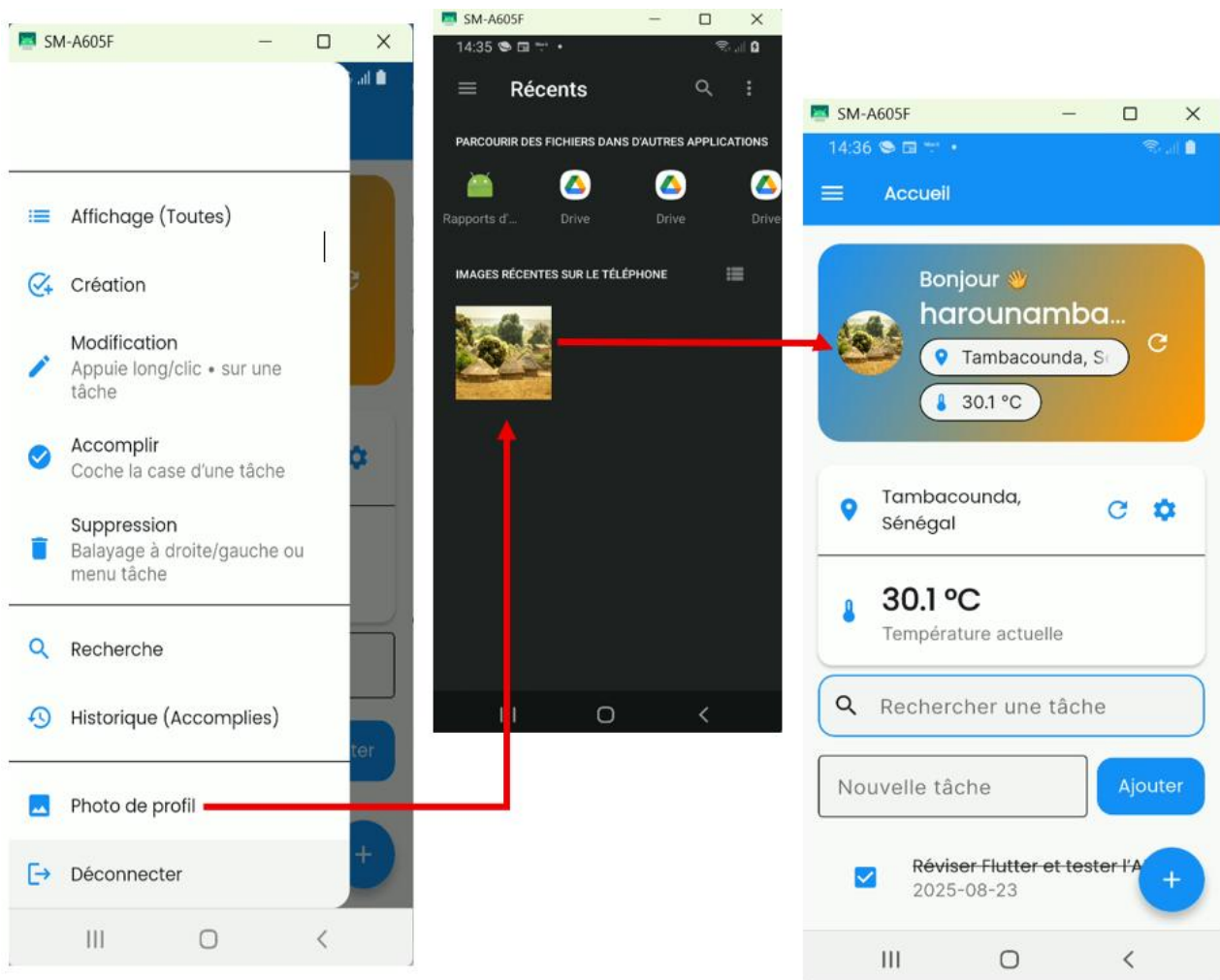


Figure 24: Choisir un photo de profil

6.2 Géolocalisation

L'application récupère la position géographique de l'utilisateur grâce à l'API Geolocator. Les coordonnées (latitude, longitude) sont converties en lieu lisible (ville, pays) via un service de reverse geocoding.

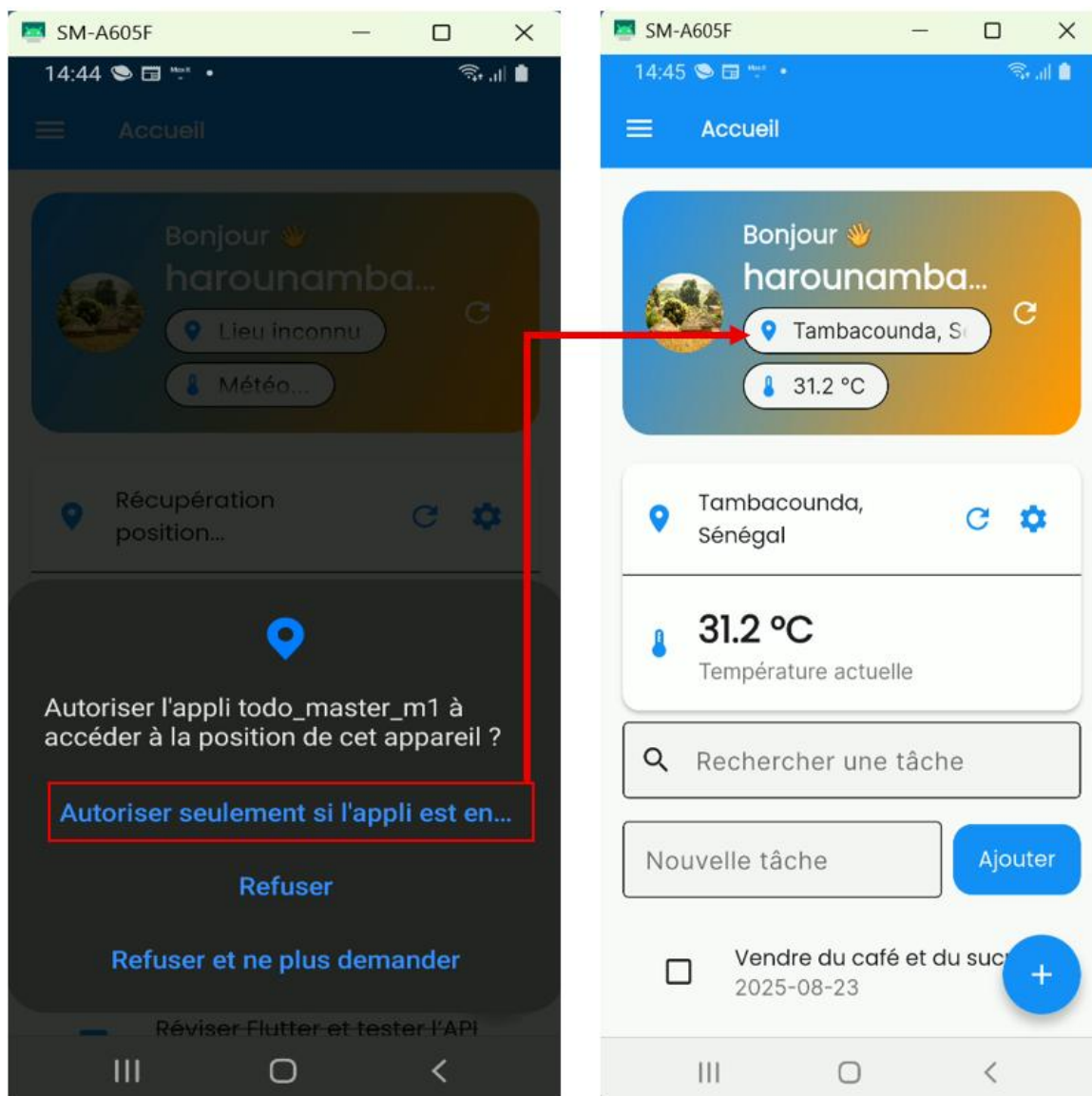


Figure 25: Géolocalisation

6.3 Météo

L'application affiche la température actuelle en fonction de la position GPS de l'utilisateur. L'API open-meteo est utilisée pour récupérer les données météorologiques en temps réel.

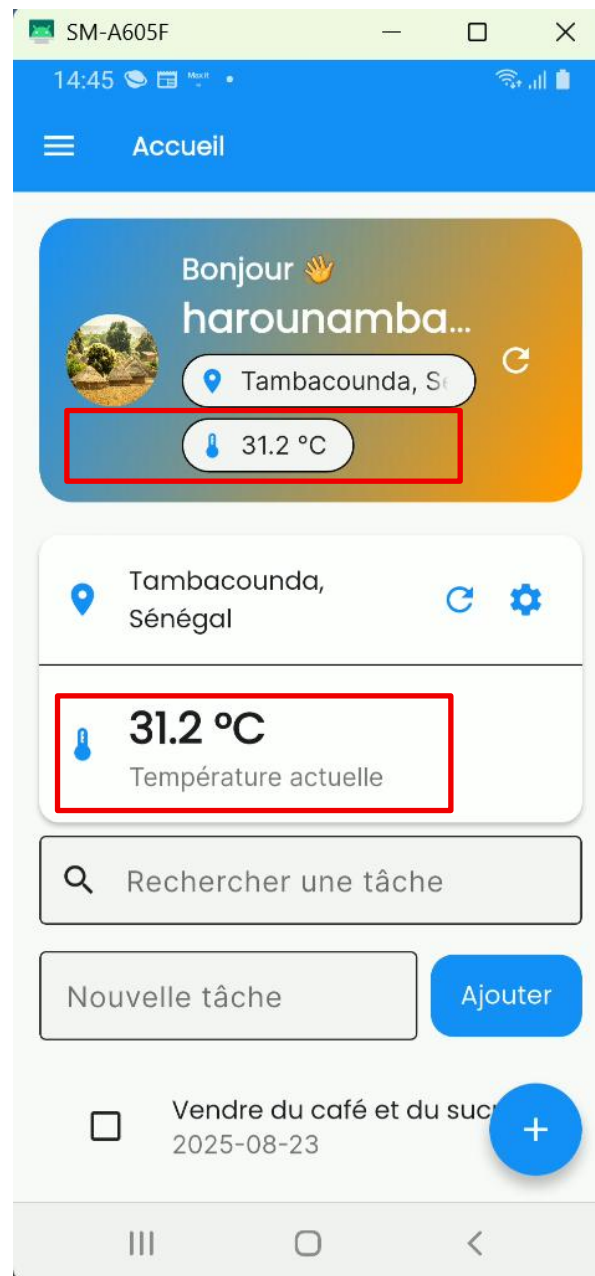


Figure 26: Météo

Conclusion

Ce projet a permis de mettre en œuvre une application Flutter complète intégrant un backend PHP/MySQL et des services tiers (géolocalisation, météo). L'approche Provider a permis une gestion efficace de l'état et une meilleure séparation des responsabilités. Les fonctionnalités répondent aux besoins d'organisation quotidienne tout en offrant un design moderne et ergonomique.