

REPUBLIQUE DU CAMEROUN

PAIX - TRAVAIL - PATRIE

UNIVERSITE DE YAOUNDE I

**ECOLE NATIONALE SUPERIEURE
POLYTECHNIQUE DE YAOUNDE**

**DEPARTEMENT DU GENIE ELECTRIQUE ET
TELECOMMUNICATIONS**



REPUBLIC OF CAMEROON

PEACE - WORK - FATHERLAND

UNIVERSITY OF YAOUNDE I

**NATIONAL ADVANCED SCHOOL OF
ENGINEERING YAOUNDE**

**ELECTRICAL AND
TELECOMMUNICATIONS ENGINEERING**

UE : EC 3522 PROGRAMMATION PYTHON

AMELIORATION D'UN LOGICIEL DE GESTION DES TONTINES

Groupe 7 :

Membres

NOMS	MATRICULES	PARTICIPATIONS
MANFOUO LONTSI SORELLE (CHEF)	23P248	22%
MINLO MINLO YAN RODRIGUE	25P861	20%
NJOYA YEYAP LEONEL	21P212	18%
ZEBAZE MAUREL NICOLAS	21P208	20%
ZEH YVAN PIERRE DONALD	21P105	20%

EXAMINATEUR : MR. MBIETIEU AMOS

ANNEE ACADEMIQUE : 2025/2026

RAPPORT FINAL DU PROJET

I. INTRODUCTION

Ce rapport présente l'état d'avancement du projet "Amélioration d'un logiciel de gestion de tontines". L'objectif principal est d'analyser plusieurs projets existants de gestion de tontines, de comprendre leur fonctionnement, et de sélectionner celui sur lequel des améliorations seront apportées. Nous avons utilisé des critères tels que la complétude de la modélisation UML, l'architecture logicielle, les fonctionnalités proposées et la documentation disponible pour comparer ces projets.

Ce projet permet d'appliquer nos connaissances en **programmation Python**, tout en développant une solution plus performante pour la gestion des tontines. L'objectif final est d'améliorer la convivialité, la fonctionnalité et l'efficacité des applications existantes.

II. LES GRANDES LIGNES DU PROJET

1. CLONAGE

Cette étape consiste à la mise en place de l'environnement de travail (GitHub) et rechercher initialement les différents projets. Ceci grâce aux commandes **cd desktop** et **git clone** que nous allons saisir dans **l'invite de commande** de l'ordinateur.

2. ORGANISATION DU TRAVAIL

Nous présentons ici la répartition des tâches pour chaque membre du groupe. Sachant que le groupe est constitué de cinq membres.

Membres	Tâches affectées
Manfouo sorelle	Gestion de projet et coordination, lire et faire marcher le projet 3 plus rédaction du rapport.
Minlo rodrige	Proposition d'implémentation des améliorations, relecture et vérification du rapport.
Njoya leonel	Etude du fonctionnement du projet 1 et réalisation de l'analyse comparative.
Zebaze nicolas	Lecture et exécution du projet 2 et réalisation de l'analyse comparative.
Zeh donald	Analyse du projet 5 et réalisation de l'analyse comparative.

3. CHOIX DU PROJET DE REFERENCE

L'objectif de cette analyse est d'évaluer quatre projets de développement d'une application web de gestion de tontines, réalisés dans le cadre d'un cours de génie logiciel et programmation Python. Les critères retenus pour cette comparaison sont les suivants :

- Complétude de la modélisation UML : Présence et qualité des diagrammes (cas d'utilisation, séquence, classes, déploiement).
- Structuration technique : Architecture logicielle (MVC/MVT), organisation des modules, gestion des vues et modèles.
- Fonctionnalités couvertes : Gestion des membres, tontines, prêts, remboursements, dons, tableau de bord, etc.
- Documentation et guides : Clarté, accessibilité, présence de manuels d'utilisation.
- État d'avancement et faisabilité : Niveau de réalisation, tests, déploiement.

Nous avons choisi ces critères car après lecture des rapports des différents groupes, le constat fait est que ceux-ci respectent tous une même structure à savoir : l'utilisation de la même base de données (MySQL) plus précisément celle de professeur déjà existant et non celle de Django, le respect des fonctionnalités basiques des tontines réelles (souscrire à la tontine, faire des prêts, des dons, des remboursements, ajouter des membres...).

Tableau Comparatif des 4 Projets

Critère	Projet 1	Projet 2	Projet 3	Projet 5
Modélisation UML	Diagrammes de base (cas d'utilisation, séquence, déploiement)	Diagrammes simplistes, peu détaillés	Diagrammes complets et clairs (cas d'utilisation, séquence, classes, déploiement)	Diagrammes présents mais peu exploitables
Architecture	MVT mentionné, structure Django basique	MVT partiellement compris	MVT bien maîtrisé, séparation claire des rôles	MVT partiellement appliqué
Fonctionnalités	Modules de base (membres, tontines, prêts, dons)	Fonctionnalités limitées, interface simpliste	Modules avancés (prêts avec garant, dons en nature, statistiques, remboursements partiels)	Fonctionnalités basiques, peu innovantes
Documentation	Guide d'utilisation présent mais sommaire	Documentation technique faible	Documentation complète : guide utilisateur, rapport technique, exemples concrets	Documentation absente ou très légère
État d'avancement	Partiellement réalisé, tests manquants	Développement en cours,	Développement abouti, backend à	Développement débutant, peu

		interface non finalisée	80%, interfaces fonctionnelles	d'avancées concrètes
Interface utilisateur	Maquettes basiques, navigation simple	Interface rudimentaire	Interface moderne et intuitive, responsive, tableau de bord graphique	Interface non développée

Conclusion : Justification du Choix du Projet 3

Malgré certaines forces du Projet 1 (notamment sa simplicité et sa structure modulaire de base), c'est le Projet 3 qui a été retenu pour les raisons suivantes :

- **Modélisation UML complète et professionnelle** : Le Projet 3 propose des diagrammes détaillés et exploitables, ce qui facilite la maintenance et l'évolution.
- **Architecture MVT bien comprise et appliquée** : La séparation des responsabilités (modèles, vues, Template) est claire, ce qui assure un code maintenable et évolutif.
- **Fonctionnalités avancées et réalistes** : Gestion des prêts avec garant, dons en nature, remboursements échelonnés, et tableaux de bord statistiques.
- **Documentation exhaustive** : Un guide utilisateur détaillé et un rapport technique complet accompagnent le projet, ce qui est essentiel pour un déploiement réel.
- **Interface utilisateur soignée** : Des maquettes modernes et une expérience utilisateur pensée, avec un tableau de bord graphique et une navigation intuitive.

Alors le Projet 1, bien que solide dans sa structure de base, pêche par son manque de profondeur fonctionnelle et son avancement moins abouti. Le Projet 3, en revanche, se distingue par sa maturité, son professionnalisme et son potentiel de déploiement immédiat. Il constitue une base idéale pour un projet réel ou une poursuite de développement.

En conclusion, nous choisissons **le projet 3**. La base de données utilisée ici est **MySQL Workbench**.

4. PISTES D'AMELIORATION

Il est question ici pour nous d'apporter des améliorations à l'application afin qu'elle puisse répondre aux attentes d'une tontine réelle. Pour cela, nous avons identifié 03 modifications que nous pourrions effectuer sur l'application et cela repartit en divers modules de l'application.

- Dans le module prêt nous allons ajouter la fonctionnalité permettant de dire dans quelle tontine le membre effectue son prêt. Car cela n'avait pas été spécifier.
- Dans le module de remboursement nous allons spécifier une période après laquelle la dette doit être remboursée en fonction de la tontine où l'on effectue le prêt. Et aussi tenir compte des intérêts dans le cas où après cette date le remboursement n'a toujours pas été effectué.

- Ajouter la fonctionnalité permettant de préciser la durée de cotisation pour chaque type de tontine existante.

5. DIFFICULTES RENCONTREES

Tout au long de notre devoir, nous avons rencontrées de nombreuses difficultés que nous allons énumérer ci-après.

Tout d'abord, nous avons rencontré des problèmes lors du clonage des projets dans les différentes machines des membres du groupe. Certains membres du groupe n'ont pas pu le faire dans leurs machines et ont juste pris les fichiers obtenus après clonage chez ceux qui ont pu le faire. Comme autres problèmes

- Configuration de l'environnement : Difficulté à installer PyCharm et SQL. Pour résoudre ce problème nous avons configurer le serveur puis vérifier la compatibilité entre la version de pycharm, MySQL et le système de la machine.
- Création de la base de données : Problème pour créer et gérer la base de données. Ici, nous avons utilisé les commandes SQL dans le terminal afin de résoudre le problème
- Problème d'ouverture du projet sur certains Pc. Pour pallier à ce problème, nous avons mis à jour les pilotes et composants systèmes de la machine afin que cela puisse être possible.
- La plateforme qui bug à tout moment suite aux problèmes de connexion.
- Nous avons rencontré beaucoup de difficultés lors de la mise en place des modifications. Mais nous avons pu faire quelques avec beaucoup de difficultés. Les autres n'ont pas aboutis au résultat attendu.
- Nous avons également rencontré des difficultés pour créer le lien GitHub.

6. PLANNIFICATION DU PROJET-DIAGRAMME DE GANTT

Le Diagramme de Gantt ici présente les principales tâches, leur durée planifiée et leur état d'avancement du rapport. D'après le diagramme, en ce jour nous devrions déjà avoir terminer les pistes d'amélioration mais ce n'est pas le cas. Nous rencontrons beaucoup de difficultés pour faire des améliorations car nous n'avons pas la maitrise des langages de programmations utilisés pour la réalisation de cette tâche.

Néanmoins, nous continuons de faire des recherches afin de palier à ce problème majeur qui nous bloque à ce niveau. Ceci dans le but de pouvoir atteindre l'objectif prédéfini plus haut

7. LIEN GITHUB DU PROJET

Lien GitHub :

<https://github.com/manfouosorelle-collab/3GELE-Projet-GestionTontine-Groupe7.git>

8. AMELIORATIONS TERMINEES

Nous avons pu accomplir une seule modification qui est celle permettant aux membres de faire le choix de la tontine ou ils doivent effectuer le prêt. Et nous avons la forme suivante :

The screenshot shows a web browser window with the URL 127.0.0.1:8000/demande_pret/. The page title is "Soumettre une demande de prêt". The form fields include:

- Montant du prêt (FCFA)**: A text input field with placeholder "Ex: 50000" and a note "Montant minimum: 1 000 FCFA".
- Observations / Motif**: A text area with placeholder "Décrivez l'utilisation prévue du prêt...". Note: "Maximum 500 caractères".
- Choisir un garant**: A dropdown menu with placeholder "Sélectionner un membre --". Note: "Sélectionnez un autre membre comme garant".
- Choisir une tontine**: A dropdown menu with placeholder "Sélectionner une tontine --". Note: "Sélectionnez la tontine dans laquelle vous souhaitez faire le prêt".
- CNI du garant (image)**: A file upload input field with placeholder "Choisir un fichier" and note "Aucun fichier n'a été sélectionné". Note: "Formats acceptés: JPG, PNG, PDF. Taille max: 5MB".

At the bottom are two buttons: "[← Retour](#)" and a green "[Soumettre](#)" button.

Les autres améliorations n'ont pas aboutis car nous rencontrions plusieurs difficultés pour y parvenir.

GanttProject [gannt project 2.gan]

- □ ×

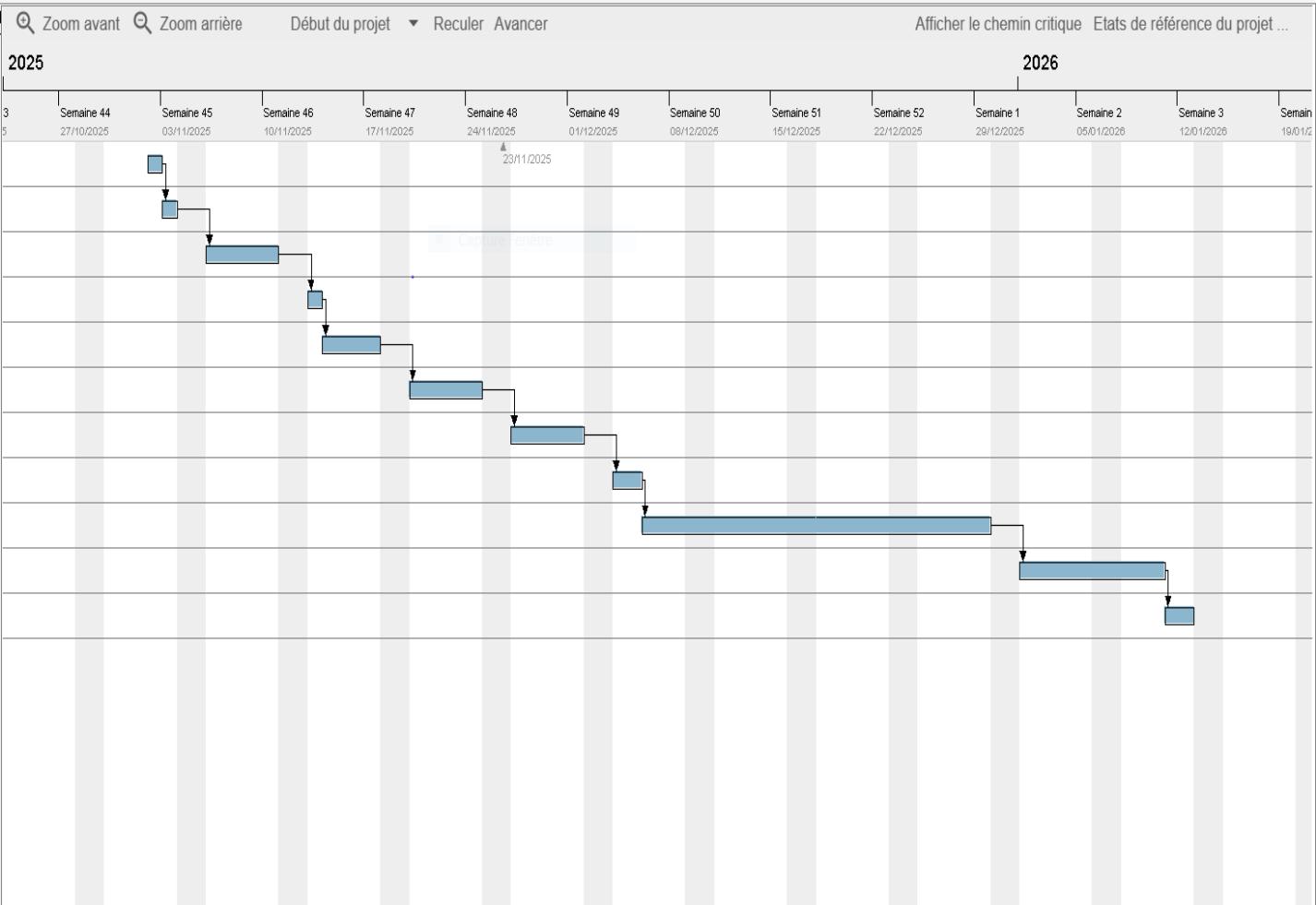
Projet Éditer Affichage Tâches Ressources Aide



Ctrl + F



Diagramme de Gantt Diagramme des Ressources



Cloud : Déconnecté

RSS Alertes Erreurs