**Etape 1 : Dégager l’idée du projet et Identifier les objectifs spécifiques**

**🎯 1. Présentation du projet**

**Nom du projet** : e-Tontine  
**Type** : Application web de gestion de tontines

**Description** :  
e-Tontine est une application web qui permet à un groupe d'utilisateurs de créer et gérer des tontines en ligne. Chaque utilisateur peut rejoindre une tontine, cotiser régulièrement, et recevoir une somme à un moment donné selon un ordre défini (rotation). L'application facilite le suivi des cotisations, la planification des bénéficiaires et l'envoi de rappels.

**🎯 2. Objectifs spécifiques du projet**

Voici des objectifs concrets que vous pouvez atteindre :

1. ✅ Permettre aux utilisateurs de s’inscrire et se connecter de manière sécurisée.
2. ✅ Créer et rejoindre des tontines existantes.
3. ✅ Gérer la rotation des bénéficiaires de manière automatique.
4. ✅ Envoyer des notifications de rappel aux membres.
5. ✅ Afficher un tableau de bord personnalisé.
6. ✅ Enregistrer les cotisations avec l’historique.
7. ✅ Gérer les rôles (créateur, membre).
8. ✅ Permettre la récupération de mot de passe par email.
9. ✅ Générer des rapports simples de participation et cotisation.
10. ✅ Assurer la sécurité et la validation des données saisies.

**Etape 2 : Élaborer l’organigramme des tâches**

| **N°** | **Phase** | Sous-tâches principales |
| --- | --- | --- |
| **1** | **Analyse & Planification** | 1.1 Définir les besoins utilisateurs  1.2 Identifier les fonctionnalités  1.3 Objectifs techniques1.4 Cahier des charges |
| **2** | **Conception** | 2.1 Diagrammes UML (cas d'utilisation, classes, séquence)  2.2 Prototype Figma  2.3 Validation du parcours utilisateur |
| **3** | **Développement Back-End** | 3.1 Modélisation base de données  3.2 Authentification (Inscription, Login, MDP oublié)  3.3 Gestion des tontines  3.4 Cotisations  3.5 Notifications |
| **4** | **Développement Front-End** | 4.1 Interface utilisateur (tableau de bord, mes tontines, historique)  4.2 Responsive design  4.3 Interface admin (si besoin) |
| **5** | **Tests & Sécurité** | 5.1 Validation des entrées  5.2 Tests fonctionnels  5.3 Sécurité (mot de passe, injection)  5.4 Sécurisation des requêtes |
| **6** | **Déploiement** | 6.1 Hébergement local/serveur  6.2 Configuration base de données  6.3 Tests finaux |
| **7** | **Collaboration & Suivi** | 7.1 Gestion de projet (Trello/Jira)  7.2 Versionning (Git/GitHub)  7.3 Partage fichiers (Google Drive)  7.4 Répartition des rôles (RACI) |

**🗓️ Étape 3 : Estimation de la durée – Diagramme de Gantt & Réseau de Pert**

🧩 **1. Estimation des durées (en jours ouvrables)**

| **Tâche ID** | **Tâche** | **Durée estimée** |
| --- | --- | --- |
| T1 | Analyse des besoins | 2 j |
| T2 | Définition des objectifs | 1 j |
| T3 | Conception UML | 2 j |
| T4 | Prototype Figma | 2 j |
| T5 | Création base de données | 2 j |
| T6 | Développement Authentification | 2 j |
| T7 | Gestion des tontines | 2 j |
| T8 | Gestion cotisations | 2 j |
| T9 | Notifications & alertes | 1 j |
| T10 | Front-end (dashboard + pages) | 3 j |
| T11 | Tests et sécurité | 2 j |
| T12 | Déploiement | 1 j |
| T13 | Organisation & suivi (Trello, Git) | en parallèle |

📊 **2. Diagramme de Gantt simplifié**

Jour → 1 2 3 4 5 6 7 8 9 10 11 12

-------------------------------------------------------

T1 ███

T2 █

T3 ██

**Légende :**

**T1 commence au jour 1 et dure 3 jours.**

**T3 commence au jour 5 et dure 2 jours.**

**T13 s’étale sur toute la durée du projet en parallèle (par exemple : gestion de projet, communication, etc.).**

T4 ██

T5 ██

T6 ██

T7 ██

T8 ██

T9 █

T10 ███

T11 ██

T12 █

T13 ███████████████████████████ (parallèle)

**3. Réseau de Pert (Résumé)**

Le diagramme de Pert montre les **dépendances** :

* T1 → T2 → T3 → T4 → T10 → T11 → T12
* T3 → T5 → T6, T7, T8 → T9 → T10 → T11 → T12
* T13 est parallèle et continue tout au long du projet

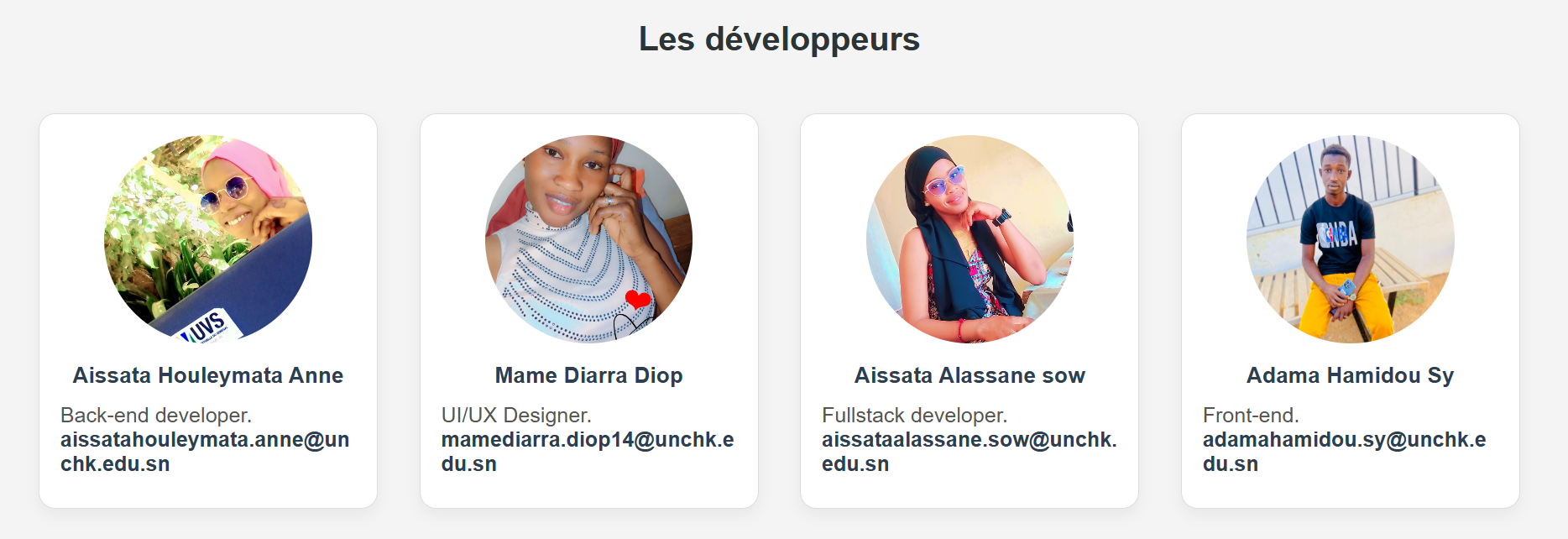
**Étape 4 : Élaborer la matrice RACI**

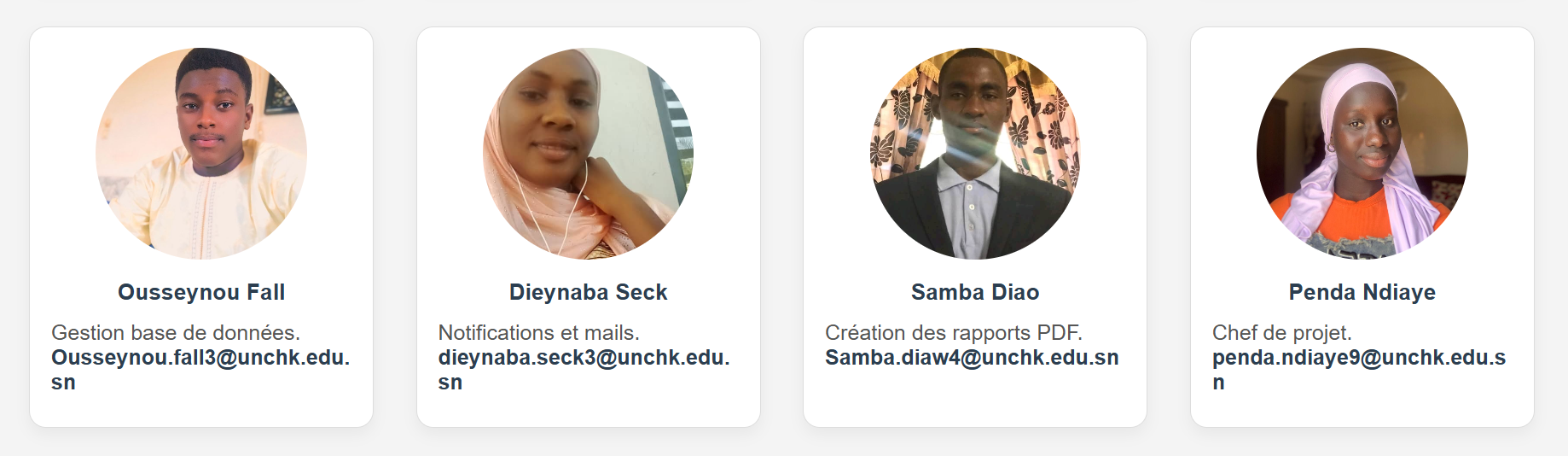
**Légende :**

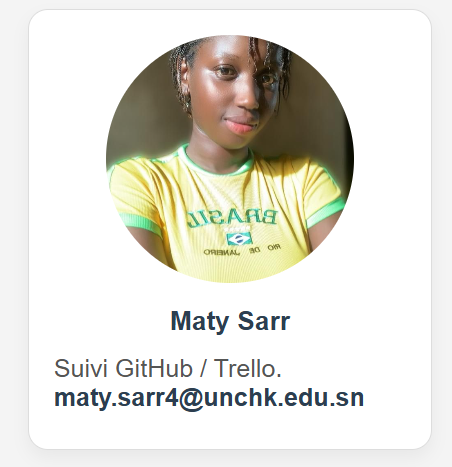
* **R** = Responsable (fait le travail)
* **A** = Approbateur (valide le travail)
* **C** = Consulté (donne des conseils)
* **I** = Informé (est tenu au courant)

| **Tâche / Activité** | **Chef de projet** | **Développeur(s)** | **Designer UI/UX** | **Client (utilisateur)** | **Tuteur / Encadrant** |
| --- | --- | --- | --- | --- | --- |
| Définir les besoins | R | C | C | I | A |
| Rédiger les spécifications fonctionnelles | R | C | I | C | A |
| Réaliser les maquettes sur Figma | I | I | R | C | A |
| Modélisation UML | R | R | C | I | A |
| Création de la base de données | I | R | I | I | A |
| Développement de l'inscription / login | I | R | I | I | A |
| Développement gestion des tontines | I | R | I | I | A |
| Mise en place système de rotation | I | R | I | I | A |
| Implémentation notifications / rappels | I | R | I | I | A |
| Tableau de bord utilisateur | I | R | C | I | A |
| Fonction récupération de mot de passe | I | R | I | I | A |
| Génération de rapports | I | R | I | C | A |
| Sécurité des données (validation, sessions...) | R | R | I | I | A |
| Tests utilisateurs | C | C | I | R | A |
| Livraison finale et déploiement | R | R | I | I | A |
|  |  |  |  |  |  |

**Les membres du groupe : (développeur de cette application)**

******

******

******