

**Session 2025**

**CATTAROSS I Evan**

**Numéro de candidat** : 02148874335

**Spécialité** : SIO - SLAM

**Numéro de réalisation** : 2

**Lien projet GitHub** : <https://github.com/EvanCattarossi/GSB-MOB.git>

**Lien portfolio** : <https://evan-cattarossi.fr/>



**Spécialité** : SLAM

**EPREUVE E6**

<b>BTS SERVICES INFORMATIQUES AUX ORGANISATIONS</b>		<b>SESSION 2025</b>
<b>Épreuve E5 - Conception et développement d'applications (option SLAM)</b>		
<b>ANNEXE 7-1-B : Fiche descriptive de réalisation professionnelle (recto)</b>		

<b>DESCRIPTION D'UNE RÉALISATION PROFESSIONNELLE</b>		<b>N° réalisation : 2</b>
<b>Nom, prénom : CATTAROSSE Evan</b>		<b>N° candidat : 02148874335</b>
<b>Épreuve ponctuelle</b> <input checked="" type="checkbox"/> <b>Contrôle en cours de formation</b> <input type="checkbox"/>		<b>Date : 13/06/2025</b>
<b>Organisation support de la réalisation professionnelle</b> Maestris BTS		
<b>Intitulé de la réalisation professionnelle</b> La réalisation s'inscrit dans le cadre d'un Projet Personnalisé Encadré, fondé sur un contexte fourni par le centre de formation. Il concerne une entreprise pharmaceutique fictive, nommée Laboratoire Galaxy Swiss Bourdin (GSB). L'objectif était de concevoir des outils destinés à améliorer le suivi de l'activité de visite médicale.		
<b>Période de réalisation :</b> Février .2025 - avril 2025 <b>Lieu :</b> Dans le cadre de la formation		
<b>Modalité :</b> <input checked="" type="checkbox"/> <b>Seul(e)</b> <input type="checkbox"/> <b>En équipe</b>		
<b>Compétences travaillées</b> <div style="margin-left: 20px;"> <b>X</b> Concevoir et développer une solution applicative  <b>X</b> Assurer la maintenance corrective ou évolutive d'une solution applicative  <b>X</b> Gérer les données         </div>		
<b>Conditions de réalisation<sup>1</sup> (ressources fournies, résultats attendus)</b>		
<b>Ressources fournies :</b> Documentation GSB : Applications Frais		
<b>Résultats attendus :</b> L'application doit Permettre aux visiteurs médicaux d'utiliser une <b>application Android native</b> pour : <ul style="list-style-type: none"> <li>Se connecter à leur espace utilisateur</li> <li>Saisir leurs frais mensuels (forfaitisés ou hors forfait)</li> <li>Suivre l'état de traitement de leurs fiches</li> <li>Permettre aux comptables de <b>valider ou refuser les fiches</b></li> <li>Utiliser les API REST pour interagir avec les données</li> </ul>		
<b>Description des ressources documentaires, matérielles et logicielles utilisées<sup>2</sup></b> - Descriptif de de l'entreprise GSB et les Objectifs - Environnement de Développement : Android Studio		
<b>Modalités d'accès aux productions<sup>3</sup> et à leur documentation<sup>4</sup> :</b> Lien GitHub: <a href="https://github.com/EvanCattarossi/GSB-MOB.git">https://github.com/EvanCattarossi/GSB-MOB.git</a>		

<sup>1</sup> En référence aux conditions de réalisation et ressources nécessaires du bloc « Conception et développement d'applications » prévues dans le référentiel de certification du BTS SIO.

<sup>2</sup> Les réalisations professionnelles sont élaborées dans un environnement technologique conforme à l'annexe II.E du référentiel du BTS SIO.

<sup>3</sup> Conformément au référentiel du BTS SIO « Dans tous les cas, les candidats doivent se munir des outils et ressources techniques nécessaires au déroulement de l'épreuve. Ils sont seuls responsables de la disponibilité et de la mise en œuvre de ces outils et ressources. La circulaire nationale d'organisation précise les conditions matérielles de déroulement des interrogations et les pénalités à appliquer aux candidats qui ne se seraient pas munis des éléments nécessaires au déroulement de l'épreuve. ». Les éléments peuvent être un identifiant, un mot de passe, une adresse réticulaire (URL) d'un espace de stockage et de la présentation de l'organisation du stockage. <sup>4</sup> Lien vers la documentation complète, précisant et décrivant, si cela n'a été fait au verso de la fiche, la réalisation professionnelle, par exemples service fourni par la réalisation, interfaces utilisateurs, description des classes ou de la base de données.

## Épreuve E5 - Conception et développement d'applications (option SLAM)

ANNEXE 7-1-B : Fiche descriptive de réalisation professionnelle  
(verso, éventuellement pages suivantes)**Descriptif de la réalisation professionnelle, y compris les productions réalisées et schémas explicatifs****Descriptif de la réalisation professionnelle :**

Au cours de la période de janvier à avril 2025, j'ai été responsable de la **conception** et du **développement d'une application mobile Android native** destinée à la gestion des frais professionnels pour les visiteurs médicaux de l'entreprise Galaxy Swiss Bourdin (GSB).

Cette application constitue la déclinaison mobile du projet GSB Frais, avec pour objectif de permettre une **utilisation terrain fluide et autonome**, sans dépendance à un poste fixe. Développée en **Java** avec une interface utilisateur en **XML** via **Android Studio**, elle consomme des **API REST** pour interagir avec le serveur web de l'application.

**Mise en place des API**

- Intégration d'un système de communication entre l'application mobile et le back-end via des **requêtes HTTP (GET, POST, PUT)**.
- Connexion aux API hébergées sur le serveur web (projet GSB-WEB), permettant :
  - l'authentification des utilisateurs (visiteurs/comptables),
  - la récupération des fiches de frais,
  - la soumission de nouvelles fiches et frais hors forfait,
  - la mise à jour de l'état des fiches côté comptable.

**Développement des fonctionnalités principales**

- Interface utilisateur responsive (en XML) avec navigation par rôles :
  - **Visiteur Médical :**
    - Connexion sécurisée
    - Saisie de frais forfaitisés (nuitées, repas, km)
    - Saisie de frais hors forfaits (date, libellé, montant)
    - Visualisation et suivi des fiches existantes
  - **Comptable :**
    - Consultation des fiches soumises par les visiteurs
    - Validation ou refus des fiches selon leur conformité

**Documentation technique**

- Commentaires complets dans le code source
- Dossier technique décrivant :
  - les architectures front (XML) et back (API)
  - les endpoints utilisés
  - la logique métier associée aux différents rôles
- Dossier utilisateur avec captures d'écran et mode opératoire

## Table des matières

1.FCHE DESCRIPTIVE ET RÉALISATION PROFESSIONNELLES .....	5
Description du Projet "GSB" .....	5
Principales Caractéristiques et Objectifs : .....	5
1. CONTEXTE DE L'APPLICATION.....	7
2.CAHIER DES CHARGES .....	9
Exigences fonctionnelles :.....	9
1. Se connecter .....	9
2. Renseigner une fiche de frais .....	9
3. Consulter mes fiches de frais.....	9
4. Valider une fiche de frais.....	10
Objectifs de cette structure .....	1
5. Travail, Gestion de Projet avec les Outils.....	2
Modalités de réalisation .....	2
6.Mode Opérateur Global de l'Application .....	3
Les visiteurs médicaux peuvent : .....	3
Les comptables disposent d'un accès spécifique dans l'application : .....	3
Gestion des utilisateurs par les administrateurs :.....	3
Fonctions avancées et sécurité : .....	4
Interface mobile : .....	4
7. Ergonomie, Design, Maquettage .....	4
◆ Conception de l'interface.....	4
2.Aspect Technique / Contraintes.....	6
Langages de programmation.....	6
Base de données .....	6
Outils & Frameworks.....	6
Architecture logique de l'application (Mobile).....	7
Étapes de saisie d'une fiche de frais (Visiteur médical).....	7
3.Concepteur de données – Application mobile.....	8
Objectifs de conception.....	8
Tables principales exploitées via API .....	8
4.Objectifs ,Délais , Attentes .....	9
Planning estimé – Application Mobile GSB Frais .....	10
Connexion à l'application.....	10
Renseigner une fiche de frais .....	12
Consulter et Modifier mes fiches .....	13
Consultation des fiches à traiter .....	14
Validation ou refus .....	15
Ajout d'un nouvel utilisateur.....	16
Visualiser les utilisateurs.....	17

Statistiques de Remboursement :.....	18
Axes d'amélioration de l'application GSB Frais.....	19
Fonctionnalités.....	19
Interface (UX/UI) .....	19
Sécurité.....	19
Technique.....	19

## **1.FCHE DESCRIPTIVE ET RÉALISATION PROFESSIONNELLES**

### **Description du Projet "GSB"**

L'objectif principal du projet "GSB" est le développement d'une application de gestion des frais de déplacement, de restauration, et d'hébergement généré par l'activité de visite médicale. Actuellement, la gestion des frais varie selon le laboratoire d'origine des visiteurs, et l'objectif est d'uniformiser ce processus.

### **Principales Caractéristiques et Objectifs :**

**Enregistrement des Frais :** L'application permet aux visiteurs médicaux d'enregistrer tous les frais engagés, qu'il s'agisse de déplacements, de repas, d'hébergement, ou d'autres activités annexes (événementiel, conférences, etc.).

**Suivi des Remboursements :** Un suivi précis des remboursements est intégré, indiquant l'état d'avancement des demandes de remboursement (enregistré, validé, remboursé).

**Uniformisation de la Gestion :** L'application vise à uniformiser la gestion des frais entre les différentes entités du laboratoire, assurant ainsi une cohérence et une transparence accrues.

**Authentification et Sécurité :** L'accès à l'application est restreint aux acteurs de l'entreprise, avec une authentification préalable. Tous les échanges avec le serveur Web sont cryptés pour garantir la sécurité des données.

Deux Modules distincts :

Module 1 : Enregistrement et suivi des frais par les visiteurs .

Module 2 : Enregistrement des opérations par les comptables.

Module 3 : Enregistrement et Suppression de comptes

**Respect des Normes de Développement :** Le développement de l'application doit suivre les normes et bonnes pratiques définies dans le document "ApplisWeb-NormesDevelpt" pour assurer la maintenabilité du code.

**Documentation Complète :** Une documentation détaillée doit être fournie, couvrant l'arborescence des pages, la description des éléments, classes, et bibliothèques utilisées, ainsi

que la liste des frameworks ou bibliothèques externes utilisés.


Gestion des Frais de Déplacement : La gestion des frais de déplacement, qui représente un poste de dépense significatif, est au cœur du projet. L'application doit assurer un suivi précis des dépenses.


Processus Informatisé : Automatisation du processus actuel, de la saisie des frais par les visiteurs jusqu'à la validation par le service comptable, la clôture et le suivi du remboursement



Voici l'interface principale de l'utilisateur, cette interface est celui qui permet à l'utilisateur de s'identifier pour pouvoir accéder aux fonctions suivant sont login mot de passe.

Se connecter

 Adresse e-mail

 Mot de passe

Se connecter

Identifiant de connexion :

Nom	Prénom	email	Mdp	rôle
Administrateur	Principal	admin@example.com	admin2024	Administrateur
Boda	Louis	louis.boda@example.com	visiteur123	Visiteur
CATTAROSSO	Evan	evan.cattarossi@example.com	comptable123	Comptable

Lors de la connexion, l'utilisateur saisit son e-mail et son mot de passe dans un formulaire. Ces identifiants sont envoyés à l'API pour vérification.

Ces comptes ont été créés pour :

- Tester les interfaces selon les rôles (visiteur, comptable, admin)
- Vérifier le fonctionnement du système d'authentification
- Simuler des cas d'usage lors des démonstrations

Les comptes peuvent être modifiés ou supprimés depuis l'interface d'administration web. L'application mobile utilise ces mêmes données, récupérées dynamiquement via les appels API.

## 1. CONTEXTE DE L'APPLICATION

Galaxy Swiss Bourdin (GSB) est une entreprise pharmaceutique résultant de la fusion entre le géant américain Galaxy et le conglomérat européen Swiss Bourdin. Cette fusion, survenue en 2009, a créé une entité leader dans le secteur pharmaceutique, avec un siège administratif à Paris et le siège social de la multinationale basé à Philadelphie, en Pennsylvanie, aux États-Unis. La France a été choisie comme un lieu stratégique pour améliorer le suivi de l'activité de visite médicale.

Secteur d'activité : L'industrie pharmaceutique, dans laquelle évolue GSB, est un secteur lucratif caractérisé par des mouvements fréquents de fusion-acquisition. Les regroupements de laboratoires ont donné naissance à des entités de grande envergure, mais des controverses récentes ont soulevé des questions autour de l'activité de visite médicale, laissant entendre des arrangements opaques entre l'industrie et les praticiens.

Réorganisation : Suite à la fusion, GSB a entrepris une réorganisation interne visant à optimiser les activités du groupe, réaliser des économies d'échelle, et intégrer le meilleur des deux laboratoires en termes de produits concurrents. Cette démarche a impliqué des restructurations, des licenciements et une modernisation de l'activité de visite médicale.

Système Informatique : Le système informatique de GSB est central dans son fonctionnement. Les fonctions administratives, la gestion des ressources humaines, la comptabilité, la direction, la communication, et d'autres services sont regroupés sur le site parisien. Une salle serveur, située au 6<sup>e</sup> étage, héberge les fonctions de base du réseau, les applications métier, et assure la sécurité des données, qui sont considérées comme stratégiques.

Voici le logo présent sur l'appli, ce logo reste simple et montre la signification des lettres GSB.







## 2.CAHIER DES CHARGES

### Exigences fonctionnelles :

#### 1. Se connecter

- **Acteur déclencheur** : Visiteur médical ou Comptable
- **Préconditions** : Aucune
- **Postconditions** : L'utilisateur est authentifié avec son rôle reconnu (Visiteur ou Comptable)

Scénario nominal :

- L'application affiche l'écran de connexion mobile (formulaire e-mail + mot de passe).
- L'utilisateur saisit ses identifiants puis appuie sur le bouton de connexion.
- L'application envoie une requête POST à l'API pour vérifier les identifiants.
- Si les données sont valides, l'utilisateur est redirigé vers son tableau de bord mobile correspondant à son rôle.

Exceptions :

- Si le login ou le mot de passe est incorrect, un message d'erreur s'affiche.
- Si l'utilisateur appuie sur le bouton de déconnexion, la session est terminée et il revient à l'écran de connexion.

#### 2. Renseigner une fiche de frais

- **Acteur déclencheur** : Visiteur médical
- **Préconditions** : L'utilisateur est connecté en tant que visiteur
- **Postconditions** : Les frais saisis sont enregistrés via l'API

Scénario nominal :

- L'utilisateur ouvre l'écran "Nouvelle fiche".
- L'application affiche les frais forfaitisés et hors forfait du mois en cours (si déjà existants).
- L'utilisateur peut modifier les valeurs forfaitisées (KM, REP, NUI) et appuyer sur "Valider".
- L'application envoie les données à l'API qui les enregistre.
- L'utilisateur peut aussi ajouter un frais hors forfait en saisissant :
  - une **date d'engagement**
  - un **libellé**
  - un **montant**
- Une fois les champs remplis, l'utilisateur valide, et l'API enregistre ce frais hors forfait

*Exceptions :*

:Si un champ est vide, un message s'affiche : "Le champ [nom du champ] doit être renseigné".

#### 3. Consulter mes fiches de frais

- **Acteur déclencheur** : Visiteur médical
- **Préconditions** : L'utilisateur est connecté
- **Postconditions** : Aucune modification de données

Scénario nominal :

- L'utilisateur appuie sur "Mes fiches".
  - L'application envoie une requête à l'API pour récupérer ses fiches.
  - Chaque fiche affiche :
    - le mois concerné
    - les frais forfaitisés avec quantités
    - le total de la fiche
    - l'état de traitement (Créée, Validée, Refusée, Remboursée)
- 

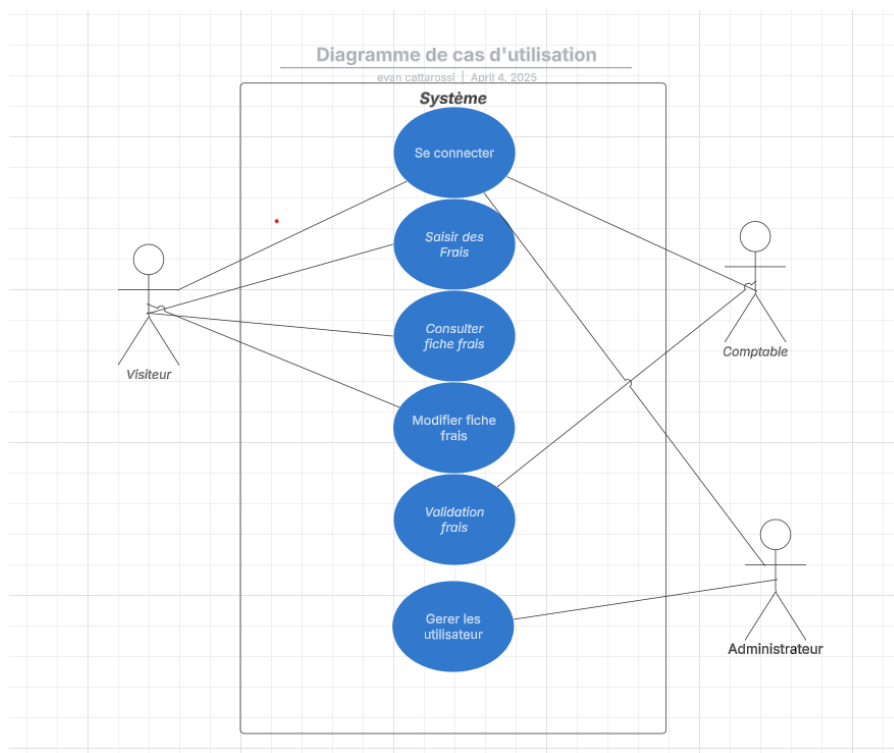
#### 4. Valider une fiche de frais

- **Acteur déclencheur** : Comptable
- **Préconditions** : L'utilisateur est connecté en tant que comptable
- **Postconditions** : La fiche passe à l'état "Validée" ou "Refusée"

Scénario nominal :

- Le comptable accède à la liste des fiches à traiter.
- Il sélectionne une fiche et visualise les détails.
- L'application propose deux actions : **Valider** ou **Refuser**.
- Le choix effectué est envoyé à l'API.
- L'API met à jour l'état de la fiche (et la date de dernière modification).

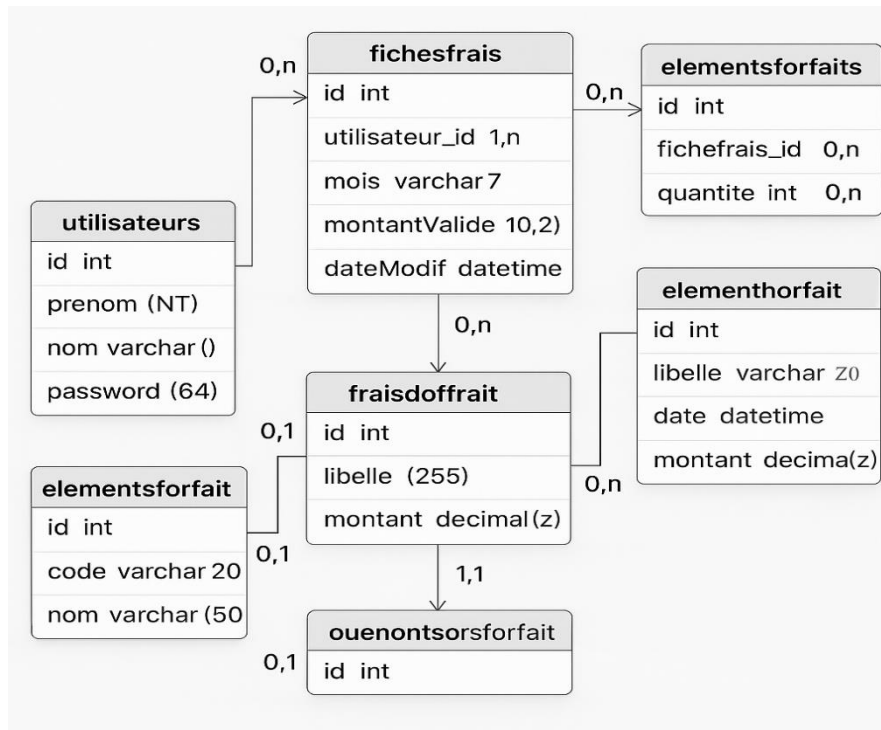
#### Diagramme Cas utilisation de l'appli :



#### 4. Conception des Données et/ou MCD (Modèle Conceptuel de Données)

L'application mobile GSB Frais utilise les **mêmes structures de données** que l'application web, accessibles via des **API REST**. Les entités, leurs relations et les règles de gestion sont donc identiques.

Le schéma MCD ci-dessous est **partagé entre les deux versions** (web et mobile) afin de garantir la **cohérence des données** et de faciliter la maintenance future.



Le MCD représente l'organisation logique des données manipulées par l'application. Il a été conçu en début de projet afin de poser les bases de la **structure de la base de données relationnelle** utilisée en SQL.

#### **Objectifs de cette structure**

- Permettre une **communication fluide entre mobile et serveur** via API.
- Garantir une cohérence parfaite avec les données utilisées par la version web.
- Faciliter les traitements côté client (application Android) par des réponses bien structurées (JSON).
- Séparer les types de frais pour offrir une meilleure **expérience utilisateur mobile**.

## **5.Travail, Gestion de Projet avec les Outils**

### **Modalités de réalisation**

Le développement de l'application mobile **GSB Frais** a été mené **en autonomie**, dans le cadre d'un Projet Personnel Encadré (PPE) au sein du BTS SIO option SLAM.

Le projet s'inscrit en parallèle de la version web. Il a nécessité l'adaptation des interactions serveur via API REST, le développement d'une interface mobile sous Android Studio, et l'intégration des fonctionnalités spécifiques au terminal mobile.

Pour mener à bien la réalisation de ce projet mobile, j'ai utilisé un **ordinateur portable personnel équipé d'Android Studio** comme environnement de développement (IDE). Cet outil complet m'a permis de développer efficacement l'application en **Java** pour la logique métier et en **XML** pour la conception des interfaces utilisateurs.

Par ailleurs, l'utilisation de **GitHub** m'a permis d'assurer un suivi régulier de l'évolution du projet, avec une gestion des versions facilitée et une **sauvegarde continue** du code source. Cela garantit la **sécurité, la traçabilité** et la possibilité de revenir à une version stable en cas de besoin.

Tous les fichiers du projet mobile « GSB Frais » ont été organisés dans le dépôt [\*\*GSB-MOB\*\*](#). Le code est structuré par fonctionnalités : connexion, affichage des fiches, saisie des frais, communication avec les **API REST**, etc.

Cette organisation claire facilite la lecture du projet, les modifications ultérieures, et permet une bonne maintenance du code. Elle est également essentielle pour suivre l'historique de développement, identifier les changements et assurer la continuité du projet.

## 6.Mode Opérateur Global de l'Application

L'application **GSB Frais Mobile** a été conçue pour répondre aux besoins spécifiques de gestion de frais professionnels **en mobilité**. Elle prend en compte les différents profils utilisateurs (visiteur, comptable), tout en garantissant la **sécurité**, l'**ergonomie** et la **portabilité** de l'application sur smartphones Android.

Chaque utilisateur doit pouvoir se connecter via son mobile en utilisant un identifiant unique (e-mail) et un mot de passe sécurisé.

- Le système gère une **session locale temporaire** pour filtrer les accès selon le rôle de l'utilisateur.
- Les échanges de données sont sécurisés grâce à des **requêtes HTTPS** vers les API hébergées sur le serveur distant.

Les visiteurs médicaux peuvent :

- Créer une fiche de frais mensuelle via un formulaire natif dans l'application.
- Renseigner des **frais forfaitisés** : kilomètres (KM), nuitées (NUI), repas (REP), avec des montants fixes.
- Ajouter des **frais hors forfait**, tels que des taxis ou des stationnements, en précisant :
  - la **date d'engagement**
  - le **libellé du frais**
  - le **montant**

*(Ajout de justificatifs en image prévu pour une future évolution.)*

Ils peuvent également :

- Consulter l'historique de leurs fiches mois par mois.
- Vérifier l'état de traitement de chaque fiche :
  - **CR** : Créée
  - **VA** : Validée
  - **RB** : Remboursée
  - **RF** : Refusée

Les comptables disposent d'un accès spécifique dans l'application :

- Ils peuvent consulter toutes les fiches envoyées par les visiteurs.
- Pour chaque fiche, ils peuvent :
  - Voir les détails des frais saisis
  - **Valider** ou **refuser** la fiche directement depuis leur interface
  - Attribuer un **statut** aux frais hors forfait : Accepté, Refusé, En attente

Gestion des utilisateurs par les administrateurs :

La gestion des comptes (ajout, modification, suppression) reste centralisée **dans l'interface web d'administration**, non incluse dans la version mobile.

Les administrateurs peuvent :

- Gérer les rôles (visiteur, comptable, administrateur)

- Visualiser les statistiques de remboursement
- Accéder à l'ensemble des activités via un tableau de bord web

Fonctions avancées et sécurité :

- L'application est prête à intégrer des **statistiques** (via les API du back-end web)
- Authentification robuste : mots de passe hachés via **bcrypt**
- Requêtes API protégées et échanges **chiffrés**
- Architecture mobile pensée pour l'avenir :
  - Upload de justificatifs
  - Double authentification
  - Notifications push

Interface mobile :

- Simple, intuitive, fluide
- Navigation par rôle (écrans dédiés)
- Développée en **Java/XML** pour Android
- Optimisée pour une utilisation tactile, avec retours visuels et messages d'erreur clairs

## **7. Ergonomie, Design, Maquettage**

L'ergonomie et le design de l'application mobile **GSB Frais** ont été pensés pour offrir une **expérience utilisateur fluide, claire et intuitive**, adaptée aux contraintes de la mobilité (petit écran, saisie rapide, accès tactile).

L'objectif était de **faciliter la navigation** pour les deux types d'utilisateurs principaux (visiteur médical et comptable), tout en assurant la **lisibilité des informations** et une **saisie simplifiée des données**.

---

### Conception de l'interface

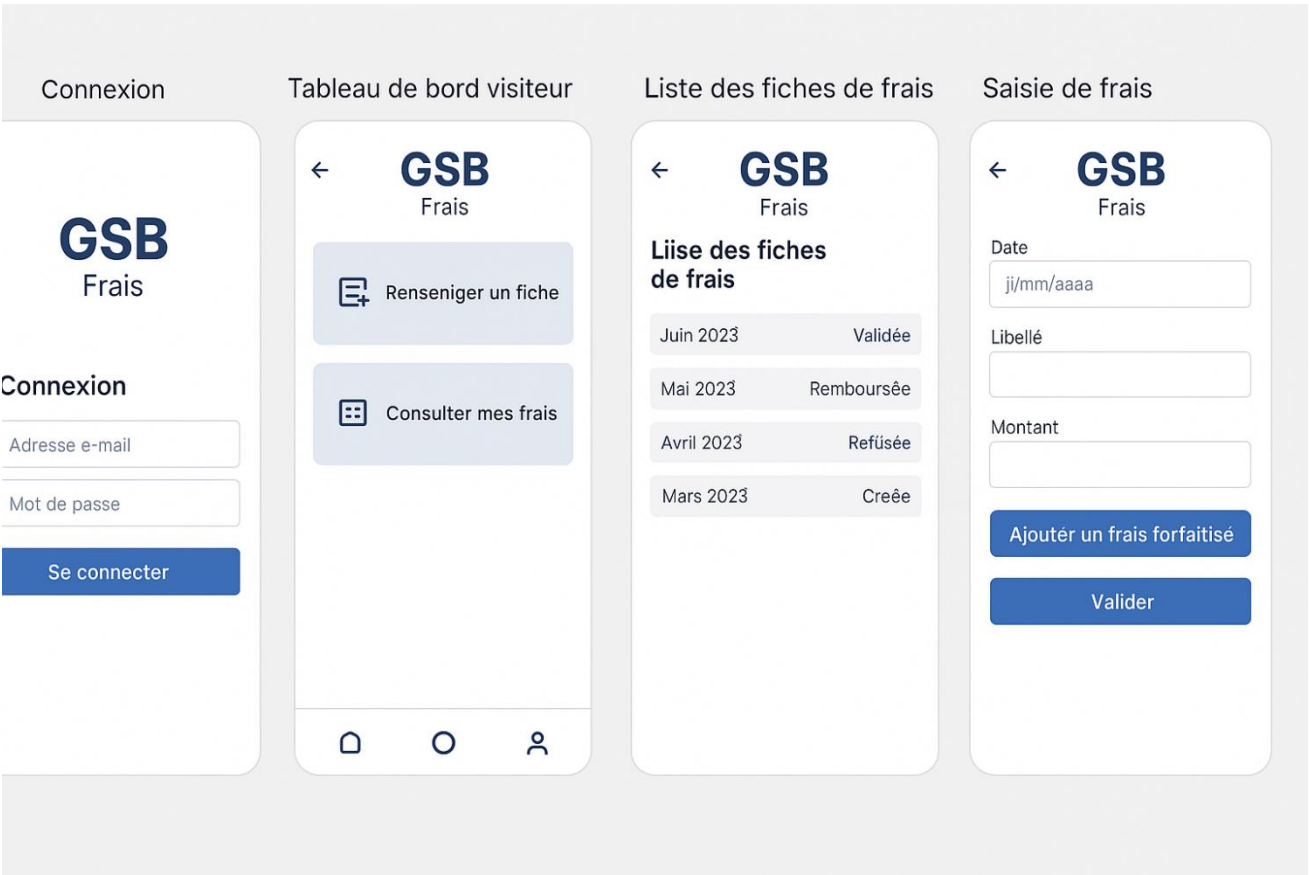
L'interface a été conçue en **XML sous Android Studio**, avec une architecture en **écrans successifs** :

- **Écran de connexion** avec champs e-mail / mot de passe
- **Tableau de bord** affiché après authentification, personnalisé selon le rôle
- **Formulaires de saisie de frais** clairs et accessibles
- **Écrans de consultation** listant les fiches par mois

- **Actions de validation / refus** pour le rôle comptable

Choix graphiques

Élément	Détail
Palette de couleurs	Tons sobres et professionnels (bleu, blanc, gris) pour rester lisible en toutes conditions
Police	Utilisation des polices système Android (Roboto), adaptées à la lisibilité mobile
Boutons	Grands, arrondis, avec retour visuel au clic (Ripple effect)
Messages d'erreur	Visibles sous les champs en rouge, avec un texte explicite
Navigation	Par menu contextuel ou retour avec flèches Android



- Écran de connexion
- Dashboard Visiteur
- Formulaire de saisie de frais
- Liste des fiches
- Interface Comptable pour validation

Accessibilité et adaptabilité

- Application **responsive**, testée sur plusieurs résolutions Android
- Textes et boutons **agrandis** pour être accessibles même sur petit écran
- Affichage adapté en **portrait uniquement** pour éviter les erreurs d'interface

## 2.Aspect Technique / Contraintes

### Technologies utilisées et architecture de l'application

L'application **GSB Frais Mobile** repose sur une architecture client-serveur, où l'application mobile native (développée en Java sous Android Studio) **consomme les données et les services** fournis par les **API PHP** déjà développées dans le projet web.

### Langages de programmation

- **Java** : Langage principal utilisé pour le développement des fonctionnalités de l'application mobile Android (logique métier, appels API, gestion des écrans).
- **XML** : Utilisé pour la conception des interfaces utilisateurs (formulaires, listes, boutons, navigation).
- **JSON** : Format d'échange de données entre l'application mobile et les API (en lecture/écriture).
- **SQL** (*côté serveur*) : Utilisé dans le back-end PHP pour manipuler la base de données MySQL, avec lequel l'application interagit via API

### Base de données

- **MySQL** : Système de gestion de base de données relationnelle utilisé côté serveur.
- **Connexion via API REST (PHP + PDO)** : Toutes les données (authentification, fiches, frais) sont récupérées ou envoyées via des **endpoints sécurisés**, sans accès direct à la base depuis l'application mobile.
- Les requêtes comme **GET, POST, PUT, DELETE** permettent d'effectuer les opérations classiques : création, modification, consultation et suppression de données.

### Outils & Frameworks

- **Android Studio** : Environnement de développement intégré (IDE) utilisé pour coder l'application mobile Android.
- **GitHub** : Outil de gestion de version et de collaboration, permettant de suivre l'évolution du code source.
- **Trello** : Outil de gestion de projet (organisation des tâches sous forme de tableau Kanban).
- **Postman** : Utilisé pour tester les API et simuler les requêtes en développement.
- **Emulateur Android** : Permet de tester l'application sur différents modèles d'appareils virtuels (smartphones, tablettes).
- **Volley / HttpURLConnection** : Bibliothèques utilisées pour effectuer les appels réseau (requêtes HTTP vers l'API).



## Architecture logique de l'application (Mobile)

L'application mobile **GSB Frais** suit une architecture structurée autour des rôles utilisateurs, avec une **navigation orientée selon le profil connecté**. La logique est organisée en **étapes claires**, illustrées par le diagramme de flux.

### Étapes de saisie d'une fiche de frais (Visiteur médical)

#### Étape 1 : Connexion

- L'utilisateur ouvre l'application et accède à l'écran de connexion.

Il saisit son e-mail et son mot de passe.

•

#### Étape 2 : Authentification via API

- L'application envoie une requête `POST` à l'API d'authentification.
- Si les identifiants sont valides, le rôle de l'utilisateur est récupéré.

#### Étape 3 : Redirection

- L'utilisateur est redirigé vers le **dashboard Visiteur**.

#### Étape 4 : Saisie d'une fiche

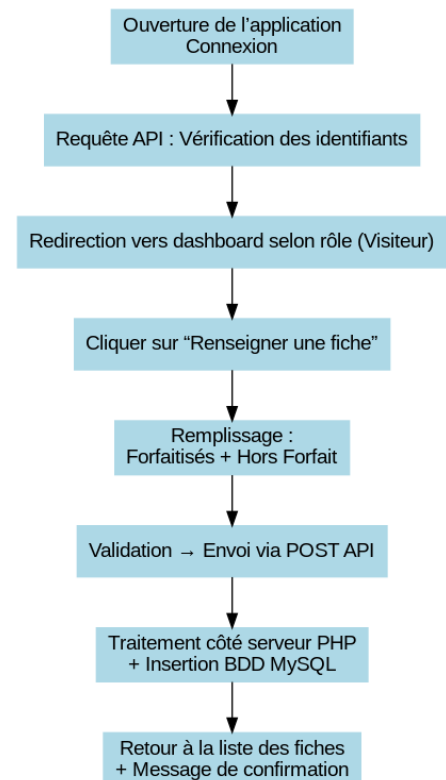
- Il clique sur **"Renseigner une fiche"**.
- Il remplit :
  - Les **frais forfaitisés** (REP, KM, NUI)
  - Les **frais hors forfait** (libellé, date, montant)

#### Étape 5 : Envoi des données

- Les données saisies sont envoyées via une requête API (POST).
- Le serveur PHP traite les données et les insère en base (MySQL).

#### Étape 6 : Confirmation

- Un message de confirmation s'affiche.
- L'utilisateur est redirigé vers la page de **consultation de ses fiches**



### 3. Concepteur de données – Application mobile

La base de données reste centralisée sur le serveur (MySQL), et les données sont accessibles uniquement via les **API sécurisées**. La structure reste la même que pour l'application web, mais **la logique métier est pilotée depuis le mobile**.

#### Objectifs de conception

- Structuration claire des données selon les rôles (Visiteur, Comptable, Admin)
- Gestion des frais par type (forfaitisés / hors forfait)
- Suivi des fiches et de leurs statuts

#### Tables principales exploitées via API

Table	Rôle
utilisateurs	Authentification, rôles (visiteur, comptable, admin), données de connexion
fichesfrais	Fiches mensuelles avec état et montant
etats	États possibles des fiches : Créée (CR), Validée (VA), Remboursée (RB), Refusée (RF)
elementsforfaitises	Frais fixes saisis par le visiteur
elementshorsforfait	Frais exceptionnels ajoutés manuellement
fraisforfait	Définition des frais standards (KM, REP, NUI)

Comme pour l'application web, l'application **GSB Frais Mobile** s'appuie sur la **même base de données distante**, initialisée à l'aide d'un script SQL. Ce script crée les principales tables nécessaires au fonctionnement de l'application mobile.

Chaque table respecte les **normes relationnelles**, avec l'utilisation de **clés primaires et étrangères**, afin d'assurer l'**intégrité des données** lors des échanges via les **API REST**.

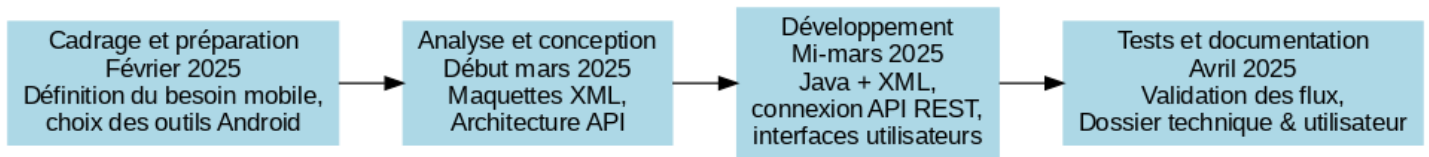
```
-- Table des utilisateurs
CREATE TABLE IF NOT EXISTS Utilisateurs (
  id INT AUTO_INCREMENT PRIMARY KEY,
  nom VARCHAR(100) NOT NULL,
  prenom VARCHAR(100) NOT NULL,
  email VARCHAR(100) UNIQUE NOT NULL,
  motDePasse VARCHAR(255) NOT NULL,
  role ENUM('visiteur', 'comptable', 'administrateur') NOT NULL,
  dateInscription DATETIME DEFAULT CURRENT_TIMESTAMP
);

-- Table des états des fiches de frais
CREATE TABLE IF NOT EXISTS Etats (
  id VARCHAR(2) PRIMARY KEY,
  libelle VARCHAR(50) NOT NULL
);

-- Table des fiches de frais
CREATE TABLE IF NOT EXISTS FichesFrais (
  id INT AUTO_INCREMENT PRIMARY KEY,
  utilisateurId INT NOT NULL,
  mois VARCHAR(6) NOT NULL, -- Format AAAA-MM
  montantValide DECIMAL(10, 2) DEFAULT 0.00,
  dateModif DATETIME DEFAULT CURRENT_TIMESTAMP,
  idEtat VARCHAR(2) DEFAULT 'CR', -- Par défaut : Créée
  FOREIGN KEY (utilisateurId) REFERENCES Utilisateurs(id),
  FOREIGN KEY (idEtat) REFERENCES Etats(id)
);

-- Table des éléments forfaitisés
CREATE TABLE IF NOT EXISTS ElementsForfaitises (
  id INT AUTO_INCREMENT PRIMARY KEY,
  ficheFraisId INT NOT NULL,
  typeForfait VARCHAR(50) NOT NULL, -- Exemple : "Forfait étape", "Frais kilométriques"
  quantite INT NOT NULL,
  montant DECIMAL(10, 2) NOT NULL,
  FOREIGN KEY (ficheFraisId) REFERENCES FichesFrais(id)
);
```

## 4.Objectifs ,Délais , Attentes



L'image représente une **frise chronologique horizontale** illustrant les **quatre grandes phases** de mon projet mobile, réparties de **février à avril 2025**. Elle repose sur une **flèche grise orientée de gauche à droite**, symbolisant la progression du temps et de l'avancement du développement.

Chaque phase est marquée par un **point circulaire bleu** accompagné d'un court descriptif :

### ◆ **Février 2025 – Phase de cadrage et préparation**

- Définition des objectifs spécifiques de la version mobile de l'application GSB Frais
- Choix de la technologie mobile (Java, XML, Android Studio)
- Création du dépôt GitHub dédié au projet GSB-MOB
- Mise en place de l'environnement de développement et outils (Trello, Postman)

### ◆ **Début mars 2025 – Phase d'analyse et conception**

- Rédaction du cahier des charges spécifique à la version mobile
- Conception des **maquettes des écrans Android** (connexion, tableau de bord, fiche de frais)
- Définition de l'architecture de communication entre l'application mobile et les **API REST** du serveur web

### ◆ **Mi-mars 2025 – Phase de développement**

- Développement des interfaces avec XML et des fonctionnalités en Java
- Mise en place de la navigation selon les rôles (visiteur/comptable)
- Connexion de l'application aux **services API existants** (authentification, saisie, consultation)
- Gestion des erreurs, validations et retours utilisateur

### ◆ **Avril 2025 – Phase de tests et finalisation**

- Tests fonctionnels des principales fonctionnalités (connexion, envoi/consultation de fiches)
- Correction des anomalies et amélioration de l'expérience utilisateur
- Rédaction de la **documentation technique et utilisateur**
- Intégration dans le dossier E6 pour soutenance

## Planning estimé – Application Mobile GSB Frais

Phase	Durée estimée	Livrables / Résultats
Préparation & outils	6 h	Cahier des charges, environnement Android Studio, dépôt GitHub
Analyse & maquettage	6 h	MCD adapté mobile, maquettes écrans XML
Dev base + connexions	8 h	Authentification, navigation, appels API REST
Fonctionnalités principales	14 h	Saisie et consultation des fiches de frais, interface comptable
Tests et documentation	6 h	Dossier utilisateur et technique, corrections finales
<b>Total</b>	<b>40 h</b>	Application mobile fonctionnelle

## 5. Mode Opérateur de l'Application Web

Cette section présente les principales **étapes d'utilisation de l'application**, illustrées par des **captures d'écran réelles** issues du projet. Elle constitue un **guide utilisateur visuel et pratique**, utile aussi bien pour la démonstration que pour l'autoformation.

### Connexion à l'application

The screenshot shows the login interface for the GSB application. At the top is the GSB logo with the text 'Galaxy Swiss Bourdin' below it. The title 'Se connecter' is centered. Below the title are two input fields: 'Adresse e-mail' (with an envelope icon) and 'Mot de passe' (with a lock icon). At the bottom is a blue button labeled 'Se connecter'. Three blue arrows point from text boxes on the right to these elements: the first arrow points to the email field, the second to the password field, and the third to the 'Se connecter' button.

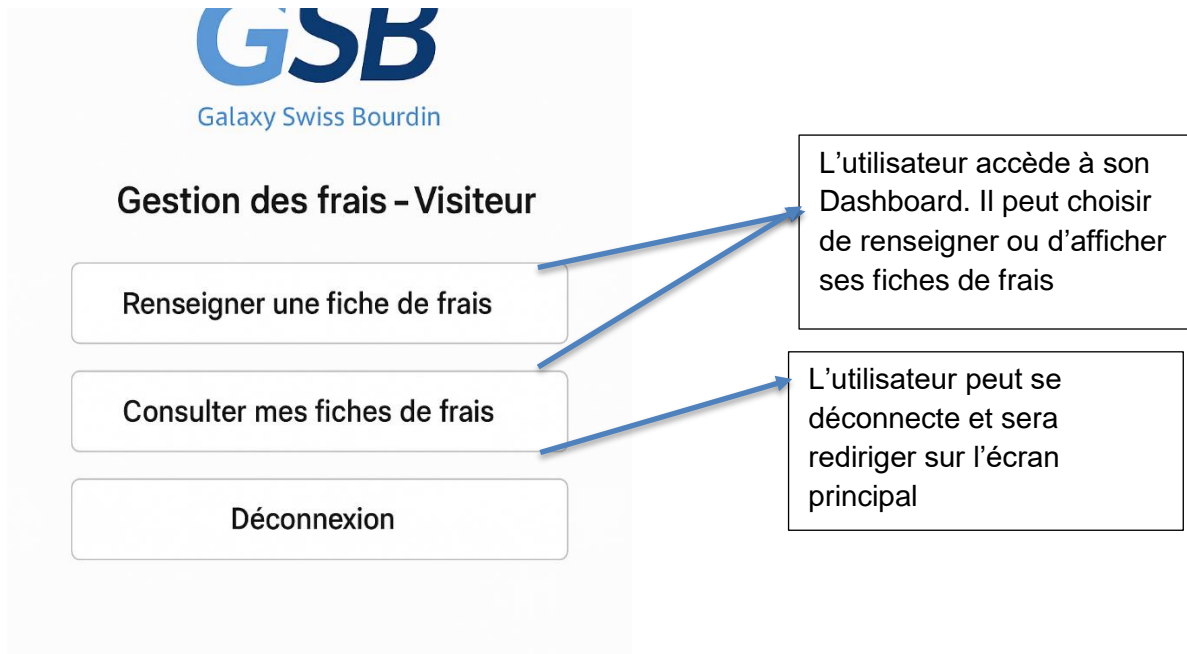
L'utilisateur se connecte avec ses identifiants (visiteur, comptable, admin)

L'utilisateur clique sur se connecter pour pouvoir avoir accès à son dashboard(accueil)

## Accueil du visiteur médical

Comme dis précédemment une fois connecté, le visiteur accède à son **dashboard personnalisé**. Il peut :

- Renseigner une nouvelle fiche de frais.
- Consulter l'historique de ses fiches de Frais



Dans un premier temps, l'utilisateur vas saisir sa fiche de frais avec les éléments forfaitisés et hors forfait , il vas donc cliquer sur **Renseigner la fiche de frais** .

Dans un premier temps l'utilisateur vas se connecter avec ses identifiants visiteur médical

Il vas donc être rediriger sur son Dashboard visiteur ou il vas pouvoir consulter et renseigner ses fiches de frais

## Renseigner une fiche de frais

Le visiteur complète sa fiche de frais mensuelle :

- **Frais forfaitisés** : nombre de kilomètres, nuitées, repas.
- **Frais hors forfait** : avec libellé, montant et date.
- Il peut ajouter plusieurs lignes et soumettre

### Renseigner Fiche de Frais

Choisir la date de fiche de frais

Choisir la date de la  
fiche de frais

#### Frais Forfaitisés

Kilométrage (KM)
Nuitée Hôtel (80,00 € / unité)
Repas Restaurant (25,00 € / unité)

Rentrer les renseignements  
Frais forfaitisés selon votre  
mois

#### Frais Hors Forfait

Date
Libellé
Montant (€)

L'utilisateur rentre la  
date, les  
informations hors  
forfait du mois en  
indiquant le libellé  
(ex : conférence)

Ajouter un frais hors forfait

Soumettre

On peut ajouter  
plusieurs frais hors  
forfaits

L'utilisateur soumet  
sa fiche de frais

# Consulter et Modifier mes fiches

Une interface claire permet d'afficher l'ensemble des fiches enregistrées avec :

- Le mois concerné
- Le montant validé
- L'état de la fiche (Créée, Remboursée, Refusée)
- Des liens pour **consulter ou modifier** chaque fiche.

Galaxy Swiss Bourdin

Mes Fiches de Frais

Mois	Montant Validé	État
02/2024	1.130,00 €	<a href="#">Consulter &gt;</a>
01/2024	944,00 €	<a href="#">Consulter &gt;</a>
12/2023	0,00 €	<a href="#">Consulter &gt;</a>
Refusée	Refusée	<a href="#">Consulter &gt;</a>

L'utilisateur clique sur consulter et aura accès à sa fiche de frais rempli

GSB

Consultation Fiche Frais

Fiche de Frais du 2025-02

Frais Forfaitisés

Type de Frais	Quantité	Montant Unitaire	Total
KM	500	0,46 €	230,0 €
Nuitée	7	80,00 €	560,0 €
REP	9	25,00 €	225,0 €

Frais Hors Forfait

Date	Libellé	Montant
2025-02-03	Conférence	110,00 €
2025-02-08	déjeuner pro	24,00 €

L'utilisateur aura le visuel sur ce qu'il a

L'utilisateur peut aussi modifier sa fiche

## Modifier Fiche de Frais

### Frais Forfaitisés

Type de Frais	Quantité
KM	500
NL	7
REP	8

### Frais Hors Forfait

01/02/2024	Montant
Conférence	110,00
déjeuner pro	25,80

Enregistrer les modifications

L'utilisateur peut modifier une ou plusieurs données sur sa fiche de frais

# Partie Comptable

## Consultation des fiches à traiter

Le comptable visualise toutes les fiches soumises par les visiteurs avec :

- Le nom de l'utilisateur
- Le mois, le montant et l'état de la fiche

Le comptable voit le nom de l'utilisateur qui a rempli la fiche

### Fiches de frais par visiteur

Nom	Mois	Montant Validé	État
Lucas Durta	Février 2023	400,00 €	Rem-boursé
Lucas Durta	Mars 2023	650,00 €	Créée
Lucas Durta	Avril 2023	210,00 €	Refusé

Le comptable voit le mois de cette fiche de frais

Le comptable voit le montant de la fiche

Le comptable voit l'état de la fiche



## Validation ou refus

Depuis cette interface, le comptable peut :

- Cliquer sur "**Rembourser**" pour valider la fiche
- Cliquer sur "**Refuser**" en cas de non-conformité

Nom	Mois	Montant	Action	
Lucas Durta	Février 2023	400,00 €	Rembourser	Refuser
Lucas Durta	Mars 2023	650,00 €	Rembourser	Refuser
Lucas Durta	Avril 2023	210,00 €	Rembourser	Refuser

Le comptable  
peut rembourser  
une fiche de frais

Le comptable peut  
refuser le  
remboursement

# Partie Administrateur

## Ajout d'un nouvel utilisateur

L'administrateur accède à un formulaire simple pour :

- Saisir les informations (nom, prénom, email, mot de passe)
- Choisir un **rôle** (visiteur, comptable ou administrateur)

The diagram illustrates the 'Ajouter un utilisateur' (Add user) form. At the top, the GSB logo and the word 'Frais' are displayed. The form title 'Ajouter un utilisateur' is centered. The form contains five input fields: 'Nom', 'Prénom', 'E-mail', 'Mot de passe', and 'Rôle' (a dropdown menu). A blue 'Ajouter' button is at the bottom. Annotations with arrows point to each field, explaining the data entered by the administrator:

- L'admin peut ajouter un nouvel utilisateur** (The admin can add a new user) points to the form title.
- L'admin saisi le nom de l'utilisateur** (The admin entered the user's name) points to the 'Nom' field.
- L'admin saisi le prénom du nouvel** (The admin entered the first name of the new user) points to the 'Prénom' field.
- L'admin saisi email du nouvel utilisateur** (The admin entered the email of the new user) points to the 'E-mail' field.
- L'admin saisi le mot de passe du nouvel utilisateur** (The admin entered the password of the new user) points to the 'Mot de passe' field.
- L'admin saisi le rôle du nouvel utilisateur** (The admin entered the role of the new user) points to the 'Rôle' dropdown menu.

## Visualiser les utilisateurs

L'administrateur peut visualiser l'entièreté des utilisateurs de l'application Mobile :

The diagram illustrates that an administrator can view all information about users in the application. A table lists three users, and callout boxes with arrows point to each column, stating that the admin can see that specific data.

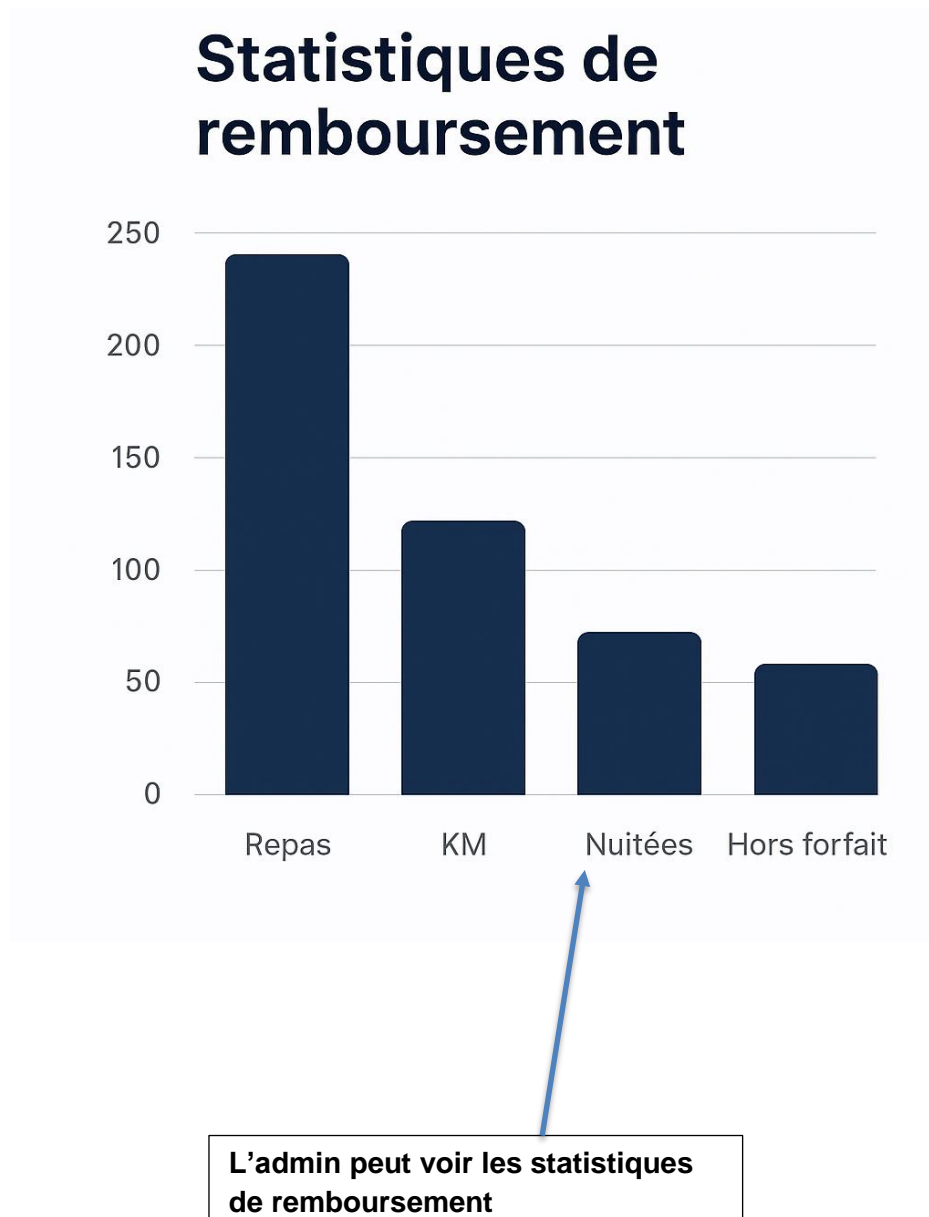
Nom	Prénom	E-mail	Rôle
Durta	Lucas	lucas.durta	Visiteur
Sophie	Sophie	sophie.martin@example.com	Comptable
Leroy	Paul	paul.leroy@example.com	Administrateur

Callouts:

- L'admin voit le nom de l'utilisateur (points to Nom)
- L'admin voit le prénom de l'utilisateur (points to Prénom)
- L'admin voit l'email de l'utilisateur (points to E-mail)
- L'admin voit le rôle de l'utilisateur (points to Rôle)

## Statistiques de Remboursement :

L'administrateur a accès à un diagrammes qui lui permet de pouvoirs voir les statistiques de remboursement de frais et de se faire un avis sur ce qui est le plus et le moins rembourser :



# **Axes d'amélioration de l'application GSB Frais**

## **Fonctionnalités**

- Ajout de justificatifs (photos, PDF)
- Notifications push pour les statuts des fiches
- Historique d'actions détaillé

## **Interface (UX/UI)**

- Optimisation de l'ergonomie mobile
- Design plus moderne et cohérent avec la version web
- Prise en charge du mode sombre

## **Sécurité**

- Authentification à deux facteurs (2FA)
- Déconnexion automatique
- Chiffrement renforcé des échanges API

## **Technique**

- Passage à une architecture plus modulaire (ex. MVVM)
- Documentation du code
- Évolutivité vers iOS avec Flutter ou React Native