

Ministère de l'Enseignement Supérieure
Et de la Recherche Scientifique

République de Côte d'Ivoire
Union - Discipline - Travail



Université
de Technologie
d'Abidjan



Niveau d'Etude : Licence3

**CAHIER DE CHARGES : APPLICATION
DE GESTION DE COMPTABILITE
UNIVERSITAIRE**

Année Académique : 2024 – 2025

Produit Par :

 **KOUAKOU YANN**

Table des matières

1. Contexte

Pour améliorer la gestion financière de l'université, l'administration a constaté un manque d'organisation dans le suivi des paiements des frais académiques des étudiants. Cela entraîne régulièrement des erreurs, des retards de traitement et une perte de temps pour le service de comptabilité. Face à cette situation, l'université décide de faire appel à une start-up technologique, dont tu es le fondateur, pour concevoir une **application de gestion de comptabilité universitaire**. Cette application aura pour but de faciliter la gestion des étudiants, l'enregistrement des versements, la production de rapports financiers.

2. Objectif

L'objectif principal de cette application est de **faciliter la gestion comptable des frais académiques** au sein de l'université. Elle permettra au service de comptabilité de :

- Suivre efficacement les versements effectués par chaque étudiant.
- Générer des rapports financiers fiables en temps réel.
- Réduire les erreurs manuelles et les pertes de données.
- Améliorer la transparence et la traçabilité des opérations financières.
- Optimiser le traitement des informations liées aux paiements étudiants.

3. Cible

Ce projet est destiné principalement au **service de comptabilité des établissements universitaires**, qu'ils soient publics ou privés. Il s'adresse également à toute institution d'enseignement supérieur souhaitant améliorer le suivi des paiements des étudiants, assurer une traçabilité financière rigoureuse, et faciliter la génération de rapports comptables pour une meilleure prise de décision.

4. Besoins

4.1 Besoins fonctionnels

➤ **Gestion des étudiants**

- Ajouter un nouvel étudiant.
- Modifier les informations d'un étudiant.
- Supprimer un étudiant.
- Consulter la liste des étudiants.
- Rechercher un étudiant par nom ou matricule.

➤ **Gestion des versements**

- Enregistrer un versement pour un étudiant.
- Modifier ou supprimer un versement.
- Lier chaque versement à une année académique.
- Afficher les versements par étudiant.
- Calculer le total versé par étudiant.
- Générer un reçu de paiement imprimable.

➤ **Gestion des années académiques**

- Ajouter une nouvelle année académique.
- Consulter les données par année.

➤ **Gestion des utilisateurs**

- Créer un compte utilisateur (Comptable, Administrateur).
- Modifier les informations d'un utilisateur.
- Supprimer un utilisateur.
- Définir les rôles et permissions (accès restreint selon le rôle).
- Changer son mot de passe.

➤ **Paramètres du compte**

○ **Modifier ses informations personnelles**

Nom, Prénom, Adresse, e-mail Numéro etc.

○ **Changer son mot de passe**

○ **Afficher les informations du compte**

➤ **Statistiques et rapports**

- Générer un rapport global des paiements sur une période donnée.
- Générer un rapport individuel par étudiant.
- Afficher des statistiques de paiements par niveau, filière, ou année académique.
- Exporter les rapports au format PDF ou Excel.

➤ **Notification**

○ **Notification de versement enregistré**

Lorsqu'un versement est effectué, l'étudiant reçoit une notification de confirmation (par mail).

○ **Rappel**

Avant chaque date limite de paiement, l'étudiant reçoit automatiquement un e-mail de rappel l'invitant à effectuer son versement afin d'éviter tout retard.

➤ **Sécurité et accès**

- Authentification par identifiants (email + mot de passe).
- Récupération de mot de passe

4.2 Besoins non fonctionnels

- L'application doit afficher des messages d'erreur clairs et utiles à l'utilisateur en cas de problème
- Fournir une documentation détaillée pour les développeurs et les administrateurs sur la configuration, la gestion et les mises à jour du système.
- Les utilisateurs doivent s'authentifier via un système de connexion par mot de passe sécurisé avec un mécanisme de hashing.

4.3 Besoins techniques

- **Back end** : PHP, Framework (Laravel)
- **Base de données** : PostgreSQL
- **Frontend** : HTML, CSS, JavaScript (avec Bootstrap)

5. Acteurs du projet

Rôle	Nom & Prénoms
Maître d'œuvre	KOUAKOU YANN
Chef Projet	

6. Planification du projet