Bujumbura, Mars/2023

**REPUBLIQUE DU BURUNDI**

MINISTERE DE L’EDUCATION NATIONALE ET DE LA RECHERCHE SCIENTIFIQUE

BUJUMBURA INTERNATIONAL UNIVERSITY

**Sous l’encadrement de :**

Msc. SHIKANEZA Alain

**FACULTE DE SCIENCES ET TECHNOLOGIES**

**Option : Génie logiciel**

**RAPPORT DE STAGE EFFECTUE AU SEIN DE L’EGLISE METHODISTE LIBRE AU BURUNDI**

**DU 23/Novembre/2022 AU 22/Janvier/2023**

**THÈME : Analyse, Conception et Réalisation d'une Application de Gestion de la Caisse de Pension « Cas de l’Eglise Méthodiste Libre au Burundi »**



**Par**

BIMENYIMANA Tony Blaise

BIU-EED-07-058

**DECLARATION**

Ce rapport de stage est mon travail original et n'a pas été présenté ailleurs pour l'obtention du diplôme.

Signature : \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Date: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

BIMENYIMANA Tony Blaise

BIU-EED-07-058

**APPROBATION**

Ce rapport de stage a été soumis pour la présentation avec l'approbation du Superviseur.

Signature: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Date: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

MSc. SHIKANEZA Alain

**REMERCIEMEMTS**

Je tiens à remercier toutes les personnes qui de prêt ou de loin ont contribué au succès de mon stage et qui m'ont aidé lors de la rédaction de ce rapport.

Tout d'abord, que mes vifs remerciements soient adressés à mon encadreur de stage, Mr SHIKANEZA Alain, Chef de Département de la faculté de Sciences et Technologies à Bujumbura International University, pour son accueil, le temps qu’il a consacré à mon encadrement et le partage au quotidien de son expertise.

Ensuite, je remercie Mgr. NSHIMIYIMANA Déogratias, Evêque et Représentant Légal de l’Eglise Méthodiste Libre au Burundi pour le temps qu’il m’a accordé et son assistance combien précieuse.

Par cette occasion, je ne m’abstiendrais de remercier mes chers parents : BIMENYIMANA Evariste et MBONIGARUYE Godelive ainsi que mes frères et sœurs pour leur appui tant matériel que moral. Que tout le monde trouve ici, l’expression de ma profonde gratitude.

**SIGLES, ACRONYMES ET ABBREVIATIONS**

API : (Application Programming Interface) : Interface de programmation d’applications

BIU : (Bujumbura International University) : Université International de Bujumbura

CAPEML  : Caisse Pension de l’Eglise Méthodiste Libre

CNI  : Carte Nationale d’Identité

CPU : (Central Processing Unit) : Unité centrale de traitement

CSS : (Cascading Style Sheet) : Feuille de style en cascade

DELL : (Digital Electronic Link Library) : Bibliothèque de liens électroniques Numériques

EMLBu : Eglise Méthodiste Libre au Burundi

GB : Giga Byte

HTML : (HyperText Markup Language) : Langage de balises pour l’hypertexte

HTTPS : (HyperText Transfer Protocol Secure): Protocole de transfert hypertexte sécurisée

IHM : Interface Homme Machine

MySQL : (My Structured Query Language) : Mon lagunage de requête structure

OS : (Operating System) : Système d’exploitation

Perl : (Practical Extraction and Reporting Language) : Langage pratique d’extraction et de rapport

PHP : (HyperText Pre-Processor) : Préprocesseur pour l’HyperText

RAM : (Rondom Access Memory) : Mémoire à accès aléatoire

RJ-45 : (Registered Jack-45) : Prise enregistréé-45

SSD : (Solid State Storage): Stockage à l’état solide

SSL : (Secure Socket Layer) : Couche de socket sécurisée

TLS : (Transport Layer Security) : Sécurité de la couche de transport

UML : (Unified Modeling Language) : Langage de modélisation unifies

XAMPP : X (toutes les plateformes), Apache, MySQL, PHP et Perl

**TABLEAUX**

[Tableau 1 : Plan du rapport de stage 3](#_Toc128661472)

[Tableau 2 : Devis simplifié de cout de déploiement 59](#_Toc128661473)

**FIGURES**

[Figure 1 : Organigramme de l’institution 10](#_Toc130332127)

[Figure 2: Planning provisionnel des taches 14](#_Toc130332128)

[Figure 3 : Exemple de Diagramme de cas d'utilisation 15](#_Toc130332129)

[Figure 4 : Exemple de diagramme de classe 16](#_Toc130332130)

[Figure 5: Exemple de diagramme de séquence 17](#_Toc130332131)

[Figure 6:Exemple de diagramme d'activité 18](#_Toc130332132)

[Figure 7: Présentation du diagramme de cas d'utilisation générale 21](#_Toc130332133)

[Figure 8: Présentation du diagramme de cas d'utilisation détaillé 22](#_Toc130332134)

[Figure 9: Présentation du diagramme de classe 23](#_Toc130332135)

[Figure 10: Le schéma explicatif des entités (utilisateur, conférence, district, paroisse, membre, enfant et conjoint) du diagramme de classe 25](#_Toc130332136)

[Figure 11:Le schéma explicatif des entités (membre, cotisation, abandon) du diagramme de classe 26](#_Toc130332137)

[Figure 12: Présentation du diagramme de séquence 27](#_Toc130332138)

[Figure 13: Présentation du diagramme d'activité d’enregistrement des cotisations 28](#_Toc130332139)

[Figure 14 : Présentation du diagramme d'objets 29](#_Toc130332140)

[Figure 15: Présentation du digramme de déploiement 30](#_Toc130332141)

[Figure 16: Modèle de cycle de vie en "V" 33](#_Toc130332142)

[Figure 17:Page d'authentification 37](#_Toc130332143)

[Figure 18:Présentation d'authentification échouée 38](#_Toc130332144)

[Figure 19: Les menus de l'application 39](#_Toc130332145)

[Figure 20: Les sous menus du menu "Eglise" 39](#_Toc130332146)

[Figure 21: Les sous menus de l'utilisateur 39](#_Toc130332147)

[Figure 22: Tableau de bord de l'application 40](#_Toc130332148)

[Figure 23: Page de liste des districts 41](#_Toc130332149)

[Figure 24: Page d'ajouter un nouveau district 42](#_Toc130332150)

[Figure 25:Liste des districts avec le message de succès après l'enregistrement 43](#_Toc130332151)

[Figure 26: Page de modification d'un district 44](#_Toc130332152)

[Figure 27: Message d'avertissement avant de supprimer un district 45](#_Toc130332153)

[Figure 28: Page de liste des paroisses 46](#_Toc130332154)

[Figure 29:Page d'ajouter une nouvelle paroisse 47](#_Toc130332155)

[Figure 30:Liste des membres 48](#_Toc130332156)

[Figure 31:Formulaire d'ajouter un nouveau membre 49](#_Toc130332157)

[Figure 32:Fiche d'identification personnelle d’un membre actif 51](#_Toc130332158)

[Figure 33:Fiche d'identification personnelle d'un membre abandonné 51](#_Toc130332159)

[Figure 34:Fiche d'identification personnelle d’un membre pensionné 52](#_Toc130332160)

[Figure 35:Page d'ajouter un abandon pour un membre 53](#_Toc130332161)

[Figure 36: Page d'ajouter un membre aux pensionnés 54](#_Toc130332162)

[Figure 37: Page pour ajouter des cotisations aux membres 55](#_Toc130332163)

[Figure 38: Liste des cotisations des membres 56](#_Toc130332164)

[Figure 39: Liste des membres abandonnés 57](#_Toc130332165)

[Figure 40: Liste des membres pensionnés 58](#_Toc130332166)

**TABLE DES MATIERES**

[CHAPITRE I : INTRODUCTION GENERALE 1](#_Toc130332242)

[I.1. Référence du plan de formation et mise en relation avec le stage 1](#_Toc130332243)

[I.2. Raison de faire ce stage 1](#_Toc130332244)

[I.3. Nature de l’institution où le stage a été effectué 2](#_Toc130332245)

[I.4. Raison du choix de l’institution spécifique où le stage a été effectué 2](#_Toc130332246)

[I.5. Activité principale de l’institution 2](#_Toc130332247)

[I.6. Plan du rapport de stage 3](#_Toc130332248)

[CHAPITRE II : DESCRIPTION DE L’EGLISE METHODISTE LIBRE AU BURUNDI 4](#_Toc130332249)

[II.1. Historique de l’institution 4](#_Toc130332250)

[II.2. Localisation géographique de l’institution 5](#_Toc130332251)

[II.4. Vision de l’institution 6](#_Toc130332252)

[II.5. Mission de l’institution 6](#_Toc130332253)

[II.6. Objectifs de l’institution 6](#_Toc130332254)

[II.7. Valeurs de l’institution 7](#_Toc130332255)

[II.8. Perspectives d’avenir de l’institution 7](#_Toc130332256)

[II.9. Services offerts par l’institution 9](#_Toc130332257)

[II.10. Organigramme de l’institution 10](#_Toc130332258)

[CHAPITRE III : ANALYSE ET DESCRIPTION DES ACTIVITES 11](#_Toc130332259)

[III.1. Présentation des services d’accueil et planification des activités durant la période de stage 11](#_Toc130332260)

[III.2. Description des activités réalisées 11](#_Toc130332261)

[III.3. Corrélation entre la réalité de terrain et les cours théoriques appris en classe 12](#_Toc130332262)

[III.4. Description du problème observé 12](#_Toc130332263)

[III.5. Analyse et critique du système existant 12](#_Toc130332264)

[CHAPITRE IV : CONCEPTION DU SYSTEME AVEC UN LANGAGE DE MODELISATION (UML) 14](#_Toc130332265)

[IV. 1. Planning provisionnel des taches de la réalisation du projet 14](#_Toc130332266)

[IV. 2. Présentation du langage d’UML 14](#_Toc130332267)

[IV.2.1. Avantages d’UML [2] 15](#_Toc130332268)

[IV.2.2. Abrégé de quelques diagrammes 15](#_Toc130332269)

[IV.3. Analyse et spécification des besoins 18](#_Toc130332270)

[IV.3.1. Identification des acteurs 18](#_Toc130332271)

[IV.3.2. Identification des besoins 18](#_Toc130332272)

[IV.4. Présentation formelle des besoins 20](#_Toc130332273)

[IV.4.1. Diagramme de cas d’utilisation générale 20](#_Toc130332274)

[IV.4.2. Diagramme de cas d’utilisation détaillé 22](#_Toc130332275)

[IV.4.3. Diagramme de classes 22](#_Toc130332276)

[IV.4.4. Diagramme de séquence 27](#_Toc130332277)

[IV.4.5. Diagramme d’activités 28](#_Toc130332278)

[IV.4.6. Diagramme d’objets 29](#_Toc130332279)

[IV.4.7. Diagramme de déploiement 30](#_Toc130332280)

[CHAPITRE V : PRESENTATION DE L’APPLICATION 31](#_Toc130332281)

[V.1. Outils de développement 31](#_Toc130332282)

[V.2. Modèle de cycle de vie du logiciel utilisé 33](#_Toc130332283)

[V.3. Les Tests 35](#_Toc130332284)

[V.3.1. Tests unitaires effectués 35](#_Toc130332285)

[V.3.2. Tests d’intégrations effectués 35](#_Toc130332286)

[V.3.3. Tests de validations effectués 35](#_Toc130332287)

[V.3.4. Sécurité du logiciel 35](#_Toc130332288)

[V.3.5. Manuel utilisateur du logiciel (les interfaces de l’application avec des explications) 37](#_Toc130332289)

[V.3.6. Cout estimatif du déploiement du logiciel 59](#_Toc130332290)

[CHAPITRE VI: CONCLUSION GENERALE ET SUGGESTIONS 61](#_Toc130332291)

[VI.1. Conclusion générale 61](#_Toc130332292)

[VI.1.1 Appréciation critique du déroulement du stage 61](#_Toc130332293)

[VI.1.2. Appréciation critique du problème observé 61](#_Toc130332294)

[VI.2. Suggestions 61](#_Toc130332295)

[REFERENCES 62](#_Toc130332296)

# **CHAPITRE I : INTRODUCTION GENERALE**

## **I.1. Référence du plan de formation et mise en relation avec le stage**

J'ai effectué un stage à l'Eglise Méthodiste Libre au Burundi afin de développer mes compétences, telles que la compréhension du fonctionnement de l'institution et de ses objectifs stratégiques, ainsi que l'acquisition d'une expérience pratique dans l'application des théories et des concepts appris en classe. J'ai également bénéficié d'une supervision de stage qui m'a aidé à améliorer mon apprentissage. Mon superviseur professionnel m'a également évalué à la fin de ma formation pour s'assurer que le stage était aligné sur les objectifs d'apprentissage

## **I.2. Raison de faire ce stage**

Le stage à l'Eglise Méthodiste Libre au Burundi permettra le développement d'une application de gestion de la caisse de pension qui facilitera entre autres :

* Accès à une population cible : L'Eglise Méthodiste Libre au Burundi compte un grand nombre de membres qui bénéficient d'un régime de pension. Un stage dans cette organisation permettrait pour moi d'accéder à une population cible pour tester et améliorer l'application de gestion de la caisse de pension.
* Compréhension des besoins de l'utilisateur : Ce stage offrira l'opportunité de comprendre les besoins spécifiques des utilisateurs en matière de gestion de la caisse pension, tels que les processus d'inscription et de gestion, les problèmes liés à l'identification des bénéficiaires.
* Découverte des processus et des systèmes de gestion existants : Ce stage offrira également la possibilité de découvrir comment les pensions sont actuellement gérées dans l'organisation et d'identifier les lacunes dans les processus existants. Cette information pourrait être utilisée pour améliorer l'application de gestion de la caisse de pension.
* Expérience pratique dans le développement de logiciels : Un stage dans le développement d'une application de gestion de la caisse de pension offrirait une expérience pratique dans le développement de logiciels, y compris la collecte et l'analyse des exigences, la conception de l'application, la programmation et le test.

## **I.3­. Nature de l’institution où le stage a été effectué**

L'Eglise Méthodiste Libre au Burundi est une communauté des croyants religieux. Elle est l'une des plus anciennes Églises protestantes du Burundi. Elle compte aujourd'hui plusieurs milliers de membres dans tout le pays et est présente dans de nombreuses communautés locales. L'Eglise Méthodiste Libre au Burundi se consacre à l'évangélisation et à la mission, ainsi qu'à l'aide sociale et à l'éducation. Elle est dirigée par un évêque et est organisée en districts et en paroisses. Elle a des relations fraternelles avec d'autres Églises Méthodistes dans le monde entier.

## **I.4. Raison du choix de l’institution spécifique où le stage a été effectué**

Après avoir constaté que la production des rapports des contribuables se fait avec beaucoup de contraintes et que la gestion des cotisations de la caisse de pension de l’EMLBu se faisait de façon très archaïque, c’est à dire qu’il ne dispose pas un logiciel de gestion des rapports des cotisations des membres de la CAPEML , j’ai pensé alors à développer un logiciel de gestion afin d’éviter les processus de traitement manuels pour rendre le travail facile.

## **I.5. Activité principale de l’institution**

L’Eglise Méthodiste Libre au Burundi se consacre principalement à l’évangélisation. Toutefois, en plus de cela elle contribue également dans le développement social et particulièrement dans les domaines de la santé et de l’éducation.

## **I.6. Plan du rapport de stage**

Tableau 1 : Plan du rapport de stage

|  |  |
| --- | --- |
| PERIODE | NOM DE L'ACTIVITE/MODULE COMPLET |
| 23/11/2022 - 30/11/2022 | Analyse du cas de l'existant |
| 1/12/2022 - 23/12/2022 | Implémentations des diagrammes UML |
| 26/12/2022-22/1/2023 | Développement et les tests de l'application |

# **CHAPITRE II : DESCRIPTION DE L’EGLISE METHODISTE LIBRE AU BURUNDI**

## **II.1. Historique de l’institution**

Rév . Wesley Haley de nationalité canadienne fut l’initiateur de l’Eglise Methodiste Libre au Burundi. Il passa plusieurs années à se preparer pour venir au Burundi. En date du 6 Avril 1931, il reçut un message via telegramme lui autorisant d’aller en Afrique Centrale. En date du 31 Mai 1932, Haley et son fils Blake, en compagnie d’un autre missionnaire du nom de Frank Adamson, prirent le chemin vers le Nord. Arrivés au Congo-Leopordiville, ils trouvèrent difficile de continuer en voiture, Blake et Frank Adamson retournèrent et Haley continua le voyage jusqu’à ce qu’il arriva au Burundi. Il passa deux semaines au Burundi contemplant ce que Dieu lui avait révelé .

Après avoir retourné en Afrique du Sud, il chercha l’occasion de retourner au Burundi; c’est par là qu’il eut écrit aux Chefs de Mission , disant: **“Si vous ne pouvez pas m’aider pour que je commence l’activité de la Mission au Burundi, vous pouvez me rayer de la liste des missionnaires et me laisser moi-même faire comme je peux.”**

Haley avait un esprit de dévéloppement et portait l’évangile dans les endroits non encore évangélisés. En date du 2 Mai 1935, il arriva à Muyebe où il installa une tente. Dans ce même mois, en date du 5, il commença à construire. A ce jour vinrent deux cents personnes, les uns puisèrent l’eau, les autres apportèrent des pierres pendant que les autres fabriquaient des briques. Dans peu de jours, six jeunes garcons et son domestique vinrent demander d’étudier; Haley fut très content et dit : **“Gloire à Dieu! Dans de très petits grains, Dieu fait pousser des gros arbres. Ooh! Si je pouivais voir Dieu utiliser notre égliser pour amener des milliers et des milliers des Burundais à Christ**.”( Soul A fire, par G. Bates, p. 45).

En date date du 22 Septembre 1935, au moment ou il venait de passer quatre mois entrain d’enseigner , il pensa qu’il fait bon moment d’appeller ceux qui voulaient confesser leurs péchés et ceux qui voulaient recevoir Jésus. Plus de cinquante personnes se mirent debout pour montrer leur volonté qu’on prie pour eux.

En 1985, l’Eglise Méthodiste Libre au Burndi fut provisoirement dotée d’une conférence générale (Conférence Juridictionnelle), elle fut dirigée par Nzeyimana Noé, le Premier Evêque, natif.Après quinze ans, en 2000, l’Eglise fut dirigée par Evêque Buconyori A. Elie jusqu’ en 2013. Suite à la mort de l’Evêque Buconyori A. Elie une année avant la fin d’un de ses mandats, Rév Nzigo Onésphore fut élu par le conseil d’administration de l’Eglise pour terminer la période d’une année qui restait. De 2014 à ce jour où le Livre de Discipline de l’Eglisée est révisé, l’Eglise est dirigée par Evêque Nshimiyimana Déogratias.

Dans sa deuxième session tenue en Novembre 2004 à Bujumbura,la Conférence générale trouva que le nom de l’Eglise « Eglise Libre Méthodiste  - E.L.M. », comme traduit au paravant de la langue anglaise à la langue française, n’est pas cohérent avec le nom originaire en langue anglaise et là où l’Eglise a commencé en Amérique où elle s’appelait « Free Methodist Church ». Ainsi, la conférence décida son changement pour enfin s’appeler « Eglise Méthodiste Libre ». De ce moment, l’Eglise s’appelle EGLISE METHODISTE LIBRE AU BURUNDI (EMLBu).

## **II.2. Localisation géographique de l’institution**

L’EMLBu est situé dans la zone Ngagara, quartier 2 en commune Ntahangwa dans la province de Bujumbura-Mairie, au Burundi précisément à la chaussée Mgr. Elie Buconyori (au Sud de l’Université Espoir d’Afrique, à l’Ouest se trouve la clinique Van Norman).

**II.3. Cadre légal de l’institution**

L’Eglise Méthodiste Libre au Burundi a été agréée pour la première fois par l'Arrêté Royal du 28 novembre 1961 dans le Bulletin Officielle du Rwanda Urundi, 3è année No 3 du 15 Février 1962, 39è No 3 du 15 Février 1962. Cette agrément est ensuite complété par la Prise d'Acte No 530/1369/CAB/2019 du 25 juin 2019 en Conformité avec la loi No 1/35 du 31 Décembre 2014 portant Cadre Organique des Confessions Religieuses au Burundi

## **II.4. Vision de l’institution**

La vision de l’Eglise Méthodiste Libre est d’être une communauté biblique saine de personnes saintes multipliant les disciples, les dirigeants, les groupes et les églises. Le chemin vers cette vision exige que chaque église locale soit une église saine avec un leadership rempli de l’Esprit travaillant sur un plan pour atteindre nos résultats attendus.

## **II.5. Mission de l’institution**

La mission de l’Eglise Méthodiste Libre au Burundi est de faire connaître à toutes les personnes partout dans le monde l’appel de Dieu à la plénitude par le pardon et la sainteté en Jésus-Christ, et d’inviter à devenir membre et d’équiper pour le ministère tous ceux qui répondent par la foi.

La Grande Commission biblique (Matthieu 28 :18-20 ; Actes 1 :8) ordonne aux croyants de faire des disciples en tout lieu et parmi tous les peuples.

## **II.6. Objectifs de l’institution**

L'Eglise Méthodiste Libre au Burundi a pour objectif de promouvoir la foi chrétienne et d'encourager la croissance spirituelle de ses membres. Elle se concentre sur l'enseignement de la Bible et la diffusion de la parole de Dieu dans la société burundaise. En plus de cela, l'Eglise Méthodiste Libre s'efforce également d'avoir un impact positif sur la vie sociale, culturelle et économique du pays.

## **II.7. Valeurs de l’institution**

L'Église Méthodiste Libre au Burundi est fondée sur les valeurs chrétiennes et prône l'amour, la paix, la justice et la compassion. Elle met l'accent sur l'importance de la famille, de la communauté et de la solidarité dans la vie chrétienne. Les valeurs clés de l'Église méthodiste libre au Burundi sont :

**L'amour** : L'Église Méthodiste Libre au Burundi met en avant l'amour de Dieu et l'amour du prochain. Elle encourage ses membres à aimer Dieu de tout leur cœur, de toute leur âme et de toute leur force, et à aimer leur prochain comme eux-mêmes.

**La paix** : L'Église Méthodiste Libre au Burundi promeut la paix en tant que valeur fondamentale de la vie chrétienne. Elle encourage ses membres à travailler pour la réconciliation et à être des agents de paix dans leurs communautés.

**La justice** : L'Église Méthodiste Libre au Burundi est engagée dans la promotion de la justice sociale. Elle prône l'égalité des chances pour tous et s'efforce de lutter contre la pauvreté et l'injustice.

**La compassion** : L'Église Méthodiste Libre au Burundi valorise la compassion et l'empathie envers les personnes dans le besoin. Elle encourage ses membres à aider les plus vulnérables de la société, tels que les orphelins, les veuves et les malades.

## **II.8. Perspectives d’avenir de l’institution**

Au cours des années à venir, le programme de développement de l'Église Méthodiste Libre au Burundi dans l’autonomisation sociale et économique de l'église visera à :

1. Disposer d'un fonds de pension des dirigeants ministériels et laïcs bien organisé et fonctionnel de l'Église Méthodiste Libre au Burundi répondant de manière adéquate aux besoins des pasteurs et des dirigeants laïcs à la retraite.

2. Avoir un programme de conférences annuelles bien organisé et fonctionnel, mobiliser et former les surintendants de district et les pasteurs de l'Église Méthodiste Libre au Burundi pour l'autonomisation économique avec du matériel pertinent et contextualisé pour l'éthique du travail, l'entrepreneuriat et l'éradication de la pauvreté et un personnel formé pour le mettre en œuvre.

3. Disposer d'un programme de services bancaires de microfinance bien organisé et fonctionnel aux besoins de prêts aux petites entreprises et d'investissement dans chaque église locale de l’Eglise Méthodiste Libre au Burundi avec des règles et des angulations pertinentes et contextualisées régissant les programmes de microfinance bancaires.

4. Avoir 4 hôpitaux bien organisés et fonctionnels et 10 dispensaires bien organisés et fonctionnels de l'Eglise Méthodiste Libre au Burundi répondant aux normes du Ministère de la Santé avec des équipements adaptés et modernes pour faire face aux défis liés à la santé.

5. Avoir une mutualité de santé bien organisée et fonctionnelle basée sur un programme ecclésiastique répondant aux questions énergétiques liés à la santé des pasteurs et des responsables laïcs de l'Église Méthodiste Libre au Burundi.

6. Avoir un programme de durabilité bien développé, organisé et fonctionnel avec différents projets fournissant des ressources pour soutenir le ministère de l'Église Méthodiste Libre au Burundi au niveau de la conférence générale.

7 Avoir mis en place des stratégies de collecte de fonds qui permettront à l'Eglise Méthodiste Libre au Burundi de construire au moins 50 bonnes églises, c’est à dire de bonnes églises chaque année.

8. Ayant construit le siège de l'Église méthodiste libre au Burundi accueillant les officiers de tous les départements de la conférence générale.

9. Avoir des imprimeries bien organisées et fonctionnelles, une librairie et une bibliothèque de l'église méthodiste libre au Burundi avec des matériaux pertinents et contextualisés pour chaque département pour le développement de l'église.

## **II.9. Services offerts par l’institution**

L'Eglise Méthodiste Libre au Burundi offre un certain nombre de services à la communauté, notamment :

**Services spirituels** : L'Église méthodiste libre offre des services spirituels tels que des cultes, des études bibliques, des groupes de prière et des services funéraires.

**Services sociaux** : L'Église méthodiste libre est impliquée dans des activités sociales telles que la fourniture d'une aide alimentaire, le soutien aux orphelins et aux veuves, et la création de centres de formation professionnelle pour les jeunes.

**Services éducatifs** : L'Église méthodiste libre a créé des écoles primaires, secondaires et des centres de formation professionnelle pour aider les jeunes à acquérir des compétences pour mieux se préparer à l'avenir.

**Services de santé** : L'Église méthodiste libre est impliquée dans la prestation de soins de santé dans les communautés locales. Elle a créé des centres de santé pour offrir des soins médicaux de base aux personnes qui n'ont pas accès aux soins de santé.

Services de conseil : L'Église méthodiste libre offre des services de conseil aux membres de la communauté qui ont besoin d'aide pour résoudre des problèmes personnels ou familiaux.

## **II.10. Organigramme de l’institution**

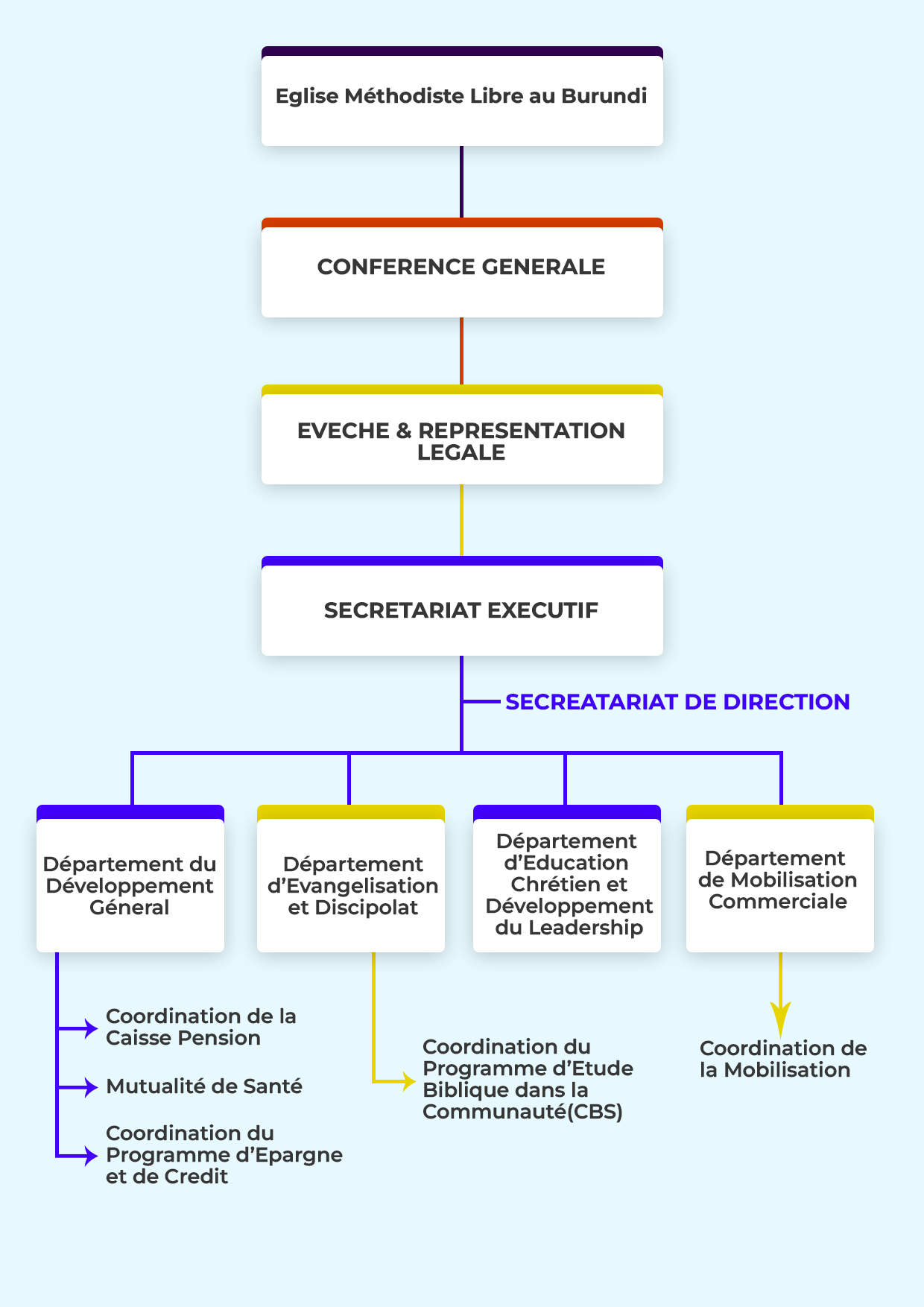
La figure 1 présente l’organigramme de l’Eglise Méthodiste Libre au Burundi :****

Figure 1 : Organigramme de l'entreprise [1]

# **CHAPITRE III : ANALYSE ET DESCRIPTION DES ACTIVITES**

## **III.1. Présentation des services d’accueil et planification des activités durant la période de stage**

Le service d'accueil de l'Eglise Méthodiste Libre au Burundi est un service essentiel pour assurer une expérience positive et chaleureuse aux visiteurs, membres de l'église et invités.

Alors, pour la première semaine de mon stage, j'ai prévu de m'introduire et de me familiariser avec l'Eglise Méthodiste Libre au Burundi dans le département de développement général, ses valeurs, les services offerts et son environnement de travail.

Semaine 2-3 : J'ai prévu de réaliser une étude de cas et d'analyser les besoins spécifiques de l'Eglise Méthodiste Libre au Burundi en matière de gestion de la caisse pension, en collaboration avec mon encadreur de stage qui est le coordinateur de la CAPEML.

Semaine 4-7 : J'ai prévu de concevoir et de développer l'application de gestion de la caisse pension en utilisant les compétences acquises précédemment et en tenant compte des besoins spécifiques de l'Eglise Méthodiste Libre au Burundi. Enfin j’ai prévu à faire des Tests et validation de l'application avec l'encadreur de stage et les utilisateurs et la finalisation de l’application.

## **III.2. Description des activités réalisées**

Pendant la période de stage à l'Eglise Méthodiste Libre au Burundi dans le département de développement général, j'ai appris les technologies de développement d'applications. J'ai collecté aussi les données nécessaires pour commencer à développer l'application en collaboration avec mon encadreur de stage. Nous avons identifié les problèmes et les avons résolus en utilisant des outils de développement appropriés.

## **III.3. Corrélation entre la réalité de terrain et les cours théoriques appris en classe**

Il existe une forte corrélation entre la réalité de terrain et les cours théoriques appris en classe. Les cours théoriques fournissent des connaissances fondamentales sur les concepts, les théories et les bonnes pratiques. Ces connaissances peuvent ensuite être mises en pratique lors de la réalisation de projets de développement de logiciels sur le terrain.

Les cours théoriques couvrent des sujets tels que les processus de développement de logiciels (dans le cours de programmation web avancé, IHM et structure de données et algorithme), les modèles de développement (dans le cours d’Atelier de l’ingénierie du logiciel), les méthodes de gestion de projet, les techniques de conception et de développement de logiciels (Dans le cours d’Outils informatique de gestion), les tests et la maintenance de logiciels, entre autres. Tous ces sujets sont pertinents et nécessaires pour la réalisation de projets de développement de logiciels sur le terrain.

## **III.4. Description du problème observé**

Durant la période de stage, j’ai observé certains problèmes courants dans la gestion de la caisse de pension comprennent :

Manque de suivi : Il est important de tenir un suivi régulier des contributions des membres afin de garantir la stabilité financière de la caisse de pension.

Mauvaise gestion des données : La gestion inadéquate des données peut entraîner des erreurs de calcul et la perte de données importantes, ce qui peut affecter la confiance des employés dans la caisse de pension.

## **III.5. Analyse et critique du système existant**

Le Département de Développement Général de l’EMLBu dispose une machine de bonne performance avec une imprimante pour imprimer les rapports, mais ne dispose pas un logiciel qui peut gérer la caisse de pension des membres ; ils utilisent des fiches en papier et peuvent même perdre les informations d’un membre quand le papier se déchire ou peut être en mauvais état quand il entre en contact avec l’eau ou l’humidité.

J’ai constaté aussi qu’il est difficile pour le coordinateur du département des pensions de gérer les districts, les conférences et les paroisses des membres suivant leurs catégories respectives et la gestion des conjoints ou conjointes des membres et leurs enfants. Avec le système existant, le coordinateur a aussi les difficultés de gérer les cotisations, gérer les membres qui sont pensionnés ou qui ont abandonné à cotiser et faire les calculs des frais de pension afin d’imprimer des rapports trimestriels ou annuels. En cas de l’indisponibilité du coordinateur, son secrétaire peut gérer les membres, gérer les cotisations des membres, gérer les pensions et les abandons.

**III.6. Proposition de solutions à ce problème**

La proposition qui pourrait aider à résoudre ce problème est d'utiliser une application de gestion. Celle-ci permettra de réduire les erreurs de traitement des données, d'améliorer la qualité comme l'impression des rapports pour chaque district et paroisse, de faciliter les calculs et d'assurer une transparence totale dans les statistiques des cotisations des membres.

# **CHAPITRE IV : CONCEPTION DU SYSTEME AVEC UN LANGAGE DE MODELISATION (UML)**

## **IV. 1. Planning provisionnel des taches de la réalisation du projet**

La figure ci-dessous montre retro planning des tâches journalier et hebdomadaire que j’ai effectué durant la période de stage.

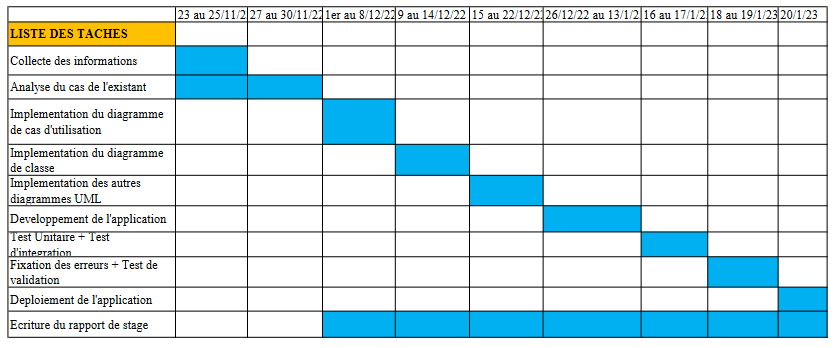


Figure 2: Planning provisionnel des taches

## **IV. 2. Présentation du langage d’UML**

Le langage de modélisation unifiée (UML) est un langage de modélisation normalisé permettant aux développeurs de spécifier, visualiser, construire et documenter les artefacts d'un système logiciel. Ainsi, UML rend ces artefacts évolutifs, sécurisés et robustes en exécution. UML est un aspect important impliqué dans le développement de logiciels orientés objet. Il utilise la notation graphique pour créer des modèles visuels de systèmes logiciels.

### **IV.2.1. Avantages d’UML [2]**

Les avantages d’UML sont les suivants :

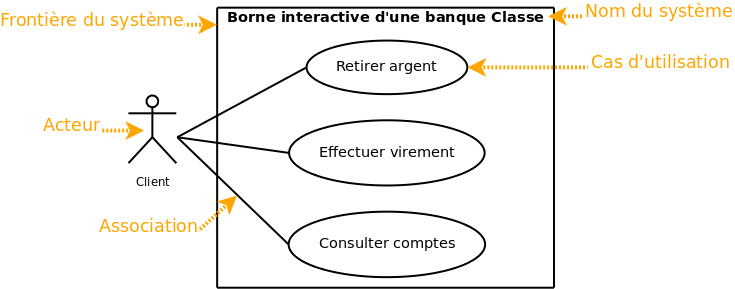
* UML est un langage formel et normalisé clair et précis. Encourage l'utilisation d'outils.
* UML est un support de communication performant. L’analyse devient plus facile. Facilite la compréhension de représentations abstraites complexes.

### **IV.2.2. Abrégé de quelques diagrammes**

**Diagramme de cas d’utilisation**

En langage de modélisation unifié (UML), un diagramme de cas d'utilisation peut servir à résumer les informations des utilisateurs de votre système (également appelés acteurs) et leurs interactions avec ce dernier. La création de ce type de diagramme UML requiert un ensemble de symboles et de connecteurs spécifiques. [3]

Figure 3 : Exemple de Diagramme de cas d'utilisation



**Diagramme de classes**

Un diagramme de classe est un type de diagramme UML qui décrit un système en visualisant les différents types d’objets au sein d’un système et les types de relations statiques qui existent entre eux. Il illustre également les opérations et les attributs des classes. Ils sont généralement utilisés pour explorer les concepts de domaine, comprendre les exigences logicielles et décrire les conceptions détaillées. [4]

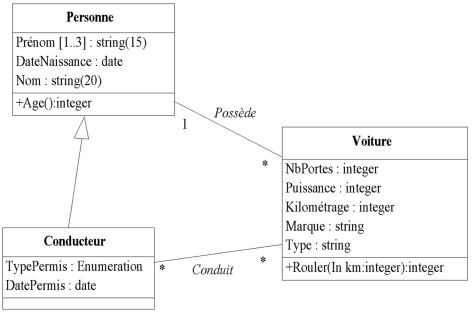


Figure : Exemple de diagramme de classe

**Diagramme de séquence**

Un diagramme de séquence est un diagramme UML qui représente la séquence de messages entre les objets au cours d'une interaction. Un diagramme de séquence comprend un groupe d'objets, représentés par des lignes de vie, et les messages que ces objets échangent lors de l'interaction. [5]

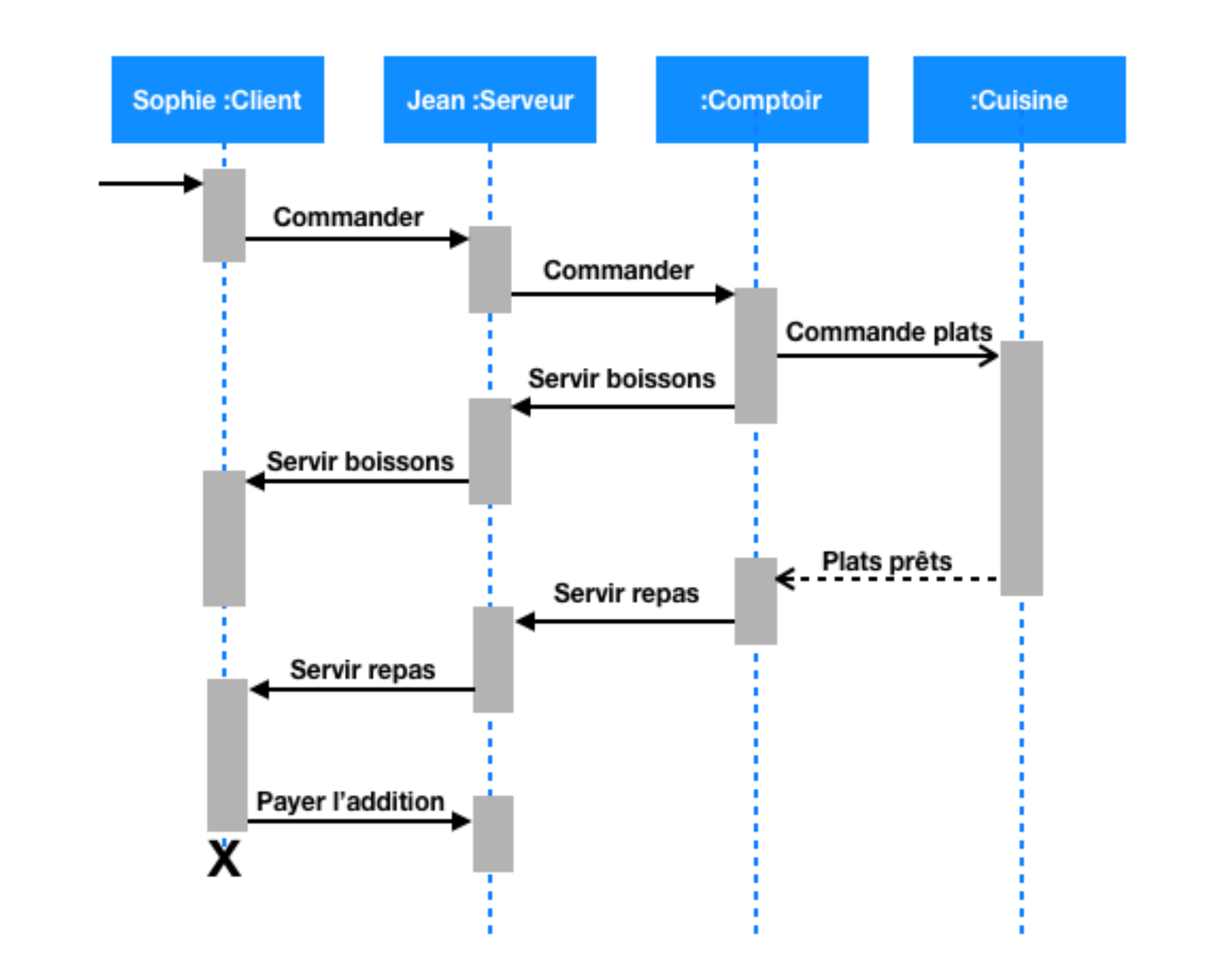


Figure 5: Exemple de diagramme de séquence

**Diagramme d’activité**

Dans le langage UML, un diagramme d'activité fournit une vue du comportement d'un système en décrivant la séquence d'actions d'un processus. Les diagrammes d’activité montrent les flux entre les actions dans une activité. Un diagramme d'activités UML est un organigramme illustrant les activités exécutées par un système. [6]

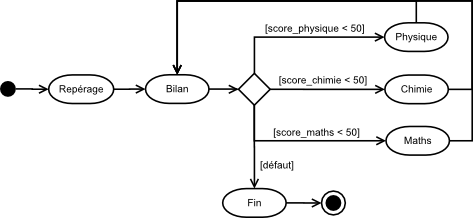


Figure 6:Exemple de diagramme d'activité

## **IV.3. Analyse et spécification des besoins**

### **IV.3.1. Identification des acteurs**

L’application de gestion de la caisse de pension (CAPEML) aura deux acteurs « Le coordinateur » et « Le secrétaire », sont des acteurs principaux; c’est-à-dire qu’ils vont interagir directement avec le système.

### **IV.3.2. Identification des besoins**

L’application que j’ai développé est composée par les besoins fonctionnels et non fonctionnels.

Les besoins fonctionnels :

Il s'agit des fonctionnalités du système. Ce sont les besoins spécifiant un comportement d'entrée / sortie du Système.

Le système doit permettre à l’utilisateur de :

* Gérer les conférences (Ajouter, Modifier, Supprimer) ;
* Gérer les districts (Ajouter, Modifier, Supprimer) ;
* Gérer les catégories des pasteurs (Ajouter, Modifier, Supprimer) ;
* Gérer les paroisses (Ajouter, Modifier, Supprimer) ;
* Gérer les membres (Ajouter, Modifier, Supprimer) ;
* Gérer les conjoints (Ajouter, Modifier, Supprimer) ;
* Gérer les enfants des membres (Ajouter, Modifier, Supprimer) ;
* Gérer les cotisations (Ajouter, Modifier, Supprimer) ;
* Gérer les abandons (Ajouter, Modifier, Supprimer) ;
* Gérer les pensionnés (Ajouter, Modifier, Supprimer) ;
* Gérer les utilisateurs (Ajouter, Modifier, Supprimer) ;
* Produire et imprimer les rapports annuels ou semestriels.

Les besoins non fonctionnels :

Il s'agit des besoins qui caractérisent le système. Ce sont des besoins en matière de performance, de type de matériel ou le type de conception. Ces besoins peuvent concerner les contraintes d'implémentation (langage de programmation, type SGBD, de système d'Exploitation...).

L’application doit être :

* Testable ;
* Facile à la maintenir ;
* Facile à utiliser ;
* Fiable à la sécurité des données ;
* Rapide.

## **IV.4. Présentation formelle des besoins**

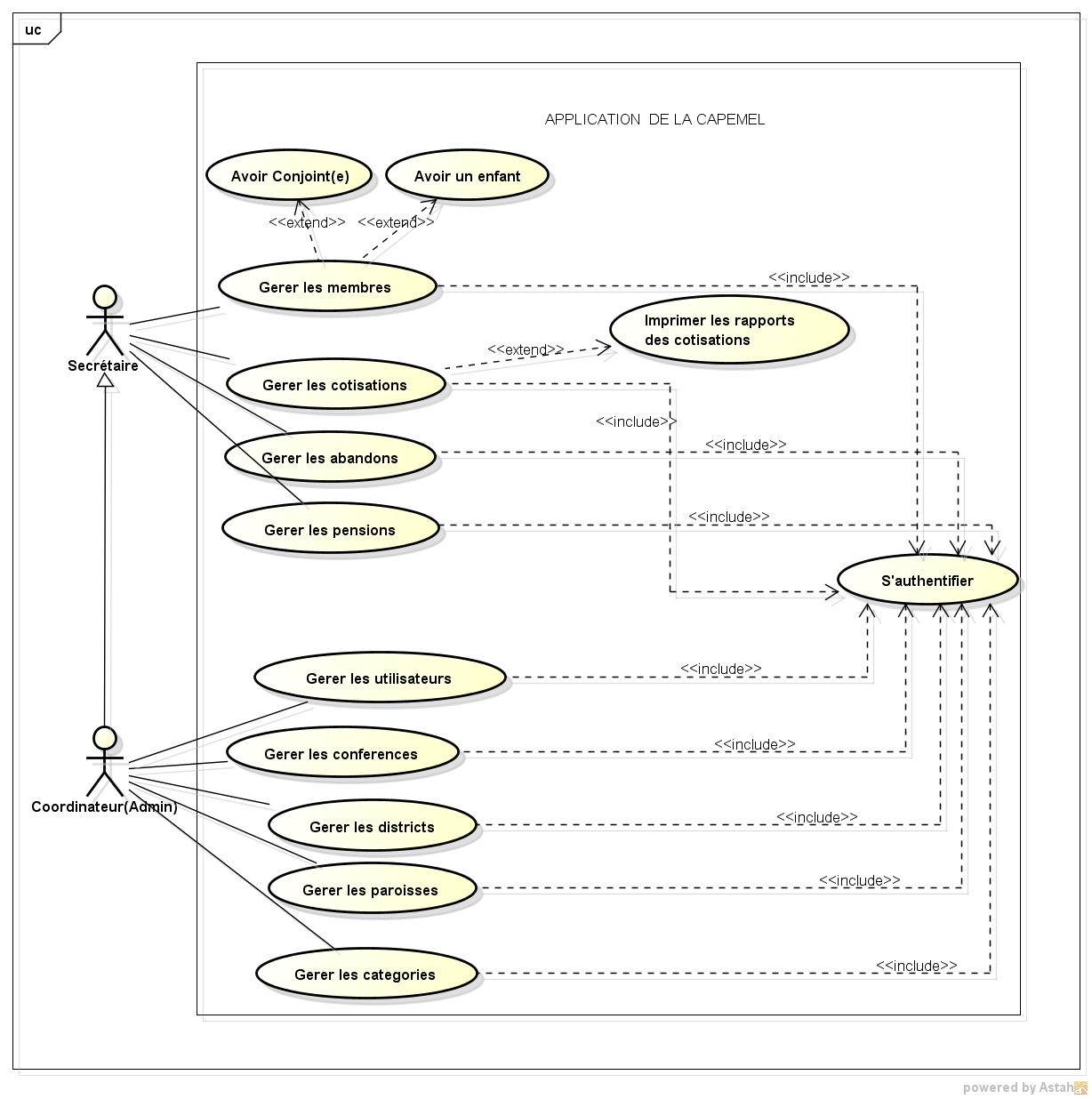
### **IV.4.1. Diagramme de cas d’utilisation générale**

La figure 7 montre comment le système du CAPEML fonctionne : le coordinateur du CAPEML va avoir accès à la gestion du système de la CAPEML en faisant des enregistrements et les modifications des différents modules. L’Administrateur (qui est le coordinateur de la CAPEML) va avoir toutes les permissions sur l’utilisation du système (Gérer les autres utilisateurs, gérer les membres, gérer les cotisations, gérer les pensions, gérer les abandons).

Pour la flèche « include » signifie que pour faire la deuxième opération, il faut d’abord passer la première là où la flèche se dirige vers.

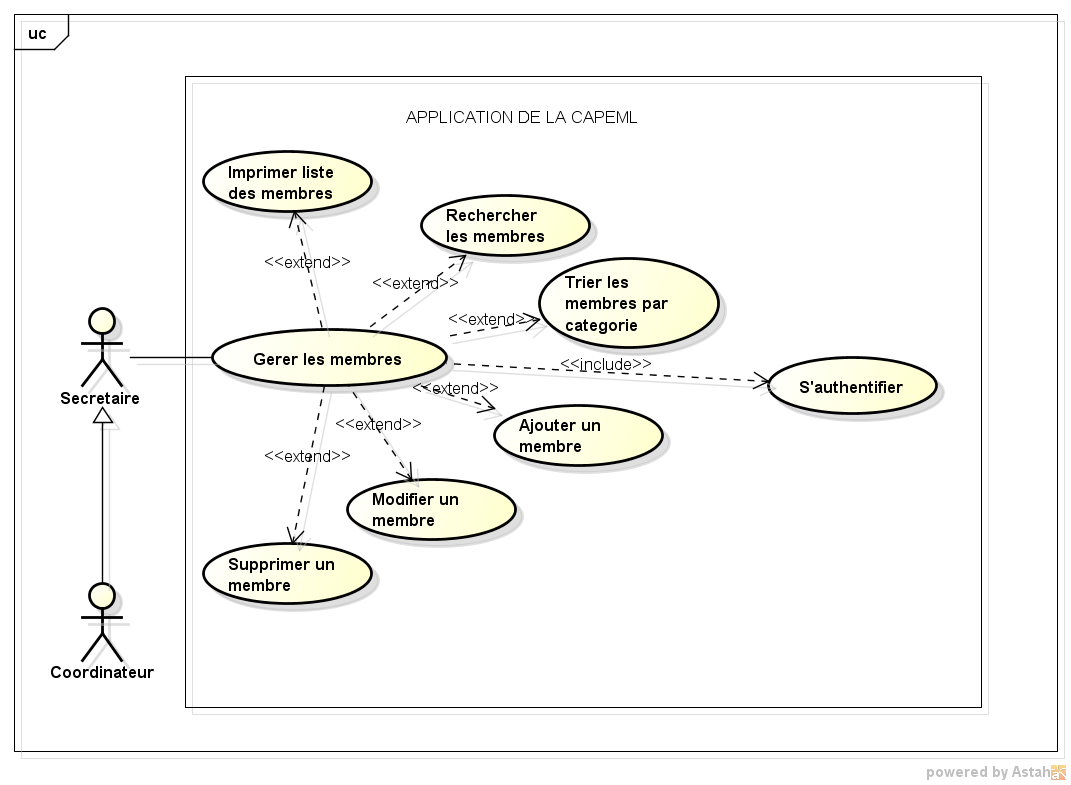
* Le coordinateur du CAPEML est comme l’administrateur de l’application et peut faire chaque fonctionnalité présente dans l’application.
* Le secrétaire a la possibilité de gérer les membres, gérer les pensions, gérer les abandons et gérer les cotisations.
* Les cas d’utilisation sont : (S’authentifier, gérer les conférences, gérer les districts, Gérer les paroisses, gérer les membres, gérer les cotisations, gérer les abandons, gérer les pensions, imprimer les rapports).

Pour la flèche « extend » signifie qu’il n’est pas obligatoire pour le membre d’avoir un conjoint ou une conjointe ou d’avoir un enfant.

Figure 7: Présentation du diagramme de cas d'utilisation générale

### **IV.4.2. Diagramme de cas d’utilisation détaillé**

Voici un diagramme de cas d’utilisation détaillé pour gérer les membres :

Figure 8: Présentation du diagramme de cas d'utilisation détaillé

### **IV.4.3. Diagramme de classes**

La figure 9 montre la communication entre les classes, les associations. Entre les classes il y a une simple association avec les noms de l’action et les multiplicités (ces multiplicités sont : 1, \*(plusieurs) ,0 ou 1,0 ou plusieurs, 1 ou plusieurs) ; par exemple, la relation entre « Membre » et « Enfant » : Un membre peut avoir 0 à plusieurs enfants et un enfant peut avoir un seul membre (comme son parent même si sa mère et son père sont tous membres).

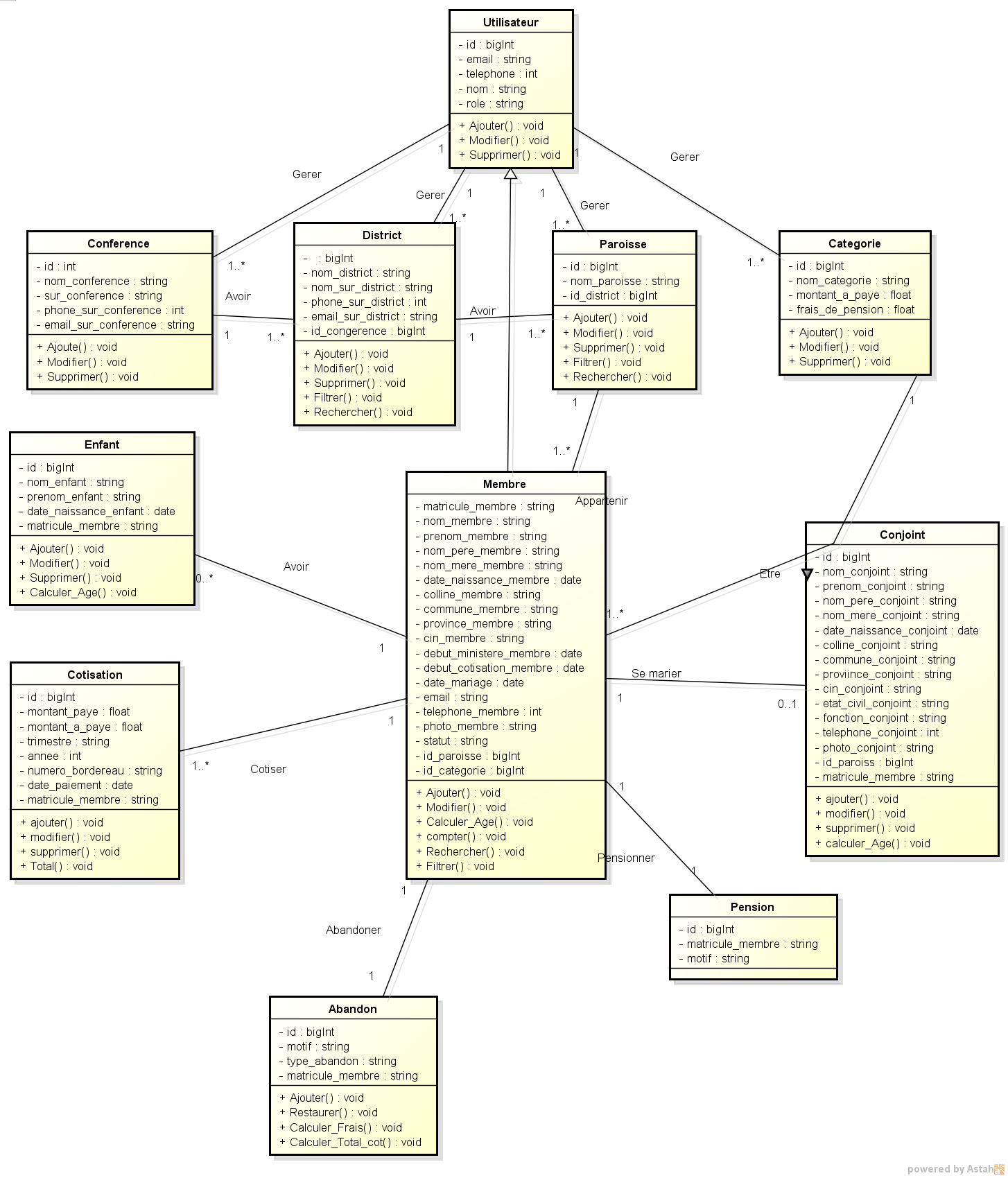


Figure 9: Présentation du diagramme de classe

La figure 10 montre quelques entités du diagramme de classe avec les multiplicités et les exemples au-dessous.

Un utilisateur peut gérer une ou plusieurs conférences de l’EMLBu, tandis que une conférence est géré par un seul utilisateur (Philémon gère la conférence Est et conférence Ouest , tandis que la conférence Nord est géré par Tony) ; une conférence peut avoir un ou plusieurs districts, tandis un district doit être d’une seule conférences ; comme exemple la conférence Ouest a le district Cibitoke et district Mwaro, tandis que le district Buhayira est de la conférence Est) ; un district peut avoir une ou plusieurs paroisses , tandis que une paroisse doit être d’un seul district ; un membre peut avoir zéro ou plusieurs enfants, tandis qu’un enfant doit être d’une seul membre ; un membre peut avoir zéro ou un(e) seul(e) conjoint(e), tandis que un(e) conjoint(e) doit être d’un seul membre.

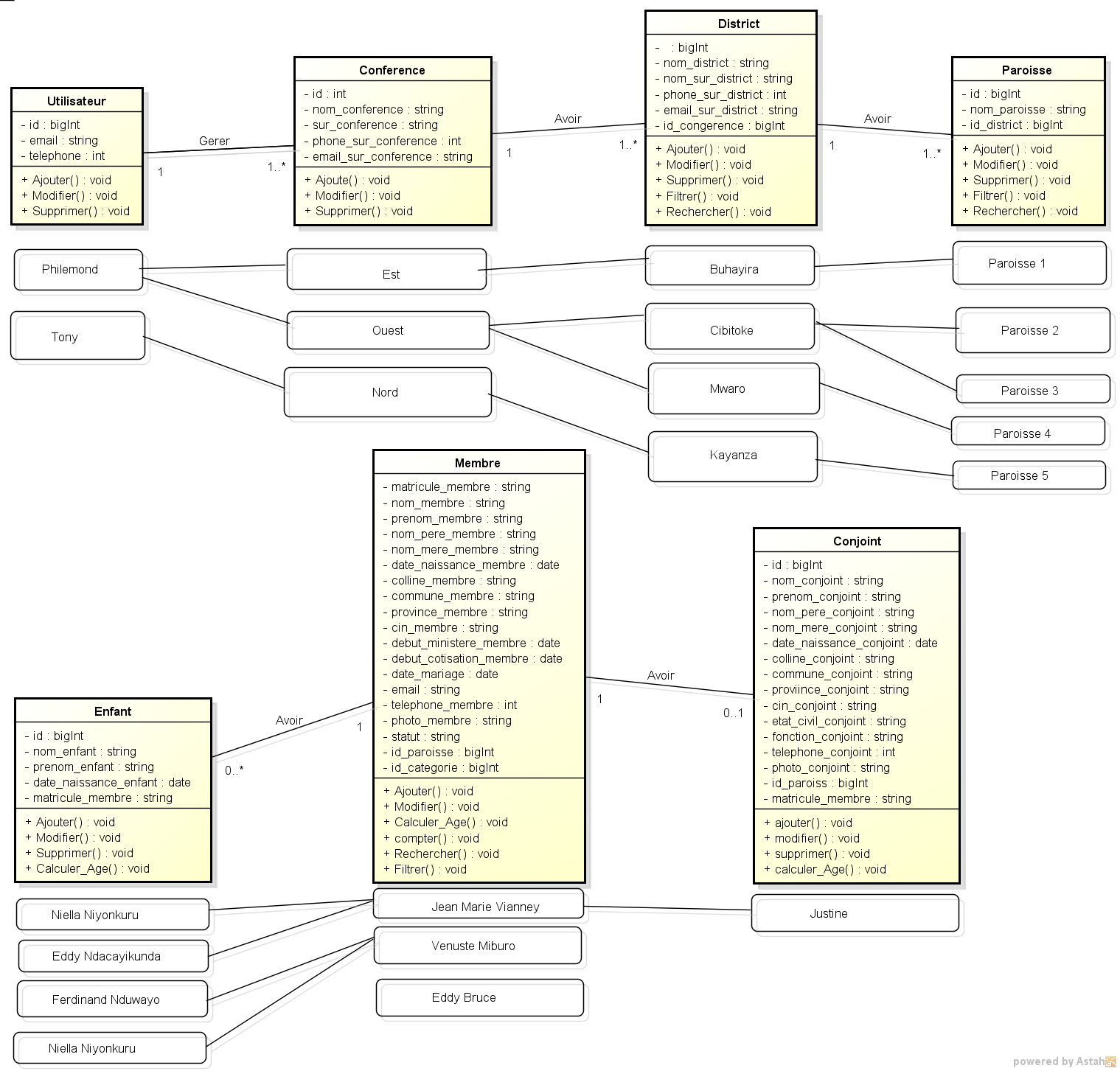


Figure 10: Le schéma explicatif des entités (utilisateur, conférence, district, paroisse, membre, enfant et conjoint) du diagramme de classe

La figure 11 montre quelques entités du diagramme de classe avec les multiplicités et les exemples au-dessous.

Un membre peut cotiser une ou plusieurs cotisations, tandis qu’une cotisation peut être cotisé par un seul membre ; un membre peut aussi abandonner une seule fois, alors qu’un abandon peut être d’un seul membre.



Figure 11:Le schéma explicatif des entités (membre, cotisation, abandon) du diagramme de classe

### **IV.4.4. Diagramme de séquence**

La figure 12 montre comment le coordinateur gère le système avec l’interaction entre le coordinateur et le système.

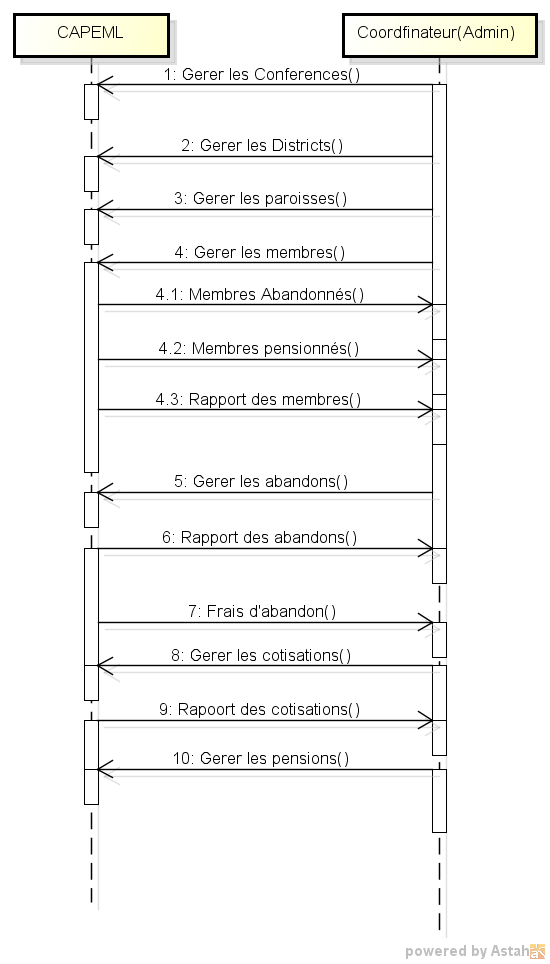


Figure 12: Présentation du diagramme de séquence

### **IV.4.5. Diagramme d’activités**

La figure 13 suivante montre comment toutes les activités d’enregistrement des cotisations du système dès la connexion du système, enregistrement des cotisations jusqu’à la déconnexion du système.

Les étapes des activités pour enregistrer les cotisations :

* L’utilisateur doit d’abord se connecter au system pour avoir accès au système.
* Si la connexion échoue il va recevoir un message d’erreur qui lui indique qu’il doit vérifier ses identifiants.
* Si la connexion réussie il entre directement dans le système.
* L’utilisateur va ensuite cliquer sur l’onglet cotisation.
* Il va ensuite cliquer sur le bouton « Ajouter cotisation ».
* L’utilisateur va ensuite compléter les champs nécessaires.
* S’il n’a pas bien complété les champs il va recevoir une erreur de validation.
* Après avoir effectué ces activités il va se déconnecter s’il veut.

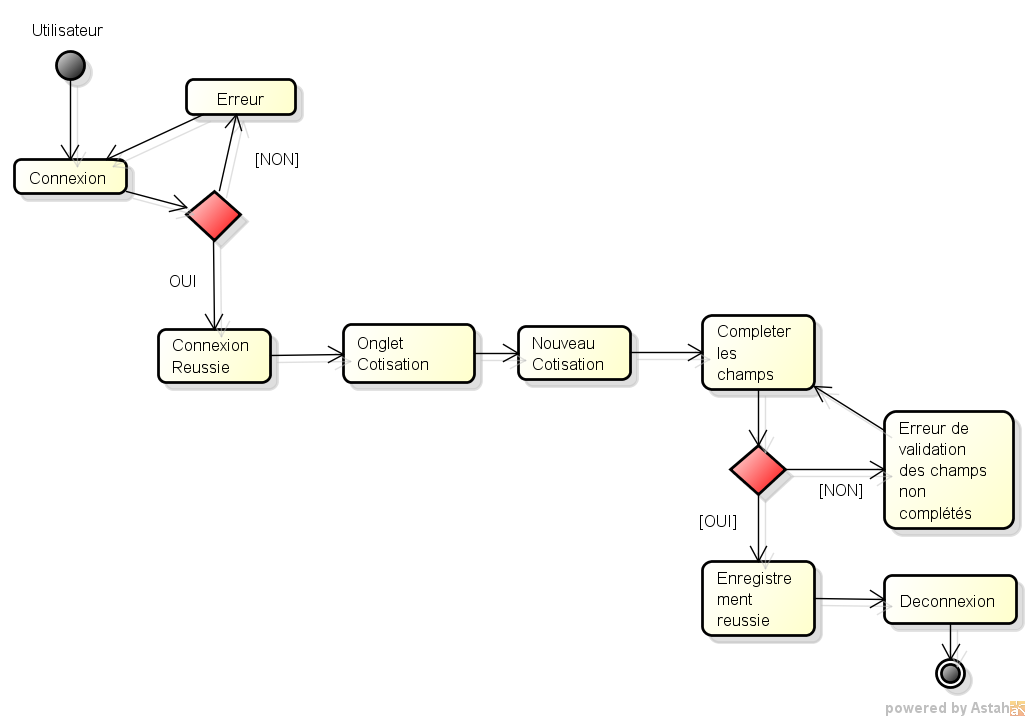


Figure 13: Présentation du diagramme d'activité d’enregistrement des cotisations

### **IV.4.6. Diagramme d’objets**

La figure 14 présente le diagramme d'objets représentant la situation de la paroisse Mirango. Cette paroisse compte deux membres, à savoir Mr. Laurent et Mr. Yvan.

Mr. Yvan a effectué deux paiements de cotisation au troisième trimestre et au quatrième trimestre de l'année 2022. Par la suite, Mr. Yvan a épousé Sylvie et a pensionné en raison d'une maladie.

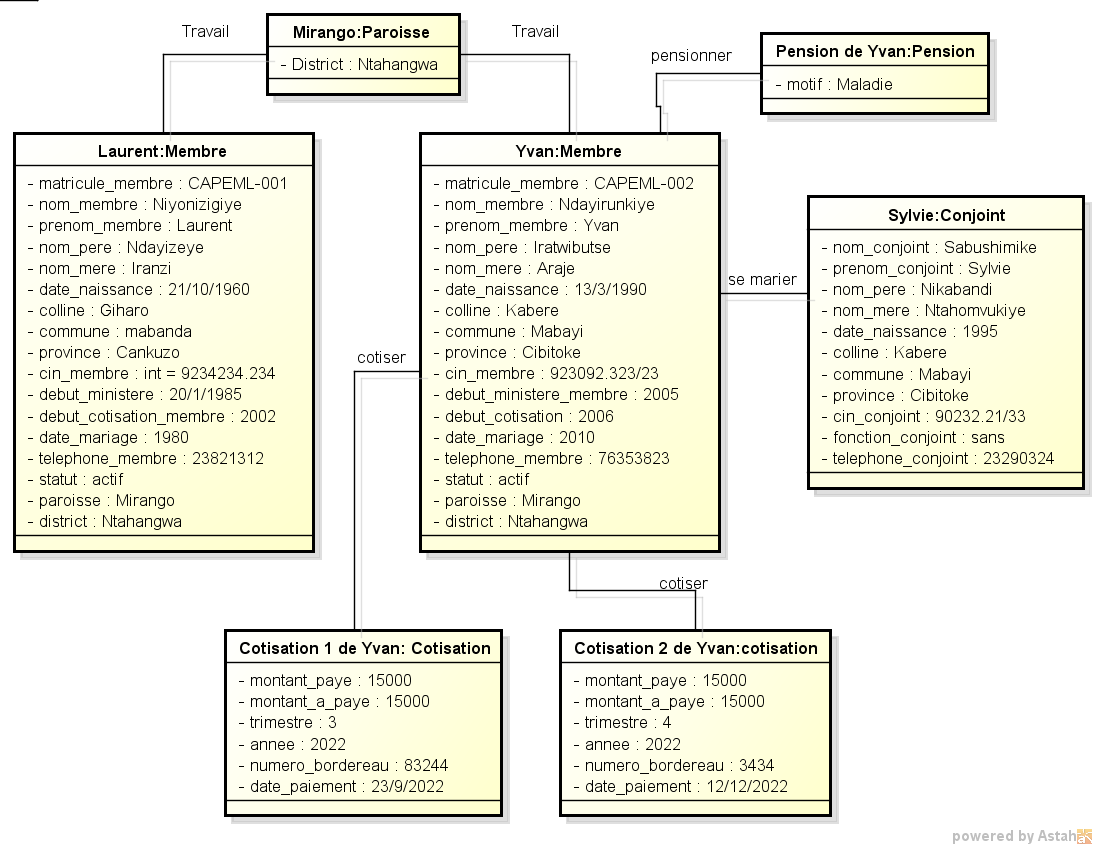


Figure 14 : Présentation du diagramme d'objets

### **IV.4.7. Diagramme de déploiement**

La Figure 15 montre le diagramme de déploiement dans lequel le coordinateur ou le secrétaire utilise un navigateur web pour accéder aux interfaces de l'application. Les fichiers tels que les images, les feuilles de style ou les pages HTML sont stockés sur le serveur web. Ces fichiers communiquent avec les contrôleurs, les modèles et les APIs pour accéder à la base de données MySQL du serveur via un fichier de connexion.

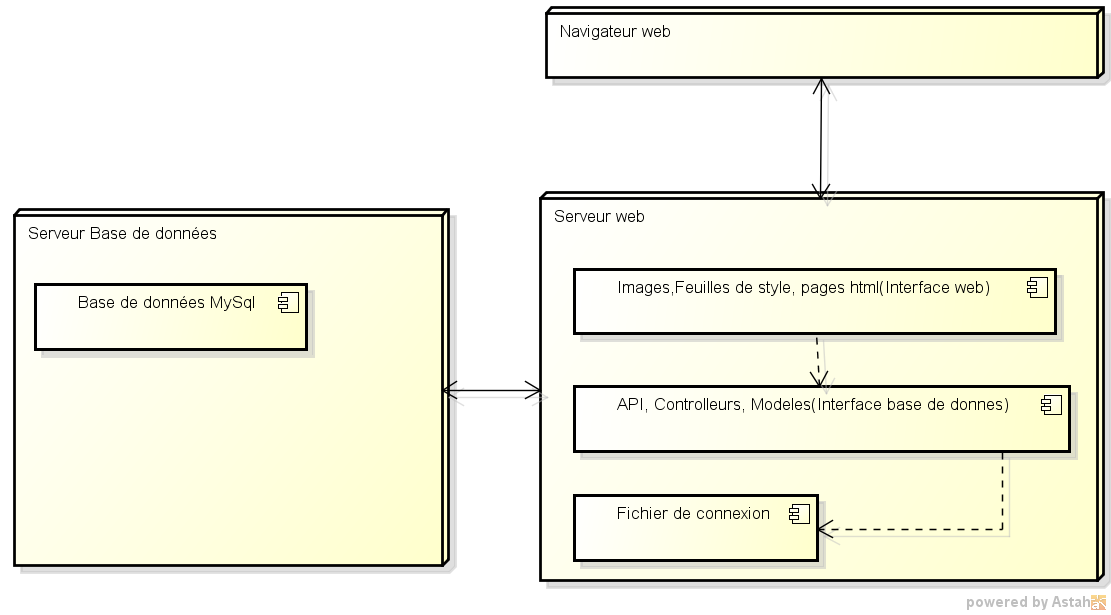


Figure 15: Présentation du digramme de déploiement

# **CHAPITRE V : PRESENTATION DE L’APPLICATION**

## **V.1. Outils de développement**

Les outils utilisés dans le développement du système sont :

* Windows 10,
* Astah Community : Astah communautaire est un outil de add-on gratuit pour UML qui est prêt à l'emploi dès que vous le télécharger, ce qui permet à vos diagrammes UML pour être rapidement et facilement améliorées, et affiner votre processus de développement avec l'approche de modélisation novateur de Astah.
* PHP : est un langage de script côté serveur. Il est utilisé pour développer des sites web statiques ou dynamiques ou des applications web. PHP signifie Hypertext Pre-processor, qui signifiait auparavant Personal Home Pages. Les scripts PHP ne peuvent être interprétés que sur un serveur sur lequel PHP est installé. Les ordinateurs clients qui accèdent aux scripts PHP ne nécessitent qu’un navigateur web. [7]
* LARAVEL : est un **Framework du langage de programmation PHP** créé par Taylor Otwel, ce framework regroupe les meilleures librairies utiles pour créer un site web. En outre, l’excellent framework laravel intègre aussi bien d’autres fonctionnalités exclusives comme l’utilisation des APIs. [8]
* JavaScript : désigne un langage de développement informatique, et plus précisément un **langage de script orienté objet.** On le retrouve principalement dans les pages Internet. Il permet, entre autres, d'introduire sur une page web ou HTML des petites animations ou des effets et de créer des contenues dynamiques. [9]
* Vue.Js : ou simplement Vue est donc un framework JavaScript, utilisé pour la partie front-end, qui a l’avantage d’être très accessible.
* HTML : signifie « HyperText Markup Language » qu'on peut traduire par « langage de balises pour l'hypertexte ». Il est utilisé afin de créer et de représenter le contenu d'une page web et sa structure.
* CSS : est l’acronyme de « Cascading Style Sheets » ce qui signifie « feuille de style en cascade ». Le CSS correspond à un langage informatique permettant de mettre en forme des pages web HTML.
* XAMPP : est un **ensemble de logiciels libres**. Le nom est un acronyme venant des initiales de tous les composants de cette suite. Ce dernier réunit donc le serveur Web **A**pache, la base de données relationnelle et système d’exploitation. [10]

**M**ySQL ou **M**ariaDB ainsi que les langages scripts **P**erl et **P**HP. L’initiale X représente tous les systèmes d’exploitation possibles, à savoir Linux, Windows et Mac OS X.

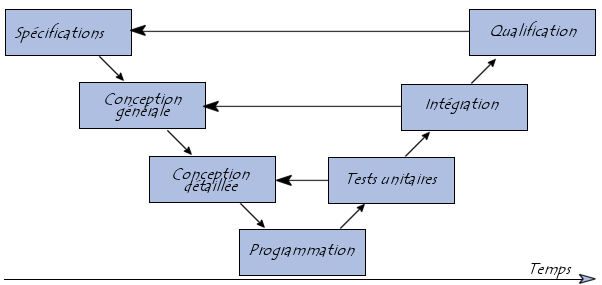
Apache : le serveur Web open source Apache est utilisé mondialement et permet de délivrer des contenus Web. L’application de serveur est mise à disposition en open source par l’Apache Software Foundation.

MySQL/MariaDB :avec MySQL, XAMPP se compose de l’un des systèmes de gestion de base de données relationnelle les plus populaires au monde. En combinaison avec le serveur Web Apache et le langage script PHP, MySQL sert à l’enregistrement de données pour des services Web. Les versions actuelles de XAMPP favorisaient MariaDB à l’insu de MySQL comme gestionnaire de base de données, marquant un détachement avec ce dernier.

## **V.2. Modèle de cycle de vie du logiciel utilisé**

Le cycle de vie du développement logiciel se définit comme l’enchaînement des étapes produisant des logiciels de la plus haute qualité et des coûts les plus bas dans un laps de temps extrêmement réduit.

Pour l’application développée, j’ai utilisé le modèle de cycle de vie en « V » qui est un modèle d'organisation des activités de développement d'un produit qui se caractérise par un flux d'activité descendant qui détaille le produit jusqu'à sa réalisation.

Figure 16: Modèle de cycle de vie en "V"

En outre, le cycle de vie d'un logiciel en V est composé de plusieurs étapes, chacune ayant une importance capitale pour garantir le succès du projet :

* Spécification des besoins : Cette étape consiste à comprendre les besoins du client et à établir les exigences du système.
* Conception : L'étape de conception comprend deux parties : la conception générale et la conception détaillée. La conception générale permet de définir l'architecture du système en fonction des besoins du client. Les spécifications techniques sont établies, et les choix technologiques sont faits. La conception détaillée quant à elle, permet de définir les détails de la conception générale.
* Développement : Cette étape consiste à écrire le code et à développer le système en utilisant les spécifications techniques.
* Intégration : Les différents modules sont intégrés pour créer le système complet.
* Qualification : Cette étape permet de vérifier que le système répond aux exigences du client et aux spécifications techniques. Il s'agit également de vérifier que le système est conforme aux normes de qualité, de performance et de sécurité.

Une fois le logiciel qualifié, il est prêt à être déployé chez le client. La maintenance du logiciel est assurée tout au long de son cycle de vie pour garantir son bon fonctionnement.

## **V.3. Les Tests**

### **V.3.1. Tests unitaires effectués**

Le test unitaire est un processus de vérification d'une unique unité de logiciel, alors j’ai testé la partie des formulaires par exemple vérifier si le champ de nombre ne peut contenir les lettres et la partie des boutons de validation (par exemple montrer le message de chargement après le clic sur le bouton).

### **V.3.2. Tests d’intégrations effectués**

Comme les tests d’intégration permettent de vérifier le fonctionnement de plusieurs unités de code au sein d'une configuration d'application, j’ai utilisé ce test pour vérifier les différents modules que j’ai réalisés par exemple module de gestion des membres et les liens avec la base de données, des fichiers ou des API en réseau qu’ils communiquent correctement.

### **V.3.3. Tests de validations effectués**

Comme le **test de validation** permet de vérifier si toutes les exigences client décrites dans le document de spécification d'un logiciel, écrit à partir de la spécification des besoins, sont respectées. J’ai vérifié si toutes les fonctionnalités écrites dans le cahier de charges sont bien fonctionnelles (Comme exemple, dans le cahier de charges il y a les demandes de faire les différentes fonctionnalités, donc il faut vérifier toutes les exigences sont respectés).

### **V.3.4. Sécurité du logiciel**

Comme la sécurité du logiciel est une notion importante, j’ai sécurisé l’application en s’attachant aux 5 aspects suivants :

* L’authentification

L’authentification consiste à savoir lier, grâce à un mot de passe par exemple une identité à une entité donnée d’un système (la partie administration d’un site, un processus en particulier ou 1 ordinateur par exemple). Chaque utilisateur a son propre mot de passe qui est crypté et son unique nom d’utilisateur pour connaitre celui qui a effectué l’opération avec le système.

* **Le contrôle d’accès**

Une fois authentifié, l’utilisateur souhaite accéder à des fonctionnalités offertes par l’application. Au préalable, il faut contrôler s’il a le droit d’y accéder.

Assurer cette fonction c’est avoir la capacité de lier une ressource (une base de données par exemple) avec des droits d’accès à cette ressource et une entité.

* L’intégrité

Il s’agit ici de prévenir l’action non volontaire ou accidentelle d’une donnée ou des services d’un système. (Par exemple empêcher la modification non autorisé ou involontaire des informations par des utilisateurs)

* **La confidentialité** des données

La confidentialité des données doit être assurée lors d’échange de **données sensibles** (mot de passe, nom d’utilisateur) Il s’agit de garantir que des données acquises illégalement soient inutilisables.

* La protection contre **l’analyse du trafic**.

La sécurité des communications repose sur des mécanismes déjà abordés et que nous approfondirons ensuite : mécanisme **d’authentification,** de **chiffrement** et de **hachage.** L’exemple le plus emblématique est l’utilisation des protocoles **SSL** (Secure Socket Layer) et **TLS** (Transport Layer Security) dans les échanges sur le Web grâce au protocole HTTPS quand l’application sera hébergée aux serveurs en ligne.

### **V.3.5. Manuel utilisateur du logiciel (les interfaces de l’application avec des explications)**

La figure 17 montre la page d’authentification que l’utilisateur (Coordinateur du CAPEML ou son secrétaire) utilise pour se connecter, il doit entrer son nom d’utilisateur et le mot de passe qui correspond aux identifiants enregistrés dans la base de données.

Apres avoir compléter ses identifiants il va ensuite cliquer sur le bouton se connecter.



Figure 17:Page d'authentification

La figure 18 montre qu’est ce qui se passe quand l’utilisateur entre les mauvais identifiants qui ne correspond pas à ceux de la base de données. Il va recevoir le message qui lui indique qu’il doit vérifier ses identifiants avant de continuer.

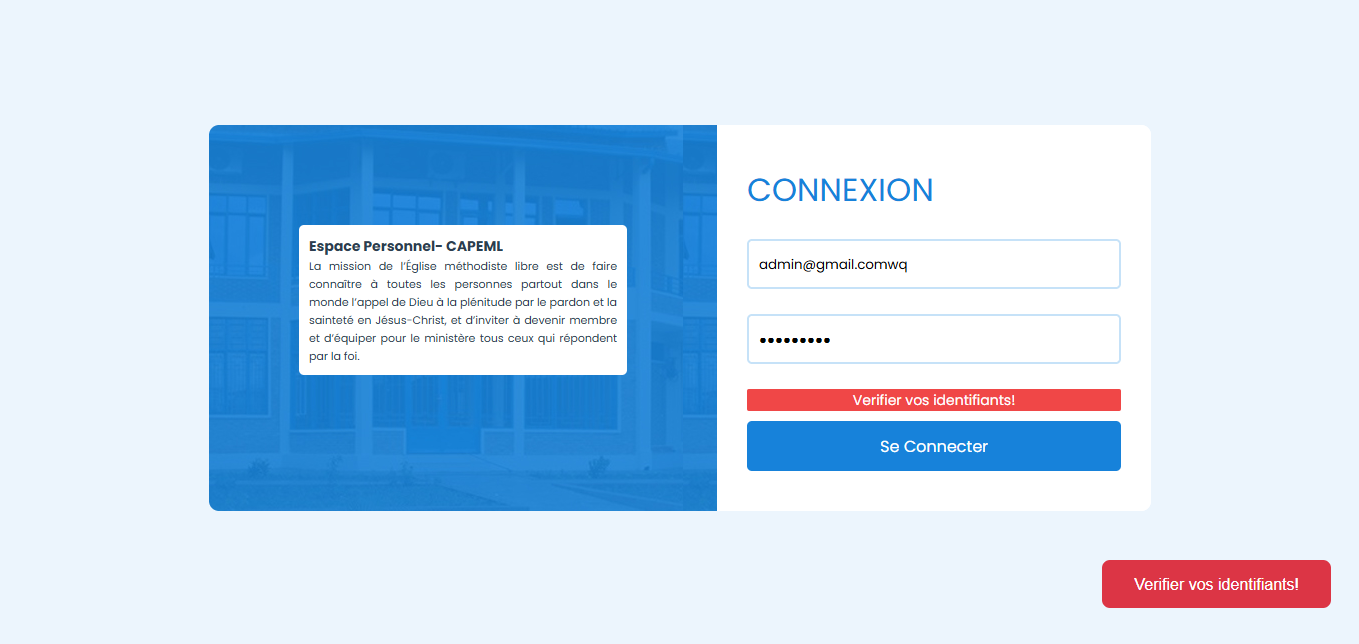


Figure 18:Présentation d'authentification échouée

S’il arrive à entrer les identifiants qui sont corrects, il va se rendre directement au tableau de bord de l’application.

Les figures 19,20 et 21 montrent les menus que l’application dispose ; les menus sont au nombre de sept avec trois sous menus sur le menu de l’église. A l’en tête de l’application il y a aussi le nom de l’application CAPEML avec aussi le nom de l’utilisateur connecté avec les sous menus sur le nom de l’utilisateur connecté.



Figure 19: Les menus de l'application

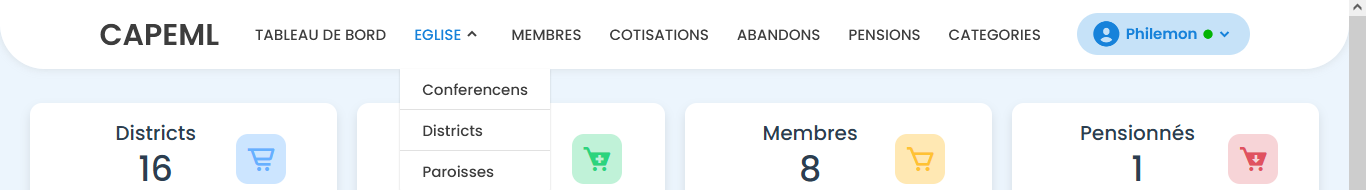


Figure 20: Les sous menus du menu "Eglise"

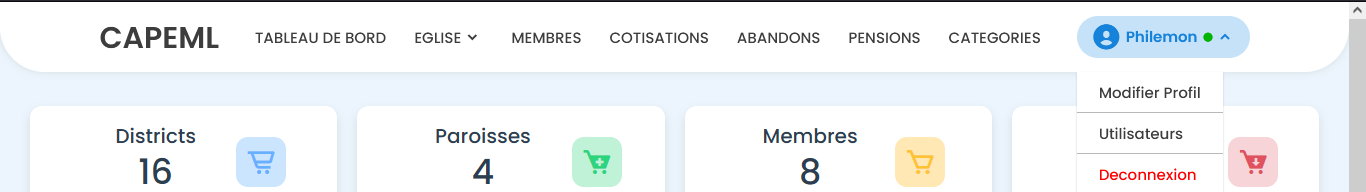


Figure 21: Les sous menus de l'utilisateur

La figure 22 montre le tableau de bord que le coordinateur du CAPEML voit s’il se connecte au système, sur le tableau de bord on y trouve quelques statistiques les plus utiles qui composent l’application comme le nombre total des districts que l’EMLBu dispose, le nombre total des paroisses disponibles, le nombre des membres qui sont en pension, le nombre total des membres qui ont abandonné à cotiser, le montant total que les membres ont déjà payés et le montant non payé.

Les cotisations récentes peuvent être vues par le coordinateur ou le secrétaire par ordre de date à laquelle elles sont enregistrées ; le coordinateur ou le secrétaire peut aussi voir le montant total que chaque district a déjà payé du premier au dernier.



Figure 22: Tableau de bord de l'application

La Figure 23 montre la page des districts, sur laquelle le coordinateur peut consulter la liste de tous les districts ainsi que leurs conférences respectives. Le coordinateur dispose également des options pour modifier ou supprimer un district en cliquant sur les boutons « Modifier » ou « Supprimer ». Il peut également imprimer la liste de districts d'une conférence ou de toutes les conférences simultanément en cliquant sur le bouton « Imprimer », situé au début du tableau.

De plus, le coordinateur a la possibilité de filtrer les districts en fonction de la conférence et de rechercher un district par nom sur cette page. Si le coordinateur souhaite ajouter un nouveau district, il peut cliquer sur le bouton « Ajouter », symbolisé par un signe plus (+), afin d'enregistrer les informations du nouveau district.

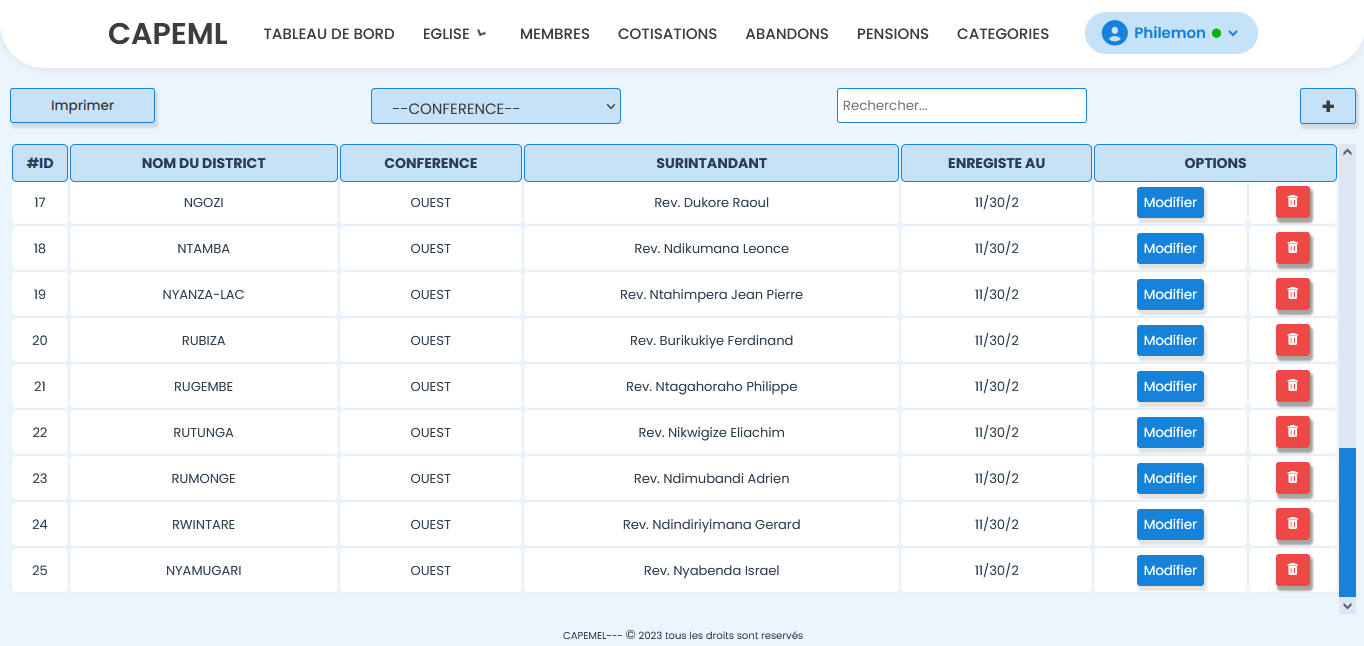


Figure 23: Page de liste des districts

La figure 24 montre la page où le coordinateur va utiliser pour ajouter un nouveau district il doit compléter correctement les champs nécessaires : le nom du district, le nom, l’email et le numéro de téléphone de surintendant de district et aussi sélectionné la conférence à laquelle le district appartient, après avoir compléter les champs il va cliquer sur le bouton « Enregistrer » pour valider et enregistrer les informations.

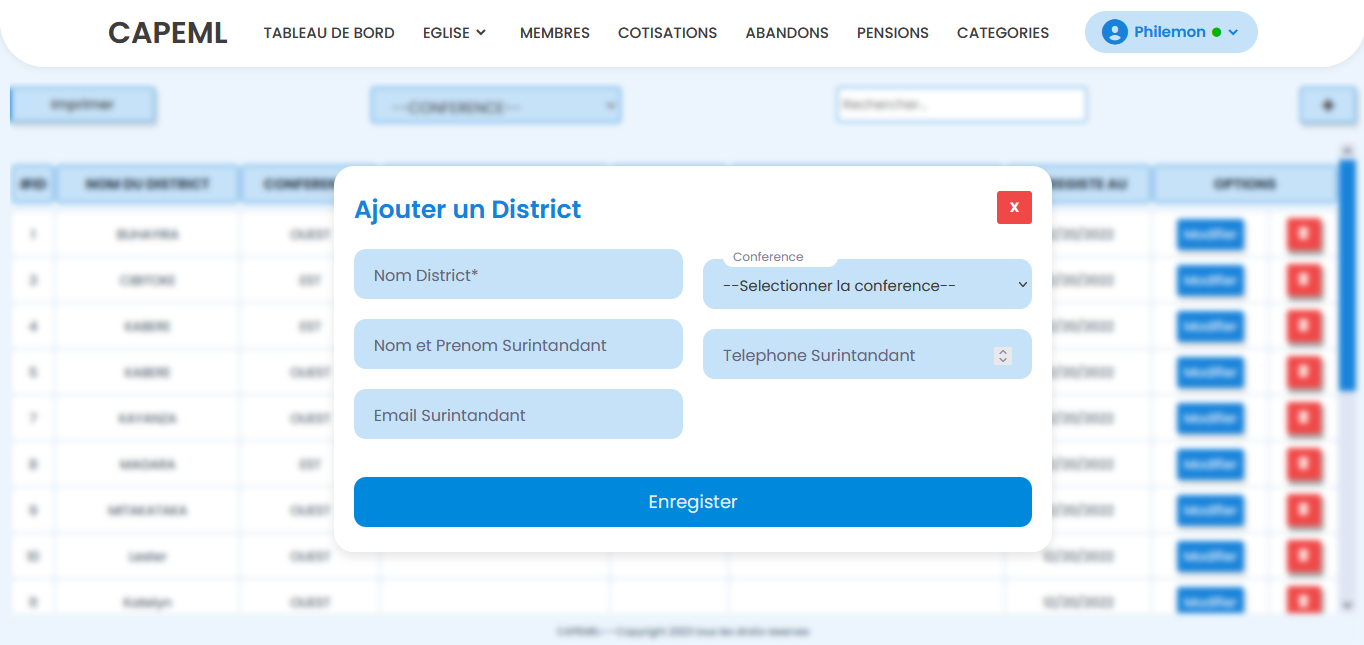


Figure 24: Page d'ajouter un nouveau district

Si le coordinateur complète les champs avec des informations qui ne sont pas permis ou qu’il n’a pas complété le champ qui est obligatoire, il va recevoir un message d’erreur de la validation qui apparait en couleur rouge à droite en bas de la page.

Au cas contraire il va recevoir un message d’enregistrement avec succès.

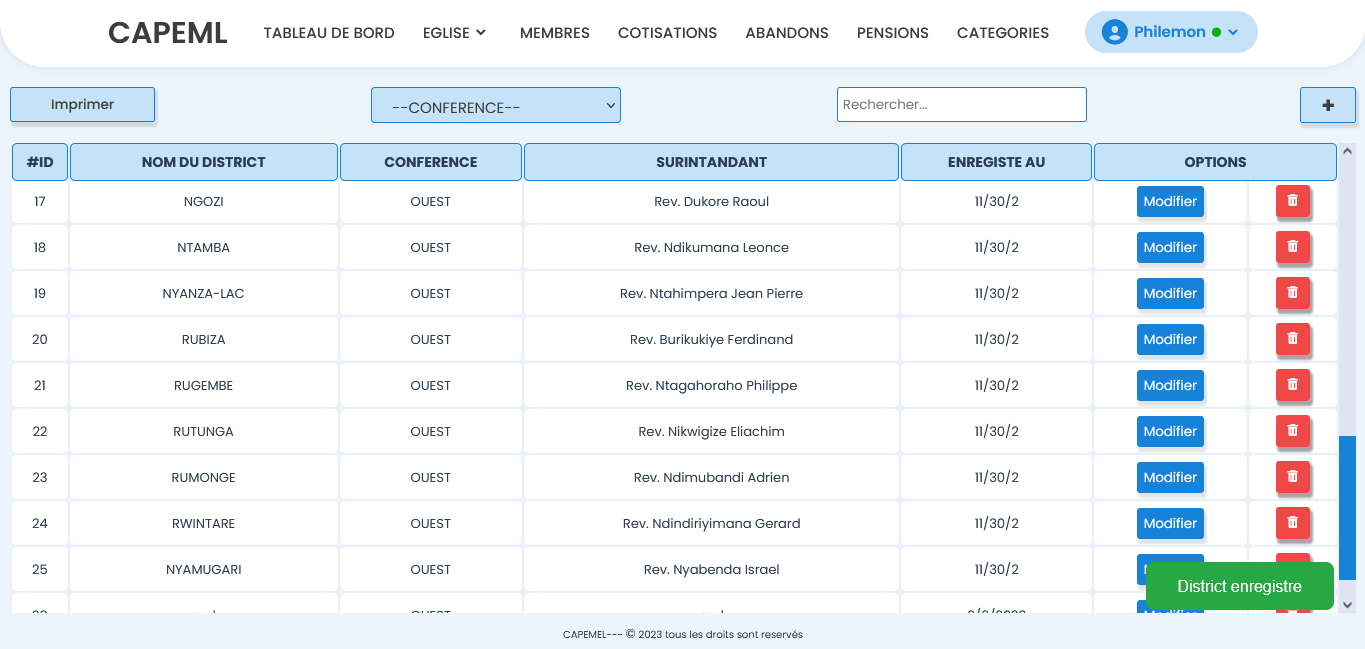


Figure 25:Liste des districts avec le message de succès après l'enregistrement

La figure 26 montre la fenêtre qui apparait après avoir cliqué sur le bouton « Modifier » pour modifier les informations du district, après avoir entré les informations il faut aussi valider la modification en cliquant sur le bouton « Modifier ».

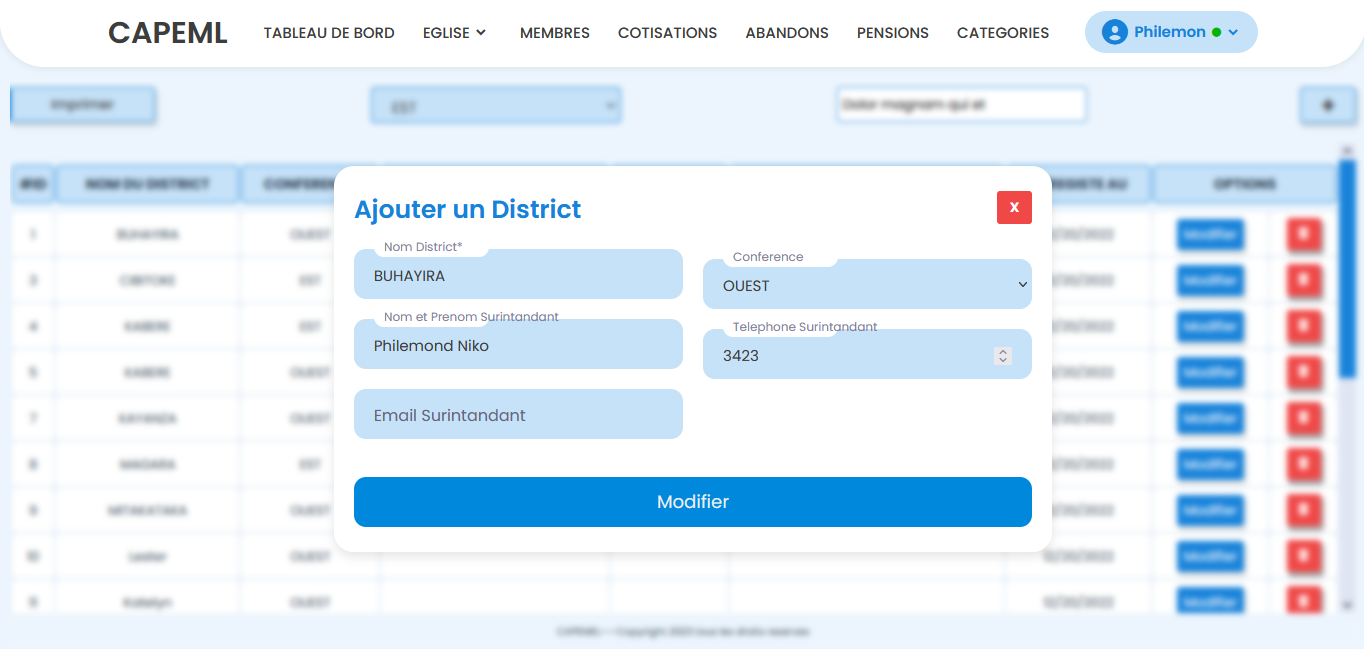


Figure 26: Page de modification d'un district

La figure 27 montre la fenêtre qui apparait après avoir cliqué sur le bouton « Supprimer » qui a l’icône d’une petite poubelle. C’est le message d’avertissement qui avertisse l’action de supprimer, si le coordinateur est d’accord de continuer l’activité de supprimer, il va valider en cliquant sur « Oui, supprimer », s’il ne veut pas continuer, il va cliquer sur « Annuler ».

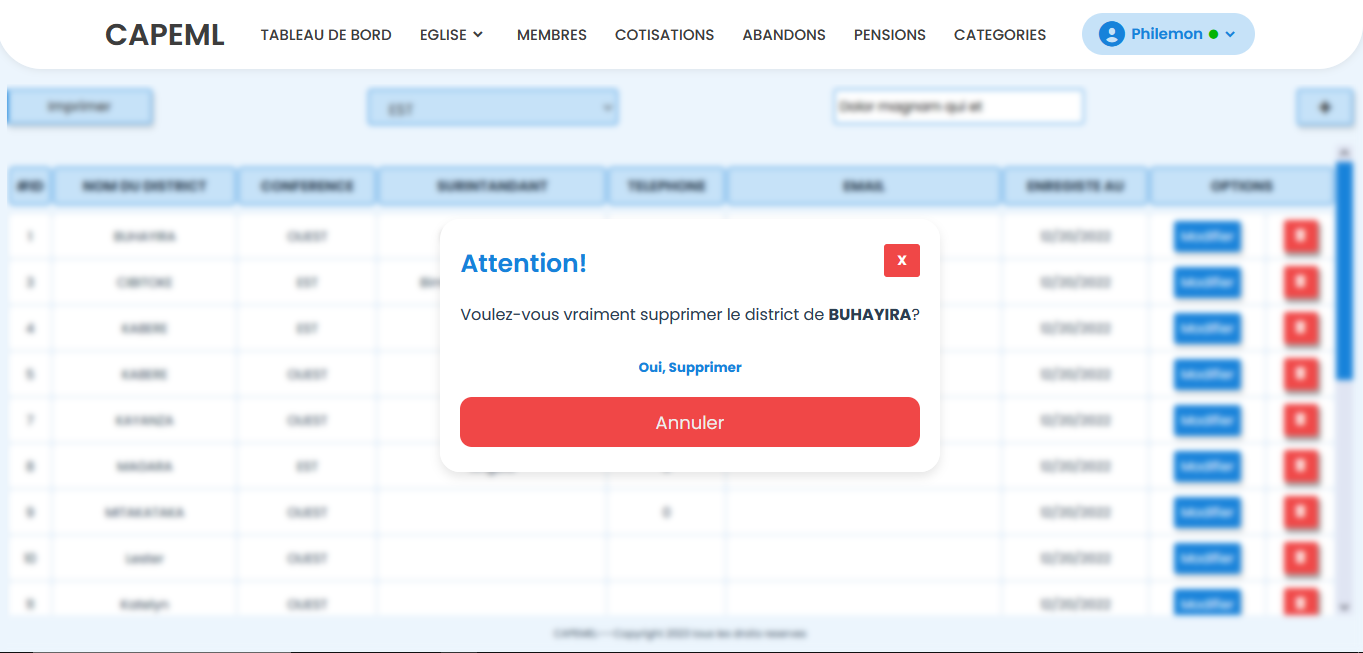


Figure 27: Message d'avertissement avant de supprimer un district

La figure 28 montre la page des paroisses, sur laquelle le coordinateur va voir la liste de toutes paroisses enregistrées ainsi que leurs districts respectives avec aussi les options de modifier ou supprimer une paroisse en cliquant sur le bouton « Modifier » pour modifier les informations de la paroisse ou « Supprimer » pour supprimer la paroisse, le coordinateur peut aussi imprimer la liste des paroisses de la conférence de son choix ou de toutes conférences en même temps en cliquant sur le bouton « imprimer » qui est au commencement du tableau.

Le coordinateur a aussi la possibilité de filtrer les paroisses qu’il veut afficher suivant la conférence et le district ; sur cette page aussi l’utilisateur a la possibilité de faire la recherche par le nom de la paroisse.

Si l’utilisateur veut ajouter une nouvelle paroisse, il va cliquer sur le bouton qui a un signe d’addition (+) qui signifie ajouter une nouvelle paroisse dans la base de données.

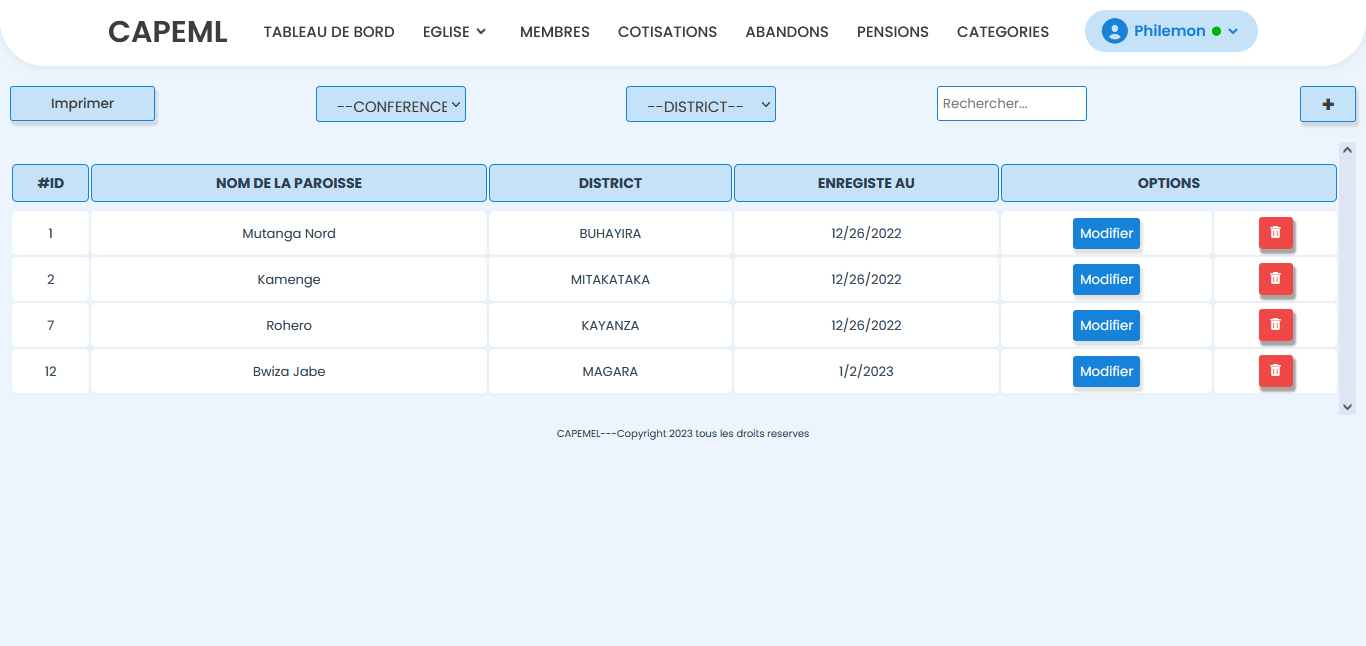


Figure 28: Page de liste des paroisses

La figure 29 montre la page sur laquelle le coordinateur va utiliser pour ajouter une nouvelle paroisse; il doit compléter correctement les champs nécessaires : le nom de la paroisse et sélectionner le district auquel la paroisse appartienne, après avoir compléter les champs il va cliquer sur le bouton « Enregistrer » pour valider et enregistrer les informations.

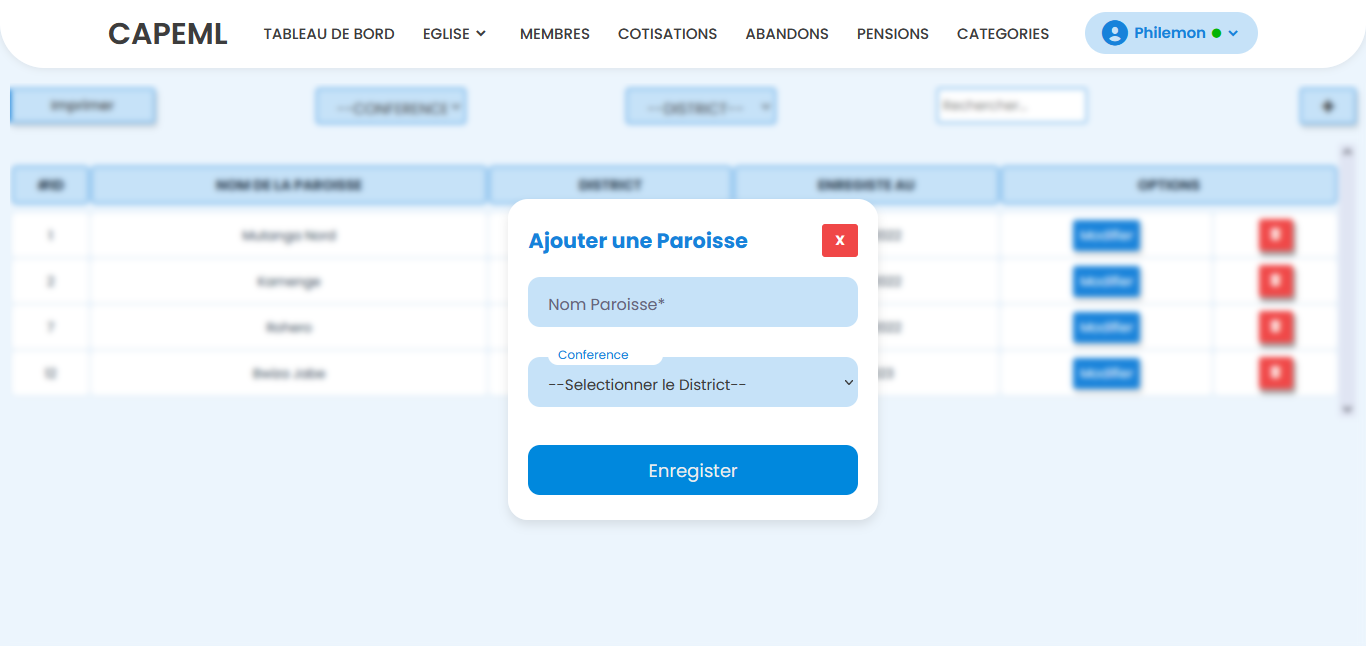


Figure 29:Page d'ajouter une nouvelle paroisse

La Figure 30 montre la page affichant la liste des membres de la CAPEML, accessible au coordinateur et au secrétaire. Cette page offre différentes fonctionnalités telles que la possibilité d'imprimer la liste des membres, d'effectuer une recherche avancée en filtrant les membres par district et paroisse, par date et par texte.

Sur cette page, la liste complète des membres est affichée avec leur numéro matricule, leur nom, leur prénom, leur district, leur paroisse, leur âge, leur catégorie, leur date de début de ministère et leur statut (actif, régulier, abandonné ou pensionné). Les options pour modifier les informations de chaque membre ou voir plus d'informations sont également disponibles.

La colonne des statuts permet de connaître le statut de chaque membre. Si le membre est actif et cotise régulièrement, son statut est affiché en vert. S'il est abandonné pour des raisons personnelles, son statut est affiché en jaune. S'il est déserteur, c'est-à-dire s'il n'est pas encore membre de l'EMLBu, son statut est affiché en mauve. Enfin, s'il est pensionné après avoir atteint l'âge de 65 ans et cotisé pendant 15 ans, son statut est affiché en rouge.

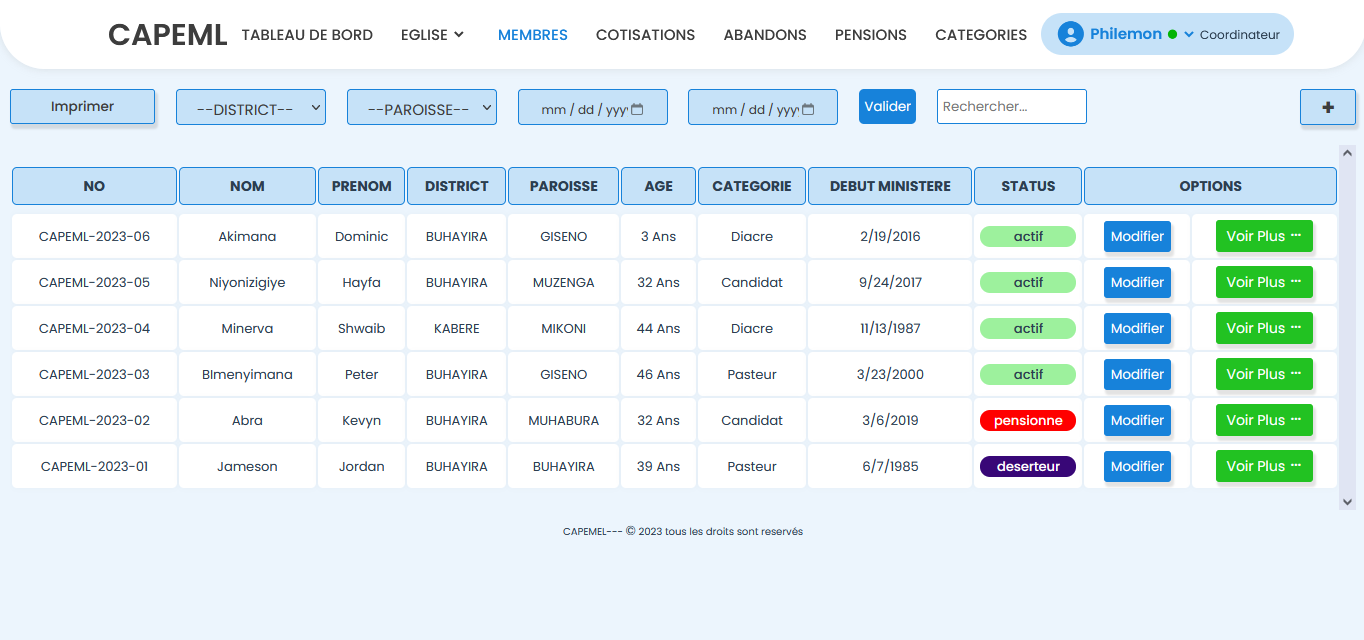


Figure 30:Liste des membres

La figure 31 montre la page sur laquelle le coordinateur ou le secrétaire va utiliser pour ajouter un nouveau membre, il doit compléter correctement les champs nécessaires : le nom, prénom, nom du père, nom du mère, date et lieu de naissance, son district et sa paroisse d’origine, sa catégorie ministérielle, le numéro de la CNI, date de mariage, son numéro de téléphone. Après avoir compléter les champs, il va cliquer sur le bouton « Enregistrer » pour valider et enregistrer les informations.

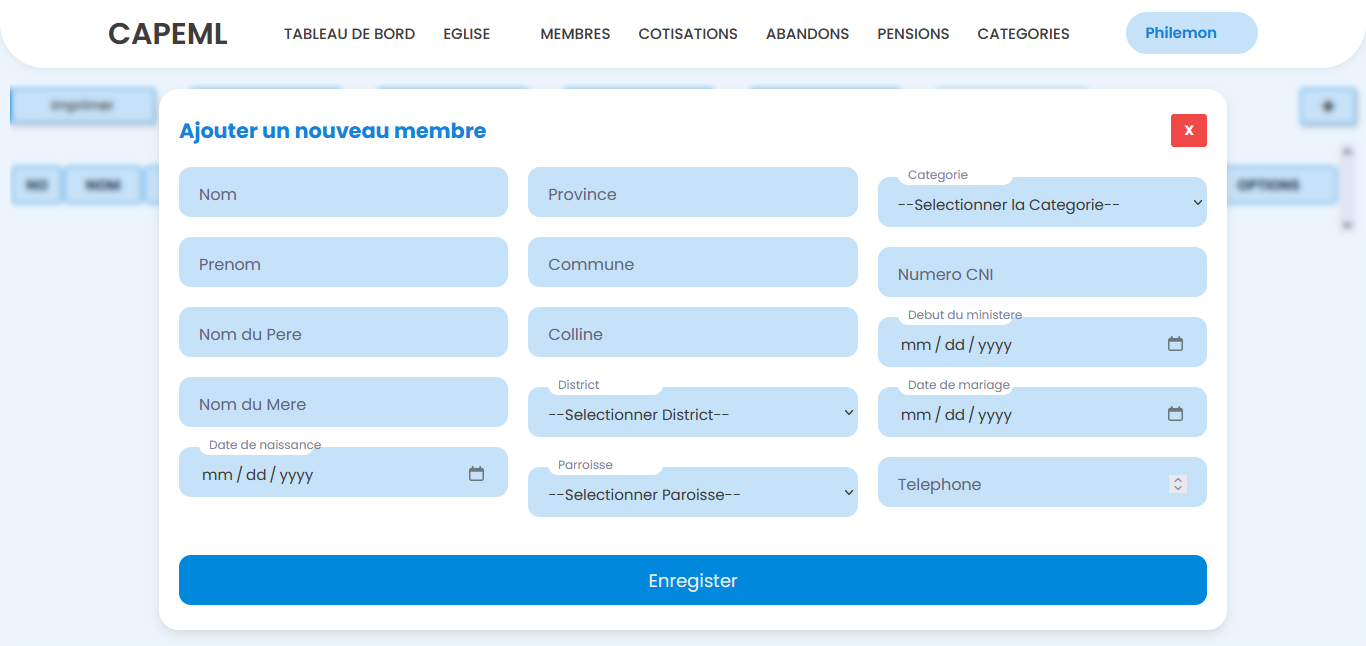


Figure 31:Formulaire d'ajouter un nouveau membre

Les figures 32, 33 et 34 montrent la fiche d’identification personnelle de chaque membre du CAPEML. Cette fiche est composée de trois parties distinctes : la première partie contient les informations du membre, la deuxième partie contient les informations du conjoint ou de la conjointe, et la troisième partie contient les informations des enfants du membre.

Dans la première partie de la fiche, on peut trouver des informations telles que le nom, le prénom, le nom du père et de la mère, le lieu et la date de naissance avec l'âge correspondant, ainsi que le statut du membre. Si le membre est abandonné, le montant qu'il recevra après l'abandon sera également indiqué en fonction des règles de partage des frais d’abandon de la CAPEML.

Dans la deuxième partie de la fiche, on peut trouver les informations du conjoint ou de la conjointe, telles que le nom, le prénom, le nom du père et de la mère, la date et le lieu de naissance, ainsi que le numéro de téléphone. Si le membre n'a pas de conjoint(e), seul le bouton "ajouter un(e) conjoint(e)" apparaîtra. Cependant, aucun membre n'a la possibilité d'avoir plus d'un(e) conjoint(e).

Dans la troisième partie de la fiche, on peut trouver les informations des enfants du membre, telles que leur nom, leur prénom et leur date de naissance. Si un enfant a plus de 21 ans, son nom apparaîtra en rouge.

En outre, sur cette page d'identification personnelle, les photos du membre et de son conjoint sont également affichées.

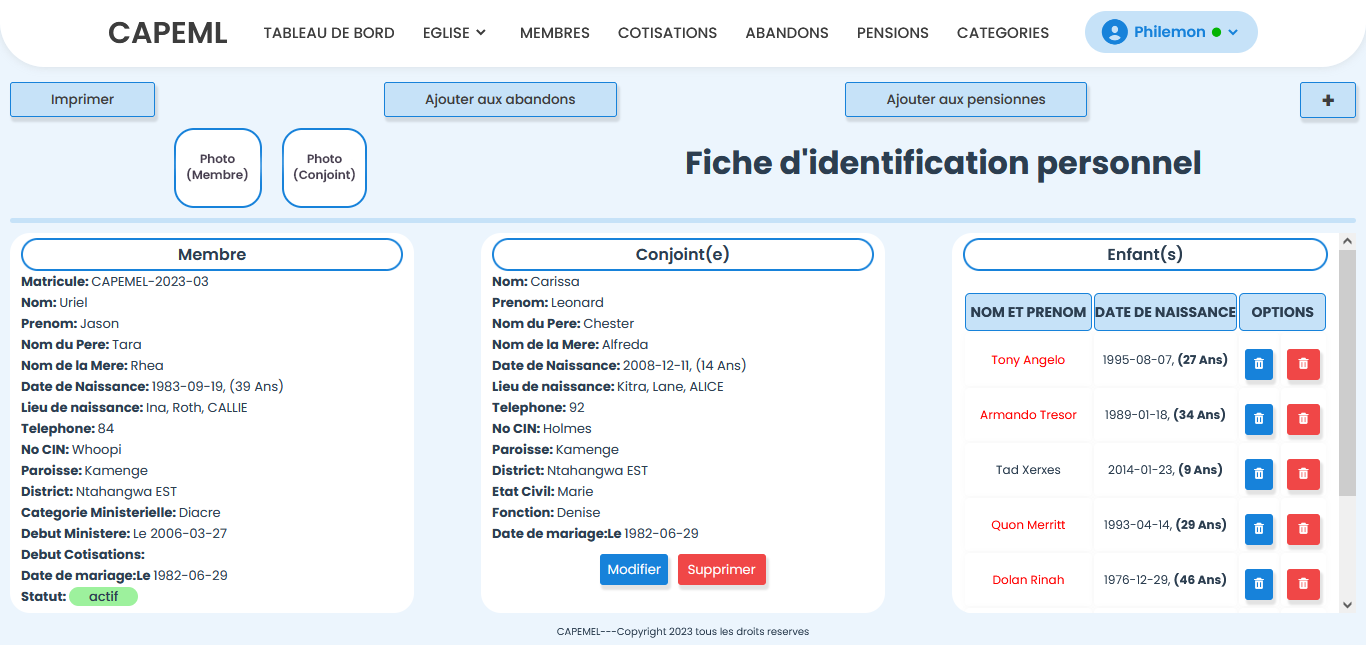


Figure 32:Fiche d'identification personnelle d’un membre actif

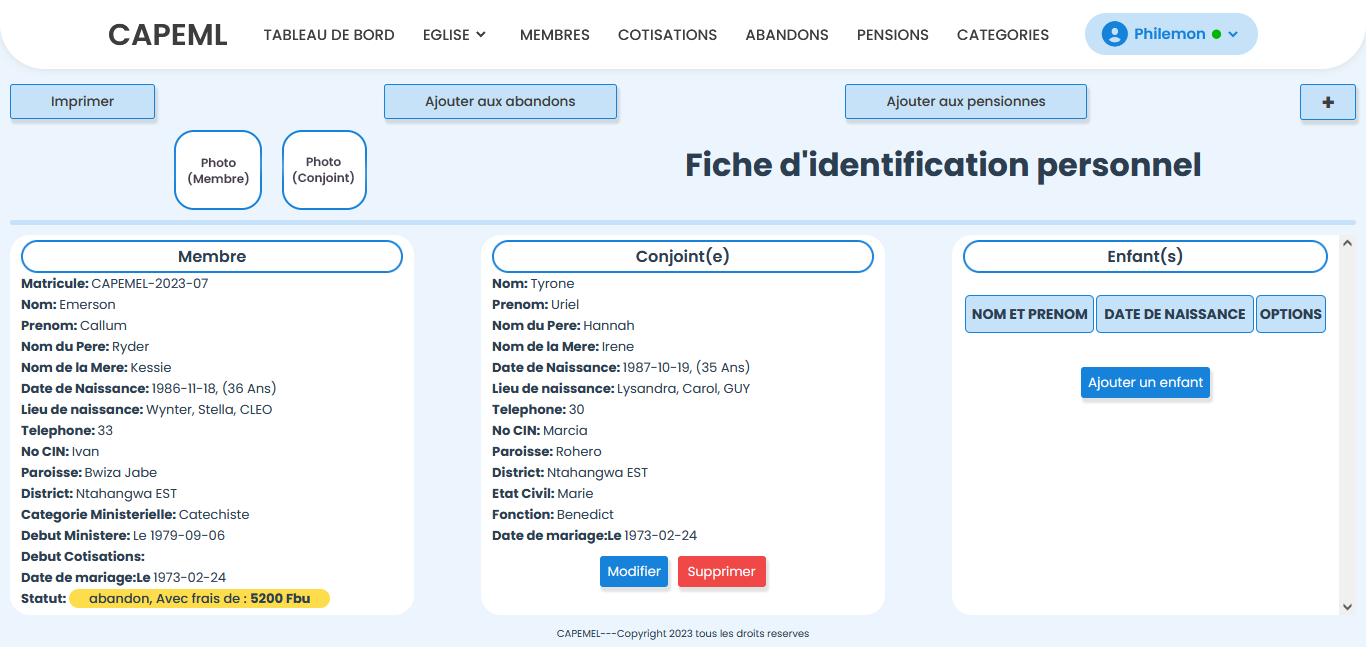


Figure 33:Fiche d'identification personnelle d'un membre abandonné

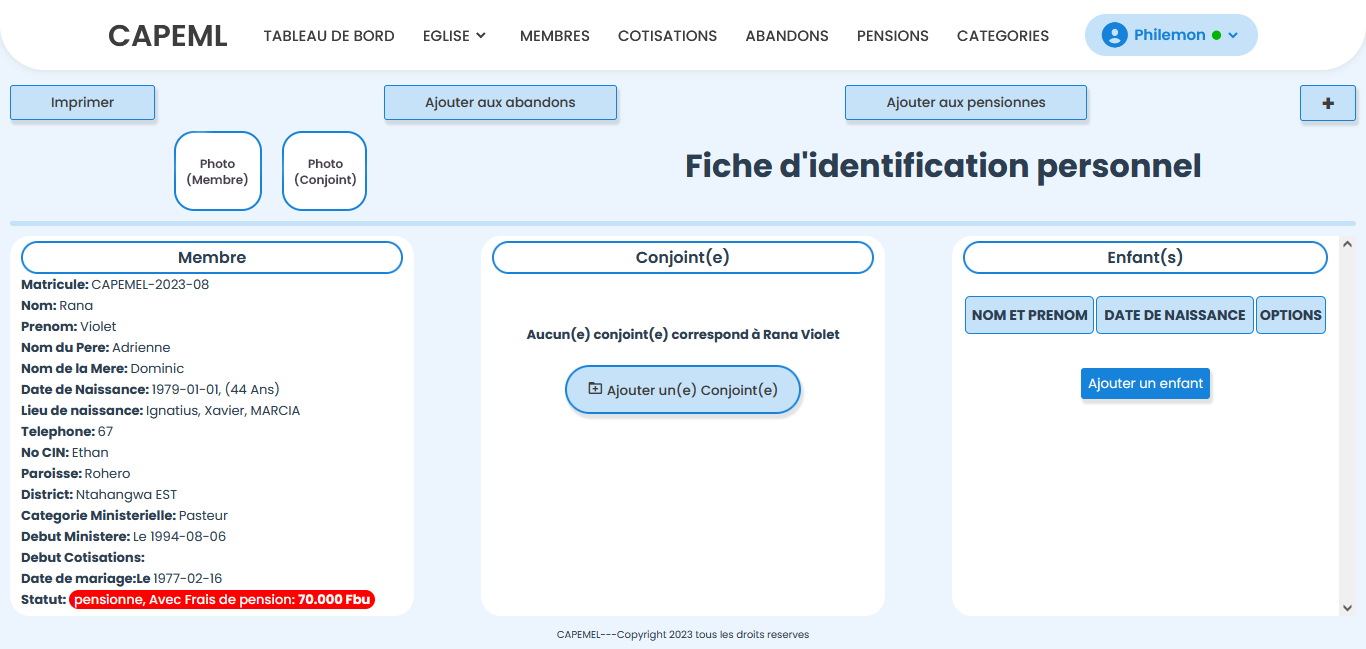


Figure 34:Fiche d'identification personnelle d’un membre pensionné

La figure 35 montre la fenêtre qui apparait si le coordinateur ou le secrétaire veut ajouter un membre aux abandons en sélectionnant le type d’abandon qu’il va attribuer à ce membre (Désertassions ou Abandon) et motif (mais le motif c’est un champ qui est facultatif).

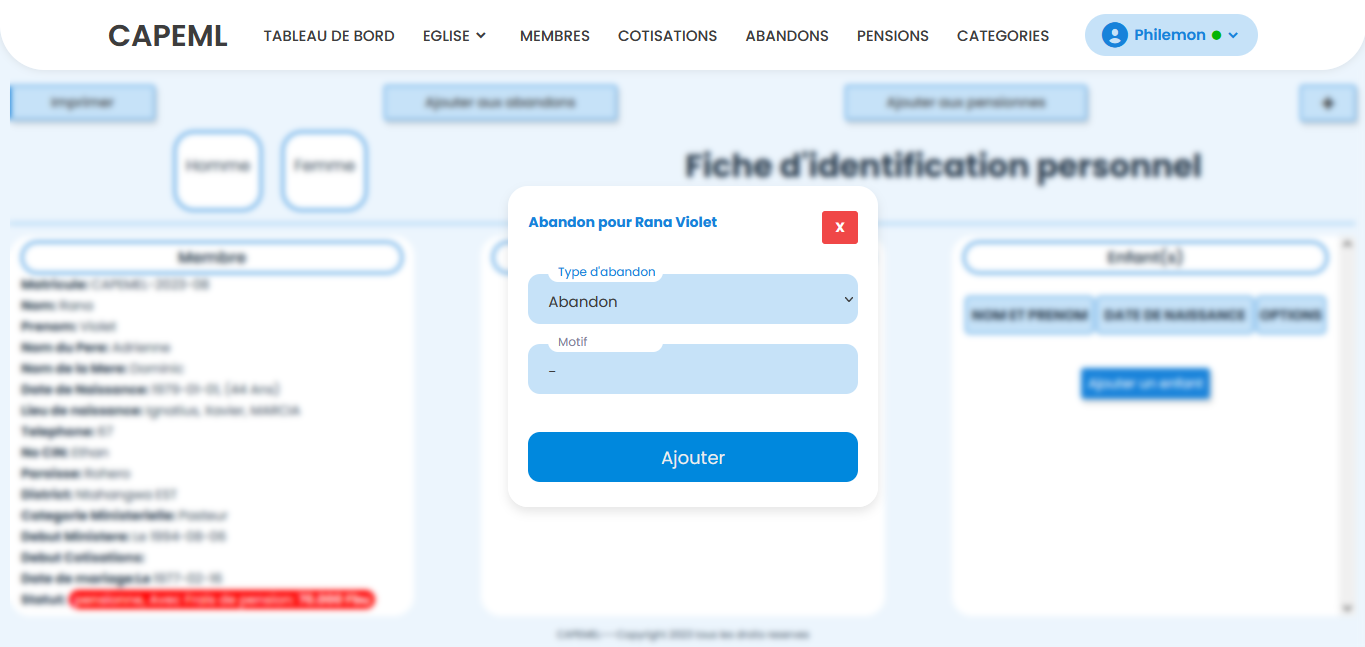


Figure 35:Page d'ajouter un abandon pour un membre

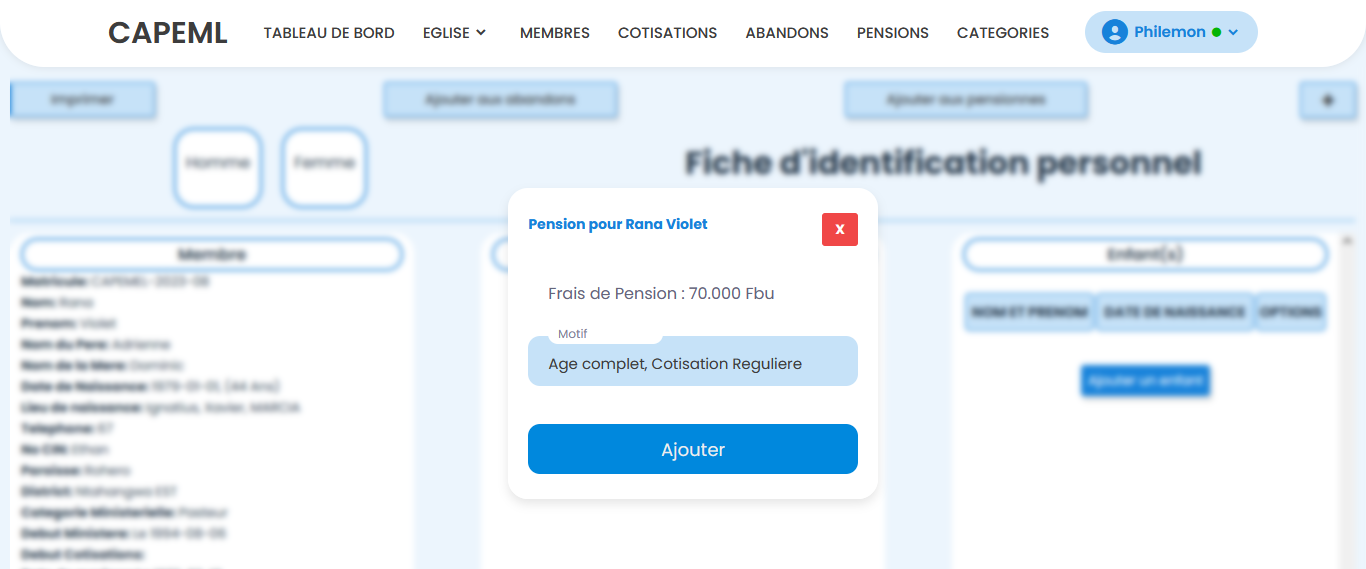
La figure 36 montre la fenêtre qui apparait si le coordinateur ou le secrétaire veut ajouter un membre aux pensionnés avec le motif (le champ qui est obligatoire), et les frais de pension vont s’afficher automatiquement suivant la catégorie du membre.

Figure 36: Page d'ajouter un membre aux pensionnés

La figure 37 montre la liste des membres pour la collecte de leurs cotisations trimestrielles. Tout d'abord, l'utilisateur doit sélectionner la conférence, puis le district, et tous les membres de ce district sélectionné apparaîtront. Seuls les membres actifs ont le droit de cotiser. Ensuite, l'utilisateur doit cliquer sur le bouton "payé" si le membre a payé la cotisation ou sur le bouton "non payé" si le membre n'a pas payé la cotisation. Avant de cliquer sur ces deux boutons, le coordinateur ou le secrétaire doit remplir les champs du trimestre, de l'année, du numéro du bordereau de paiement et de la date de paiement à la banque.

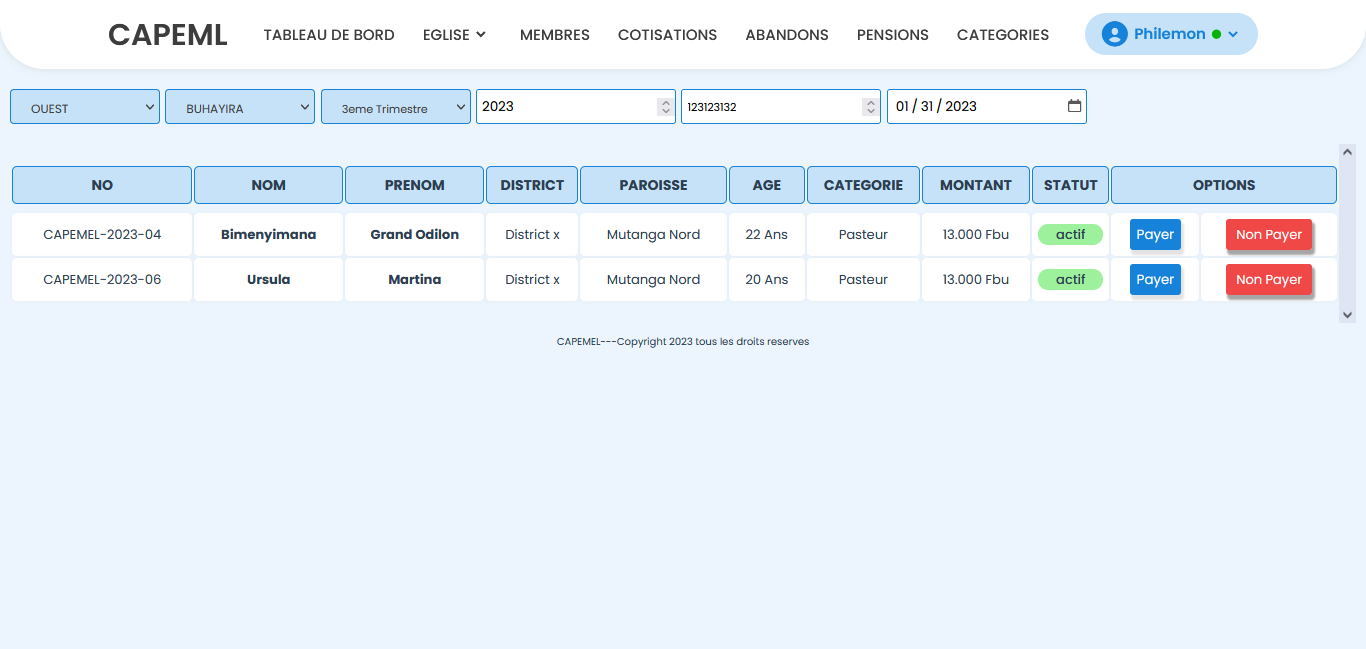


Figure 37: Page pour ajouter des cotisations aux membres

La figure 38 montre la page sur laquelle l’utilisateur (coordinateur ou secrétaire) va voir toutes les cotisations qui ont déjà payé et ceux qui ne sont pas encore payé. Sur cette page il peut voir les informations nécessaires sur les cotisations comme : le numéro de la cotisation, le matricule du membre, le nom du membre, le prénom du membre, le montant payé, le montant non payé, la période payée, numéro du bordereau, la date de paiement.

Sur cette page même, l’utilisateur a la possibilité d’imprimer le rapport par conférence, par district, par année, par trimestre ou bien rechercher par le nom du membre pour voir les cotisations du membre.

S’il veut ajouter une nouvelle cotisation, l’utilisateur va cliquer sur le bouton « ajouter cotisation » qui est en haut à droite de la page.

Sur cette page le coordinateur ou le secrétaire peut voir aussi les sommes totales des montants payés et non payés.

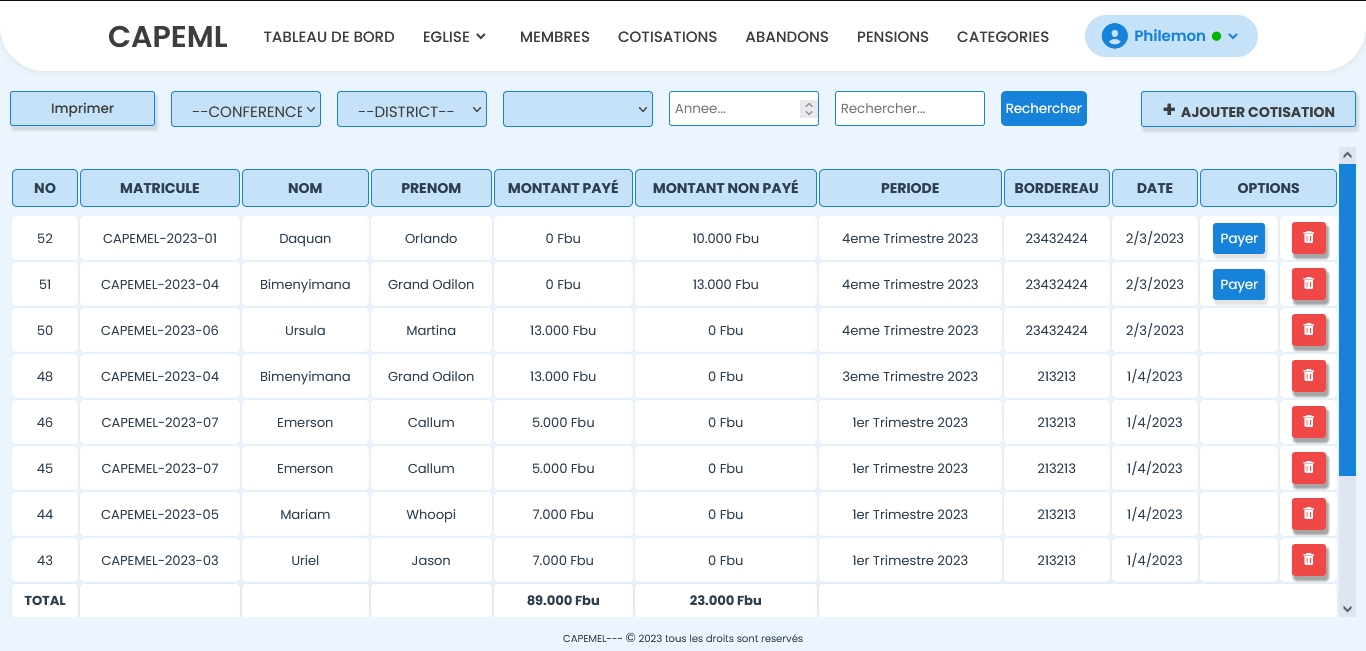


Figure 38: Liste des cotisations des membres

La figure 39 montre la page sur laquelle le coordinateur ou le secrétaire va voir la liste des membres qui ont abandonné à cotiser, l’utilisateur (le coordinateur ou le secrétaire) à la possibilité d’imprimer la liste des membres abandonnés en cliquant sur le bouton « imprimer ». Il peut aussi filtrer suivant le district et la paroisse des membres, suivant la date de création ; il a la possibilité de restaurer le membre dans les membres qui sont actifs en se basant sur les règles des abandons de la CAPEML, il peut voir aussi les informations supplémentaires du membre pour voir ses informations personnelles en cliquant sur « voir plus ».

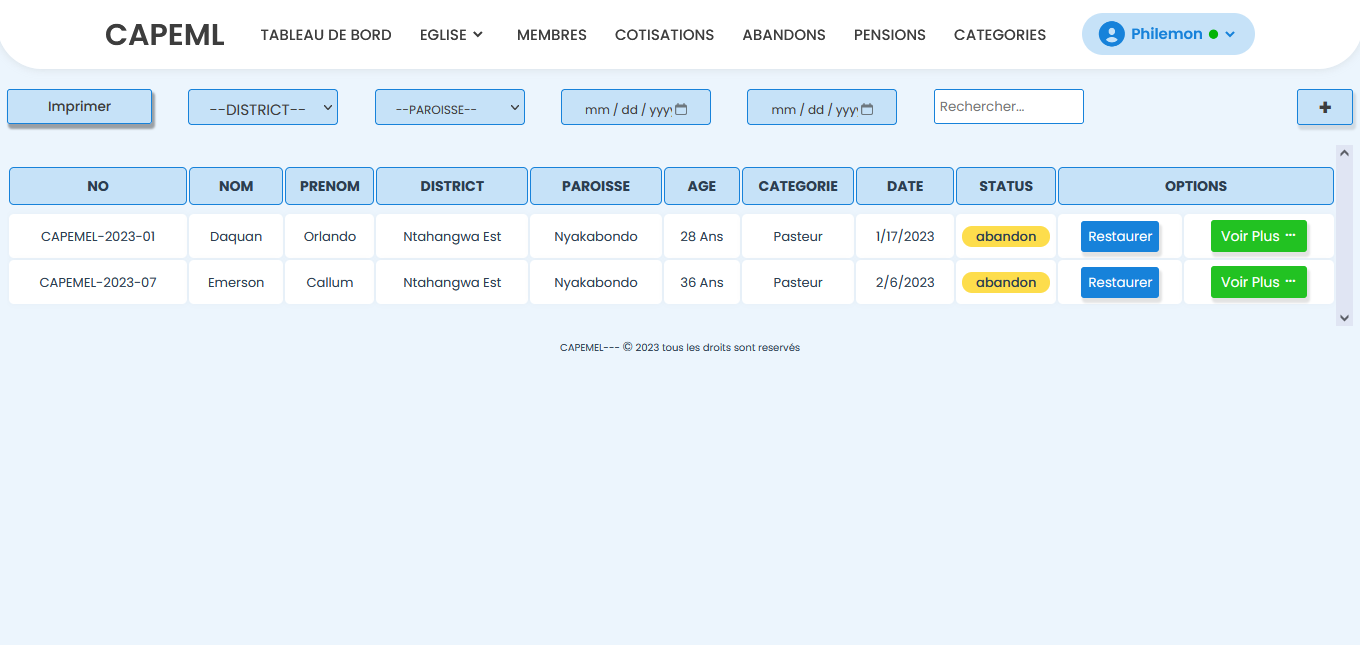


Figure 39: Liste des membres abandonnés

La figure 40 montre la page sur laquelle l’utilisateur (le coordinateur ou le secrétaire) va voir la liste des membres qui sont pensionnes, il a la possibilité d’imprimer la liste des membres pensionnes en cliquant sur le bouton « imprimer ». Il peut aussi filtrer suivant le district et la paroisse des membres, suivant la date de création ; il a la possibilité de restaurer le membre dans les membres qui sont actifs en se basant sur les règles des abandons de la CAPEML, il peut voir aussi les informations supplémentaires du membre pour voir ses informations personnelles en cliquant sur « voir plus ».

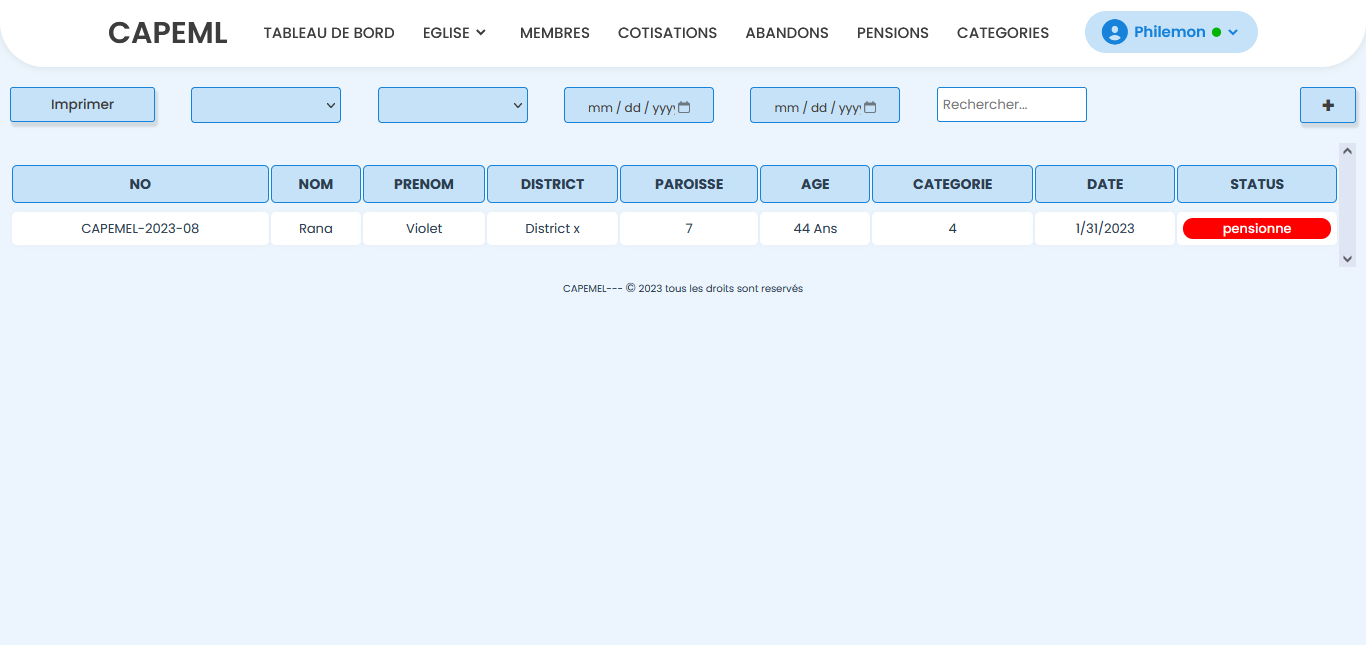


Figure 40: Liste des membres pensionnés

### **V.3.6. Cout estimatif du déploiement du logiciel**

Voici un exemple de devis simplifié de cout de déploiement de logiciel que j’ai proposé :

Tableau 2 : Devis simplifié de cout de déploiement

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Qté** | **P.U(en Fbu)** | **P.T (en Fbu)** |
|  |  |  |  |
| Le designer pour construire la maquette | 1 | 200,000 | 200,000 |
| La gestionnaire de la base de données | 1 | 450,000 | 450,000 |
| Les développeurs front-end | 2 | 550,000 | 1,100,000 |
| Les développeurs back-end | 3 | 650,000 | 1,950,000 |
| Les développeurs fullstack pour assembler  les modules | 2 | 700,000 | 1,400,000 |
| Les testeurs | 3 | 180,000 | 540,000 |
|  |  |  |  |
| Achat du nom de domaine (.org) | 1 | 50,000 | 50,000 |
| Achat du certificat SSL pour la sécurité | 1 | 52,500 | 52,500 |
| Hebergement/an | 1 | 235,000 | 235,000 |
| Maintenance | 1 | 120,000 | 120,000 |
| **TOTAL** |  |  | **6,097,500** |

Le coût du déploiementd'une application Web dépende de divers facteurs tels que la complexité, les fonctionnalités de l'application et même le pays de résidence du développeur.

La complexité de l'application : Si l'application est relativement simple, le coût de développement et de déploiement sera moindre. En revanche, si l'application est complexe et nécessite de nombreuses fonctionnalités, le coût sera plus élevé.

Les fonctionnalités de l'application : Si l'application doit intégrer des fonctionnalités avancées telles que des notifications ou une messagerie instantanée, le coût de développement et de déploiement sera plus élevé.

Le pays de résidence du développeur : Le coût de développement et de déploiement peut varier en fonction du pays où se trouve le développeur. Par exemple, si le développeur est basé au Burundi, le coût de développement et de déploiement sera probablement moins élevé qu'en Europe ou en Asie.

# **CHAPITRE VI: CONCLUSION GENERALE ET SUGGESTIONS**

## **VI.1. Conclusion générale**

### **VI.1.1 Appréciation critique du déroulement du stage**

Mon stage à l’Eglise Méthodiste Libre dans le département de développement générale était très merveilleux et aussi important pour moi. C’était difficile de comprendre comment la caisse pension est organisée et d’enregistrer les données écrites sur les papiers volants pour les mettre dans la machine ; c’était aussi un peu difficile pour moi de concevoir les interfaces utilisateurs car je devrais concevoir l’application facilement manipulable en tenant compte aux données disponibles.

### **VI.1.2. Appréciation critique du problème observé**

Je n’ai pas eu le problème sur les collectes des données parce que les données étaient vraiment claires sauf que ces dernières étaient sur les papiers et sur les fiches bien classées aussi.

## **VI.2. Suggestions**

Le département de la caisse pension a le coordinateur qui gère bien les données en les classant dans l’ordre, je suggère de continuer cette initiative pour rendre le travail plus rapide et précis.

Néanmoins, après avoir disponibilisé cet outil de gestion de la caisse de pension durant mon stage, je suggère fortement que le département soit doté d’une connexion internet avec une capacité suffisante pour lui faciliter de faire les recherches nécessaires liées à son travail.

Pour l’avenir de l’Eglise Méthodiste Libre au Burundi spécialement dans le département de la coordination de la caisse pension, je suggère d’héberger aux serveurs les services pour faciliter aux membres de gérer leurs cotisations à temps réel et partout où ils se trouvent.

# **REFERENCES**

[1] Archives de l’Eglise Methodiste Libre au Burundi, consulté le 24 Janvier 2023

[2] https://www.synonyme-du-mot.com/les-articles/pourquoi-modeliser-avec-uml, consulté le 9 Janvier 2023

[3] https://www.lucidchart.com/pages/fr/diagramme-de-cas-dutilisation-uml, consulté le 9 Janvier 2023

[4] https://creately.com/blog/fr/uncategorized-fr/tutoriel-sur-les-diagrammes-de-classe/, consulté le 9 Janvier 2023

[5] https://www.ibm.com/docs/fr/rsm/7.5.0?topic=uml-sequence-diagrams, consulté le 9 Janvier 2023

[6] https://www.ibm.com/docs/fr/rsar/9.5?topic=diagrams-activity, consulté le 9 Janvier 2023

[7] https://jobphoning.com/dictionnaire/php, Consulté le 24 Janvier 2023

[8] https://www.pappleweb.com/index/definition-de-laravel/, Consulté le 24 Janvier 2023

[9] https://www.journaldunet.fr/web-tech/dictionnaire-du-webmastering/1203585-javascript/, Consulté le 24 Janvier 2023

[10] https://www.ionos.fr/digitalguide/serveur/outils/tutoriel-xampp-creer-un-serveur-de-test-local/, consulté le 24 Janvier 2023