



PARTIE I: DOSSIER D'INSERTION





Résumé:

Le dossier d'insertion est le tout premier document rédigé durant la période du stage académique. Ici il est question pour nous de présenter la structure d'accueil et aussi le déroulement de la phase d'insertion.



I- PRESENTATION GENERALE DE LA CNPS

INTRODUCTION

- II- STRUCTURE ORGANISATIONNELLE DE LA CNPS
- III- ORGANIGRAMME DE LA CNPS

CONCLUSION





INTRODUCTION

Dans la plupart des écoles d'ingénieurs, l'obtention du Diplôme de Technicien Supérieur (DTS) présuppose l'exercice d'un stage d'application en milieu professionnel. Ce qui permet de familiariser le futur diplômé au monde du travail où il sera amené à exercer. C'est dans ce cadre qu'à travers l'IAI Cameroun, Centre d'Excellence Technologique Paul BIYA, nous avons été retenus au sein de la Direction des Systèmes d'Information (DSI) de la Caisse Nationale de Prévoyance Sociale (CNPS), pour un stage académique de trois (03) mois, allant du 01 Juin au 31 Aout 2017. Le présent dossier fait donc office de rapport d'insertion présentant le fonctionnement global de la structure, ainsi que la description matérielle et logicielle de celle-ci.





I. PRESENTATION GENERALE DE LA CNPS

1. Naissance de la CNPS

Avec l'Indépendance et la Réunification, le Cameroun, membre de l'Organisation Internationale du Travail depuis 1960, est amené à adapter sa législation aux normes internationales. Cet ajustement du cadre juridique de la protection sociale s'est fait à travers : La loi N°67-LF-07 du 12 juin 1967 instituant un code de prestations familiales ; la loi N°67-LF-08 du 12 juin 1967 qui crée la Caisse Nationale de Prévoyance Sociale en tant qu'organisme autonome en charge de la gestion du régime des prestations sociales ; la loi N°69-LF-18 du 10 novembre 1969 instituant un régime d'assurance pensions de vieillesse, d'invalidité et de décès ; l'ordonnance N°73-17 du 22 mai 1973 portant organisation de la prévoyance sociale qui confie à la CNPS, dans le cadre de la politique générale du gouvernement, le service des diverses prestations prévues par la législation de protection sociale et la loi N°77-11 du 13 juillet 1977 portant réparation et prévention des accidents du travail et des maladies professionnelles qui confie à la CNPS, la couverture et la gestion des risques professionnels abrogeant ainsi une législation antérieure, issue de l'ordonnance 59-100 du 31 décembre 1959.

2. Les Missions

La Caisse nationale de Prévoyance sociale a pour mission d'assurer dans le cadre de la politique générale du Gouvernement, le service de diverses prestations prévues par la législation de protection sociale et familiale.

A ce titre, elle couvre trois branches de sécurité sociale dont :

- Les Prestations familiales (PF)
- Les Pensions vieillesse, d'invalidité et de décès (PVID)
- Les Risques professionnels (RP)

Le système de sécurité sociale Camerounais est basé sur la solidarité ; ceux qui travaillent cotisent pour ceux admis à faire valoir leurs droits à la retraite. De même, il convient de rappeler que la principale source de financement de la CNPS reste les cotisations sociales que les employeurs reversent mensuellement.

Par ailleurs, dans un environnement socio-économique incertain, où l'organisme doit assurer ses obligations vis-à-vis des générations futures, l'astuce a consisté à entreprendre des investissements économiquement et financièrement porteurs. L'immobilier est l'option choisie. D'où l'acquisition en janvier 2010 de l'immeuble CAMAIR à Paris, la réhabilitation de l'immeuble ministériel n°1 à Yaoundé etc., qui participent de cette politique d'investissements orientée vers les secteurs productifs et présentant les garanties de sécurité et de pérennité financière.

L'organisme investit également dans l'amélioration constante de l'offre des soins de santé, dans le cadre de son action sanitaire et sociale. C'est ainsi que la rénovation profonde des plateaux techniques des structures hospitalières a été possible, par l'acquisition d'équipements à la pointe de la technologie nouvelle.





En fait, plusieurs innovations ont été réalisées dans le cadre de la consolidation des réserves pour garantir le financement à long terme des prestations sociales, tout en équilibrant la gestion de l'organisme, comme le préconisent les standards internationaux en matière de sécurité sociale.

La CNPS assure, depuis le 03 novembre 2014, la protection sociale des travailleurs indépendants, qui peuvent s'immatriculer dans le cadre de l'assurance volontaire. Ce processus d'extension de la couverture sociale permet de s'immatriculer dans l'un des 37 centres de prévoyance sociale de l'organisme à titre volontaire dès l'âge de 14 ans, sous réserve de l'accord parental évidemment.

A la CNPS, le souci est de soutenir le présent, tout en confortant l'avenir des assurés sociaux, nos véritables patrons.

3. Accueil et intégration

La réussite en milieu professionnel ou en entreprise s'appuie sur une bonne intégration. A cet effet, nous avons commencé notre stage le lundi 06 juillet 2015, et en ce même jour, nous avons entamé notre période d'insertion qui a durée 02 semaines. Aussi progressive qu'elle a été, elle s'est articulée autour des points suivants :

- Une rencontre avec la Direction de Ressources Humaines (DRH), qui nous a conduites au sein de notre direction hôte à savoir la Direction des Systèmes d'Information (DSI);
- ➢ Ici, un bref entretien nous a été accordé par le Chef de secrétariat du DSI, qui nous a aussitôt assigné au Service des Etudes et du Développement (SED) avec pour encadreur professionnel M. NDJENG NKOLO Éric.
- La description générale de la CNPS nous a été faite par ce dernier, ainsi que la présentation du service d'accueil et toute l'équipe de travail.
- Nous avons ainsi pris une période d'observation pendant laquelle nous avons étudié le milieu et les interactions entre le personnel du centre de prévoyance sociale indépendance de Yaoundé en vue de desceller les insuffisances que nous aurons à corriger par la suite. Au terme de cette période, nous avons reçu pour thème « Automatisation du processus de délivrance des attestations d'affiliation et de versement des cotisations sociales »

II. STRUCTURE ORGANISATIONNELLE DE LA CNPS

1. Organisation générale

C'est une organisation dynamique dans le temps et dans l'espace. La CNPS comprend ainsi une tutelle, un conseil d'administration, une direction générale, des directions centrales, des cellules spécialisées et des structures déconcentrées.

a. La tutelle Il s'agit du Ministère Du Travail et de la Sécurité Sociale





b. Le conseil d'administration

Il est créé auprès du Président du Conseil d'administration, un secrétariat particulier chargé notamment :

- De la préparation matérielle des réunions du secrétariat des organes délibérant de la Caisse Nationale de Prévoyance Sociale;
- De l'établissement des procès-verbaux des réunions du conseil d'administration en liaison avec les Secrétariats Particuliers du Directeur General et du Directeur General Adjoint;
- Des relations avec les membres u conseil d'administration et des relations publiques du Président du Conseil d'Administration ;

c. La Direction Générale:

Elle est assurée par un Directeur General et un Directeur General Adjoint, tous nommés par décret Présidentiel ;

d. L'Agence Comptable (AG):

Elle est placée sous l'autorité d'un agent comptable et est chargée de :

- Du maniement des fonds ;
- De la conservation des dernières valeurs ;
- Du recouvrement des titres obligatoires ;
- De la liquidation de tous les titres de paiement ;
- De la confection du compte de gestion ;
- De la centralisation des opérations comptables ;
- De la tenue des statistiques comptables ;
- De la tenue de la comptabilité analytique ;

e. L'Inspection Générale (IG) :

Placée sous l'autorité d'un Inspecteur General ayant rang de Directeur, l'Inspection Générale est chargée :

- De la définition et de la production des indicateurs de performance ;
- De l'exploitation de tout rapport émanant des services centraux et extérieurs ;
- De la production du rapport d'activités du Directeur General ;
- De l'élaboration des règles et manuels de procédures, ainsi que des méthodes de travail;
- De l'inspection pédagogique ;
- De tous travaux que lui confie le Directeur General;

f. La Direction des Audits et du Contrôle Interne (DACI) :

Placée sous l'autorité d'un Directeur, éventuellement assisté d'un adjoint, elle est chargée de la mise en œuvre de l'audit des différents services et notamment :

- Le suivi et le contrôle de l'exécution du budget ;
- L'exécution de toute mission d'audit, d'inspection et de contrôle ;
- Le contrôle et l'inspection de l'archivage des actes administratifs et des pièces justificatives des dépenses et des recettes ;
- L'évaluation des performances des services ;





- L'élaboration des rapports d'audit.
- g. La Direction des Finances et du Patrimoine (DFP) :

Placée sous l'autorité d'un Directeur et d'un adjoint chargé spécialement des Affaires Financières et de la Comptabilité Matière, elle est chargée :

- De l'ordonnancement des recettes et des dépenses ; de l'étude et du suivi des placements et prises de participation ;
- Du suivi de toutes les opérations de recettes et des dépenses ;
- De l'élaboration du budget, et du suivi de son exécution ;
- Des délégations des crédits ;
- Du suivi de la gestion des investissements immobiliers ;
- De la réception des commandes, matériels et fournitures de bureau ;
- De la conservation du contrôle physique des stocks.
- h. La Direction des Ressources Humaines (DRH) :

Placée sous l'autorité d'un Directeur et éventuellement d'un adjoint, elle est chargée :

- De l'administration et de la gestion des personnels ;
- De l'élaboration et de l'exécution du programme de formation et de recyclage du personnel ;
- De l'élaboration et du suivi du plan de carrières des personnels ;
- De la gestion de la paie des personnels ;
- De l'évaluation du rendement et des performances des personnels ;
- De l'organisation des missions et voyages.
- i. La Direction des Affaires Générale (DAG) :

Placée sous l'autorité d'un Directeur, éventuellement assisté d'un adjoint, elle est chargée :

- De l'acquisition et de la gestion des biens meubles ;
- De la gestion du parc automobile ;
- De la gestion de matériels et fournitures nécessaires au fonctionnement des services;
- De la gestion et de l'entretien du patrimoine mobilier ;
- De la gestion des immeubles abritant les services ;
- Des engagements budgétaires ;
- Des archives.
- j. La Direction des Prestations (DP) :

Elle est chargée de :

- La coordination du paiement des prestations servies dans le cadre des branches de prévoyance sociale ;
- Du contrôle du paiement des prestations ;
- De la liquidation des prestations services aux ressortissants de pays étranges liés au Cameroun par des conventions de sécurité sociale ;
- De la liquidation des prestations servies aux agents de la CNPS dans la branche des risques professionnels;





- Du suivi de tout contentieux technique résultant sociales, en rapport avec les structures du contentieux crées auprès des centres régionaux de prévoyance sociale :
- Des études relatives à la législation et à la règlementation sociale.
- k. La Direction des Systèmes d'Information (DSI) :

Elle est chargée de la mise en place et de la sécurisation des applications informatiques, bureautiques et des télécommunications.

I. La Direction du Recouvrement (DR) :

Elle est chargée :

- De l'immatriculation des employeurs et des assurés sociaux ;
- De la tenue des fichiers employeur et assurés ;
- Des appels à cotisation ;
- De la tenue des comptes cotisants et des comptes individuels assurés.

m. La Direction des Affaires Juridiques et du Contentieux (DAJC) :

Elle est chargée :

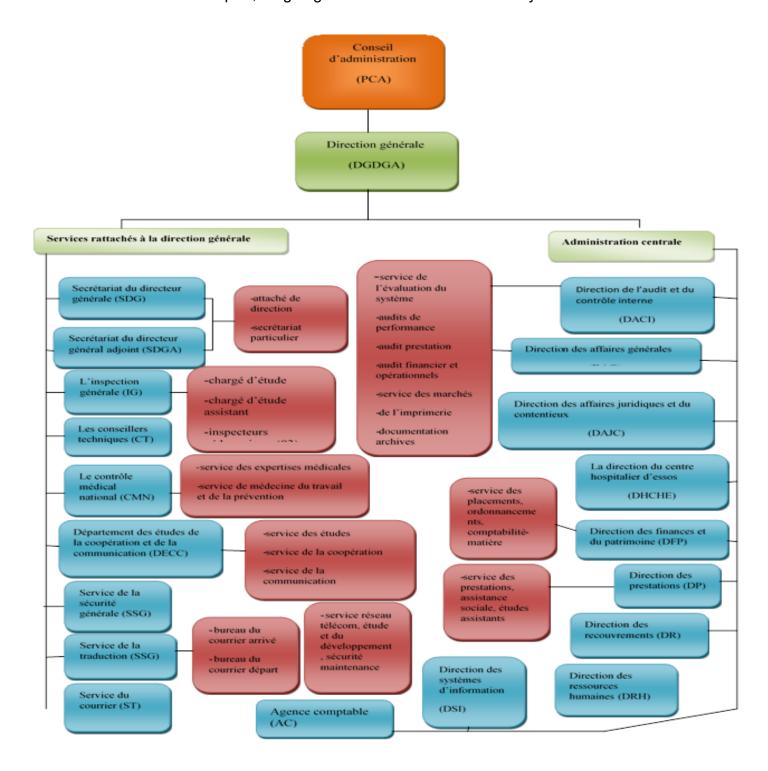
- Du contrôle de la régularité des actes pris dans le cadre de la gestion technique de la CNPS;
- De la mise en forme des projets de textes à soumettre à la sanction du Directeur Général, du Président du Conseil d'Administration et du Gouvernement, en liaison avec les Directions techniques intéressées;
- Du contrôle des régularités actes et de leur conformité à la législation et la réglementation en vigueur ;
- De l'organisation et du suivi de la défense des intérêts de la CNPS ;
- Du suivi de toute affaire contentieuse et juridique confiée par le Directeur Général;
- n. La Direction du Centre Hospitalier d'Essos (DCHE)
- o. Les Directions Régionales
- p. Les centres de prévoyance sociale de première catégorie
- q. Le centre de prévoyance sociale de deuxième catégorie
- r. Le centre de prévoyance sociale de troisième catégorie
- s. Les guichets périodiques.





III. ORGANIGRAMME DE LA CNPS

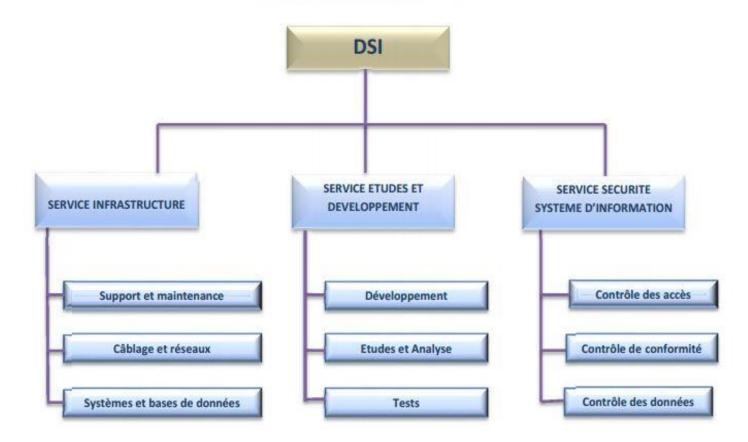
D'une manière globale, nous présenterons d'une part l'organigramme général de la CNPS et d'autre part, l'organigramme de la Direction des Systèmes d'Information.







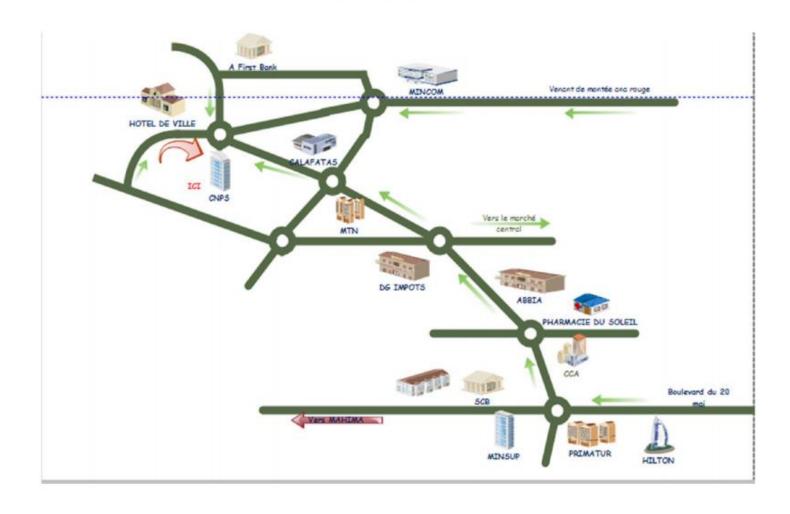
Organigramme de la DSI







Situation géographique de la CNPS







Les ressources de la CNPS

Selon ses différents services et divisions, la Caisse Nationale de Prévoyance Sociale regorge d'un grand nombre de ressources aussi bien matérielles, logicielles et humaines. Dans le cadre de notre étude, nous nous intéresserons à celles de la DSI. Ainsi, nous pouvons les consigner dans le tableau suivant :

Tableau 1: Les ressources de la CNPS

Ressources matérielles	Ressources logicielles	Ressources humaines
• Ordinateurs HP,	 systèmes 	des Directeurs
DELL	d'exploitation : • des chefs service	
• Imprimantes Laser	Windows XP, Windows 7	• des chefs services
Jet	Professional; Windows 8	adjoints
• Switch	 antivirus 	 chargés d'étude
• Routeurs	adobe Reader	 des agents
• Onduleurs	• office 2007, 2010,	
• Serveurs	2013	
• bais	messagerie interne :	
	Sappeli mails	





CONCLUSION

Au terme des deux semaines marquant notre période d'insertion, il en ressort que la Caisse Nationale de Prévoyance Sociale accueille des étudiants de plusieurs écoles de formation dans le cadre des stages académiques. Dans notre cas, l'accueil a été convivial. Il a été question pour nous à l'issus de cette observation de présenter l'aspect géographique, historique et organisationnel de la CNPS. Parvenus donc au terme de la phase d'insertion, il nous revient donc d'amorcer la phase d'analyse. Cette analyse portera sur le thème : Automatisation du processus de délivrance des attestations d'affiliation et versement des cotisations sociales.





PARTIE II: CAHIER DE CHARGES





Résumé:

Le cahier de charges est un document qui présente les conditions rattachées à l'exécution d'un projet. Il décrit précisément les besoins auxquels les intervenants doivent répondre (objectifs, cibles, concurrence, spécifications techniques, délais etc.). Cette partie permet à l'étudiant de prendre connaissance de son thème qui est dans notre cas « Automatisation du processus de délivrance des attestations d'affiliation et versement des cotisations sociales. »



- I- CONTEXTE ET JUSTIFICATION DE L'ETUDE
- II- OBJECTIF GENERAL
- III- OBJECTIFS SPECIFIQUES
- IV- BESOINS FONCTIONNELS
- V- BESOINS NON FONCTIONNELS
- VI- LES INTERVENANTS DU PROJET
- VII- RESSOURCES MATERIELS, LOGICIELLES ET HUMAINES
- VIII- ESTIMATION DU COUT DU PROJET
- IX- PLANIFICATION DU PROJET
- X- LES CONTRAINTES
- XI- LES LIVRABLES

CONCLUSION





INTRODUCTION

Dans le but de mener à bien un projet informatique, il est important que les intervenants respectent scrupuleusement les clauses définies dans le cahier des charges qui est un document préalable à la réalisation d'une application. Il facilite tout d'abord la définition et la compréhension du système étudié en ressortant les grandes lignes de son fonctionnement, ensuite présente les différents besoins émis par le client ou l'utilisateur et enfin dresse les contraintes liées au futur système. Il est établi par le maitre d'œuvre en accord avec le maitre d'ouvrage. Dans le cadre de notre étude, il sera question d'automatiser le processus de délivrance des attestations d'affiliation et de versement des cotisations sociale au sein de la CNPS, c'est ainsi que dans l'ossature de ce document, nous retrouverons le contexte, la justification et les objectifs de l'étude ainsi que les contraintes humaines, matérielles, temporelles et financières liées au projet.





I. Contexte et justification de l'étude

Dans le but de mener à bien sa mission qui est d'assurer la couverture sociale des employés au Cameroun, la CNPS procède à la collecte des cotisations sociales auprès des différents employeurs ; c'est dans cette optique qu'a été mis sur pieds le document appelé attestation d'affiliation et de versement des cotisations sociales (AAVCS), qui a pour but de certifier qu'un employé est ou a été affilié à la CNPS et de ressortir les périodes sur lesquelles son ou ses employeurs ont effectivement cotisés pour lui.

Le thème soumis à notre étude à savoir « Automatisation du processus de délivrance des attestations d'affiliation et de versement des cotisations sociales » a pour essence la gestion efficace et améliorée du processus du processus de délivrance des AAVCS. En effet cette opération vise à rendre automatique les traitements qui interviennent tout au long de ce processus.

II. Objectif général

Notre étude a pour but de mettre sur pieds une solution qui automatisera le processus de délivrances des attestations d'affiliation et de versement des cotisations sociales.

III. Objectifs spécifiques

Les objectifs spécifiques de notre étude sont :

- La gestion des dépôts de dossiers en ligne
- L'établissement automatisé des AAVCS
- L'informatisation du transfert des AAVCS entre le TSS et le GDC
- La gestion des assurés
- La production des statistiques
- La gestion des comptes

IV. Besoins fonctionnels

Les besoins fonctionnels ici représentent l'ensemble des principaux modules demandés par les utilisateurs. Ils ont purement trait à ce qu'est sensé faire l'application à la fin de sa réalisation. Dans notre cas, le futur système devra permettre :

Aux usagers de :





- > Créer un compte assuré
- > Déposer son dossier d'AAVCS en ligne
- Consulter l'évolution de son dossier

Aux techniciens de la sécurité sociale de :

- Recevoir les dossiers d'AAVCS
- Valider ou rejeter un dossier
- Etablir l'AAVCS
- Transmettre les AAVCS au GDC

Aux gestionnaires de compte de :

- Recevoir l'AAVCS
- Imprimer l'AAVCS

Aux gestionnaires d'assuré d':

- Ajouter un assuré
- Ajouter un emploi
- Enregistrer une cotisation
- V. Besoins non fonctionnels

A la suite des besoins fonctionnels, viennent les besoins non fonctionnels qui représentent la convivialité de l'application. Ainsi donc les besoins non fonctionnels retenus dans le cadre de notre projet sont :

- Une ergonomie simple et digeste
- L'application devra être multiplateforme
- L'application devra être extensible

VI. Les intervenants du projet

Noms et prénoms	Fonctions	Rôles
M. NDJENG NKOLO Éric	Chef du service d'étude et développement de la CNPS	Encadreur professionnel
M. Assawoga Nicolas	Enseignant à l'IAI Cameroun	Encadreur académique
M. NGOLLE DIN Charles	Etudiant à l'IAI Cameroun	Maitre d'œuvre

VII. Ressources matérielles, logicielles et humaines

Pour mener à bien notre projet, nous auront besoin de :





Ressources matérielles :

Les ressources matérielles sont :

- ✓ Laptop (dual-core 2.4 2.4GHz, RAM 4Go, disque dur 320Go)
- ✓ Imprimante
- ✓ Clé USB
- ✓ Onduleur

Ressources logicielles

Les ressources logicielles sont :

- ✓ Windows 8
- ✓ Wamp server
- ✓ Macromedia Dreamweaver
- ✓ Bi defender
- ✓ Navigateur (Mozilla Firefox, opéra mini, Google chrome)
- ✓ Power AMC
- ✓ Microsoft office 2016
- ✓ Netbeans
- ✓ Sublime text

Ressources humaines

- ✓ Analyste concepteur
- ✓ Programmeur
- ✓ Testeur

VIII. Estimation du cout du projet

Fiche d'évaluation du projet			
Matériel	Quantité	Prix	
Laptop	1	300.000	
Imprimante	1	65.000	
Clé USB	3	15.000	
Onduleur	1	85.000	
Total		465.000	
Logiciels	Quantité	Prix	
Windows 8	1	190.000	
Wamp server	1	Gratuit	
Macromedia	1	300000	
Dreamweaver			
Bi defender	1	99.000	
Navigateur	3	Gratuit	
Power AMC	1	4643950	
Microsoft office 2016	1	300.000	
Netbeans 8.1	1	Gratuit	

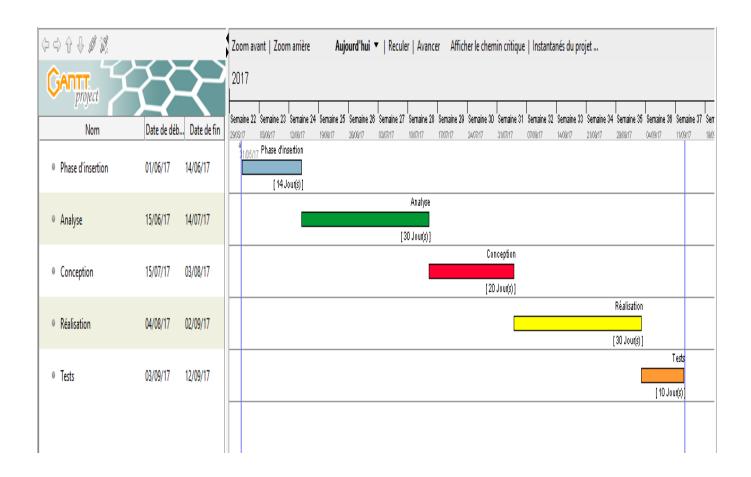




Sublime text	1		Gratuit	
Total	otal		5.532.950	
Main d'œuvre	Prix par semaine	Nombre de semaine	Montant	
Analyste concepteur			1.200.000	
Programmeur			750.000	
Testeur	200.000	1	200.000	
Total		2.150.000		
Marge d'erreur				
(logiciels + matériels + main d'œuvre) * 10% = 814.795				
Cout total du projet		7.333.155		

IX. Planification du projet

Le projet sera réalisé selon l'organisation temporelle suivante :







X. Contraintes

Les contraintes auxquelles sont soumis notre projet sont les suivantes :

- Contrainte de temps : l'application devra être livrée dans un délai de quatre mois
- Contrainte de qualité : dans le cadre de ce projet, nous allons utiliser le langage de modélisation UML1.9, le langage de programmation Java EE et nous utiliserons une architecture trois tiers
- Contrainte de cout : bien que le développement de cette application se fait dans le cadre d'un stage académique, il mobilise des ressources matérielles, logicielles et humaine, d'où une évaluation totale du cout du projet sera importante

XI. Les livrables

Les livrables de notre projet sont les suivants :

- ➤ CD-ROM de l'application
- Le dossier d'insertion
- ➤ Le dossier d'analyse
- Le dossier de conception
- Le dossier de réalisation
- ➤ Le guide d'installation et d'utilisation





CONCLUSION

Le présent cahier de charges nous a permis d'évaluer les différents aspects du projet. Aussi nous avons pu ressortir les attentes du commanditaire en ce qui concerne « Automatisation du processus de délivrance des attestations d'affiliation et versement des cotisations sociales, destiné à faciliter la déclaration des employés. Il en ressort que les différentes informations recensées permettront une meilleure compréhension du sujet et une approche plus objective de la phase d'analyse.





PARTIE III: DOSSIER D'ANALYSE





Résumé:

Le dossier d'analyse a pour but d'affiner le choix de la méthode à utiliser tout au long du projet. Il permet dans la même lancée de faire comme son nom l'indique une analyse détaillée du projet à mettre en œuvre.



- I- ETUDE DE L'EXISTANT
- II- METHODOLOGIE
- III- CAPTURE DES BESOINS FONCTIONNELS

CONCLUSION





INTRODUCTION

Après la rédaction du cahier de charges et la spécification des besoins, nous aborderons l'étape suivante qui consiste à analyser notre existant et de concevoir le système en se basant sur le langage UML. L'objectif de cette étape est de déterminer de façon détaillée et précise ce que le système devra faire, afin de répondre aux objectifs établis dans le cahier de charges tout en respectant les contraintes de coûts et délais. La réussite d'un projet repose sur cette phase. Si une erreur s'y glisse bien que le langage utilisé soit itératif et incrémental, elle aura des répercussions néfastes sur le futur système. C'est pour cette raison que nous avons mis l'accent sur la conception minutieuse de cette partie jugée délicate et déterminante ; afin de minimiser les risques, d'éviter les erreurs et de réaliser une solution répondant aux normes des systèmes d'information actuelles. Par conséquent, ce dossier d'analyse sera structuré autour des diagrammes de cas d'utilisation et séquences.





I- ETUDE DE L'EXISTANT

1- Recueil de l'existant

La caisse nationale de prévoyance sociale est implantée sur l'ensemble du territoire et subdiviser en centre de prévoyance social (CPS) et chaque CPS a à sa tête un Chef de centre.

Pour être un assuré social à la CNPS, il faut au préalable être un employé dans une entreprise ou une structure affiliée à la CNPS et en suite s'affilier sois même en tant que salarié de cette structure. Pour s'affilier, l'employé se rend dans un CPS menu de sa carte d'identité et de ses bulletins de paye mensuel, là-bas il rencontre un gestionnaire d'assuré (GDA) qui après avoir vérifié si sa structure est affiliée à la CNPS, va l'ajouté dans le système en tant qu'assuré social en enregistrant son nom, son adresse et son numéro de téléphone.

Tous les mois l'employeur effectue des cotisations sociales pour chacun de ses employés et cela tout au long de leurs contrats respectifs. Pour cela il se rend dans un CPS où il rencontre un GDA qui va procéder à l'enregistrement des cotisations des différents employés en insérant dans le système le nom de l'employeur, le nom de l'assuré et la date de la cotisation.

Cependant il est possible qu'un assuré décide de changer de structure et donc d'employeur à un moment donné pour diverses raisons, pour continuer à être un assuré social à la CNPS, il faudrait que cette nouvelle structure soit aussi affiliée à la CNPS et il devra s'affilier à nouveau en tant que salarié de cette nouvelle structure et ce processus est renouvelable à volonté. Pour s'affilier avec une nouvelle structure il aura besoin d'un document qui atteste qu'il est déjà un assuré de la CNPS et qu'il est à jour au niveau de ses cotisations. Ce document est l'attestation d'affiliation et de versement des cotisations sociales et il est délivré par la CNPS.

Pour entrer en possession de ce document, l'usager doit se rendre dans un CPS. Lorsque l'usager arrive au centre, il doit passer par plusieurs étapes avant l'établissement de son AAVCS, et à chaque étape correspond un agent CNPS. Ces étapes sont :

- Le passage à la guérite :
 - Arrivé au centre, l'usager est identifié à la guérite par le vigile qui consigne ses données d'identification (nom, prénom, numéro de CNI) dans un registre.
- La réception et l'orientation au niveau du poste avancé :
 Une fois à l'intérieur du CPS, l'usager est reçu au niveau du poste avancé où l'agent responsable lui remet un ticket (sur lequel est marqué la lettre R et un numéro compris entre 1 et 80) qui indique l'ordre de passage de l'usager dans le bureau total.





- L'établissement de l'AAVCS dans le bureau total :

Arrivé au bureau total, l'usager est reçu par un technicien de la sécurité sociale qui vérifie la complétude de son dossier et consulte l'état de l'assuré dans le système; en effet, le technicien de la sécurité sociale effectue une fouille dans le système, il recherche les différents emplois qu'a eu l'usager étant affilié à la CNPS ainsi que les périodes sur les quelles ses employeurs ont effectivement cotisé pour lui. Si tout est en règle, il renseigne ensuite les résultats de sa recherche dans un document Word prédéfini qui en fin de compte sera l'AAVCS, l'imprime et le transmet manuellement au gestionnaire de compte qui le vérifie à nouveau et le signe puis le renvoi au technicien de la sécurité sociale qui le délivre à l'usager.

2- Critique de l'existant

D'après le recueil de l'existant énoncé ci-dessus, nous avons pu ressortir les insuffisances suivantes :

- Chaque usager vient physiquement déposer son dossier d'AAVCS au sein du CPS ce qui entraine une longue file d'attente;
- ➤ La recherche des informations relatives à l'AAVCS se fait de manière manuelle dans le système, causant ainsi une perte de temps et une recherche fastidieuse des données ;
- Le TSS renseigne manuellement les résultats de ses recherches dans le fichier Word modèle, on note là une possibilité d'erreur au moment de la saisie ;
- Le TSS imprime l'AAVCS puis l'envoi au gestionnaire de compte qui doit le vérifier à nouveau, le signé et le renvoyer au TSS qui le délivre ensuite à l'usager. Ici les transactions physiques causent une perte de temps.

3- Problématique

Au vu des limites ressorties ci-dessus ne serait-il pas judicieux d'opter pour une solution d'automatisation du processus de délivrance des attestations d'affiliation et de versement des cotisations sociales ?

4- Propositions des solutions

A la suite des différentes critiques apportées dans le but d'améliorer le processus de délivrance des AAVCS, nous proposons de concevoir une application qui permettra aux :

- Usagers de :
 - ✓ Déposer les dossiers d'AAVCS en ligne ;





✓ Consulté l'évolution de son dossier

❖ TSS de :

- ✓ Recevoir les dossier d'AAVCS sur leur poste
- ✓ Valider ou rejeter un dossier
- ✓ Etablir un AAVCS
- ✓ Transmettre l'AAVCS au gestionnaire de compte ;
- Gestionnaires de compte de :
 - ✓ Recevoir l'AAVCS établi
 - ✓ Imprimé l'AAVCS ;
- II- Méthodologie
- 1- Présentation du langage de modélisation

UML (Unified Modeling Language) se définit comme un langage de modélisation graphique et textuel destiné à comprendre et décrire les besoins, spécifier et documenter les systèmes, esquisser des architectures logicielles, concevoir des solutions et communiquer des points de vue. UML unifie à la fois les notions et les concepts orientés objets. Il ne s'agit pas d'une simple notation graphique car les concepts transmis par un diagramme ont une sémantique précise et sont porteurs de sens au même titre que les mots d'un langage. C'est un langage standard de modélisation des systèmes d'information à objet. La version d'UML utilisé ici est la version 1.3 qui compte 9 diagrammes, et une subdivision du système en 2 grandes vues :

- La vue statique : représente le système physiquement et comporte 5 diagrammes :
 - Diagramme de cas d'utilisation ;
 - Diagramme de classe ;
 - Diagramme d'objet ;
 - Diagramme de déploiement ;
 - Diagramme de composant.
- La vue dynamique : représente les interactions effectuées dans le système. Elle comporte 4 diagrammes :
 - Diagramme d'activité ;
 - > Diagramme de collaboration ;
 - > Diagramme d'état transition ;
 - Diagramme de séquence.





UML étant un langage formel et normalisé, il est un bon support de communication dans l'élaboration d'une solution informatique, sa polyvalence et sa souplesse font de lui un langage universel. Pour développer des solutions, on a besoin de lui associé à une méthode générique qui s'attache à ses diagrammes, par exemple la méthode UP (Unified Process ou Processus Unifié), il en existe plusieurs : R-UP, X-UP, 2TUP, AUP, AM, ...

Approd	che générale	Merise	UML
		Méthode de modélisation	Langage de modélisation
		de données et traitement	des données, système de
		orienté base de données	notation orienté objet
		relationnelles	
Cycle	Cycle de vie	Rend compte de tous les	Ne définit pas le cycle de vie
		niveaux du cycle de vie	
	Cycle d'abstraction	Propose les niveaux	Permet de modéliser ces
		d'abstraction vérifier la	niveaux mais en utilisant le
		cohérence du système	même formalisme
	Cycle de décision	Prévoit les points de	Prévoit aussi un cycle
		passage et de contrôle dans	complet de décision
		la mise en œuvre	
Approche fonctionnelle		Utilise une approche	Le diagramme de fonction
		descendante ou le système	laisse place à des
		réel est décomposé en	diagrammes d'interactions
		activités, elles-mêmes	expliquant des scénarios de
		déclinées en fonctions. Les	cas d'utilisation qui
		fonctions sont composées	permettent de situer les
		de règles de gestion, elles-	besoins de l'utilisateur dans
		mêmes regroupées en	le contexte réel.
		opérations.	
Modéli	sation	Se base sur les concepts de	Se base sur une
		domaine d'acteur, de flux,	identification préalable du
		de modèle conceptuel	domaine
Les ac	teurs	Analyse les acteurs	Analyse les acteurs du point
		globalement	de vue du rôle joué par
			rapport au périmètre





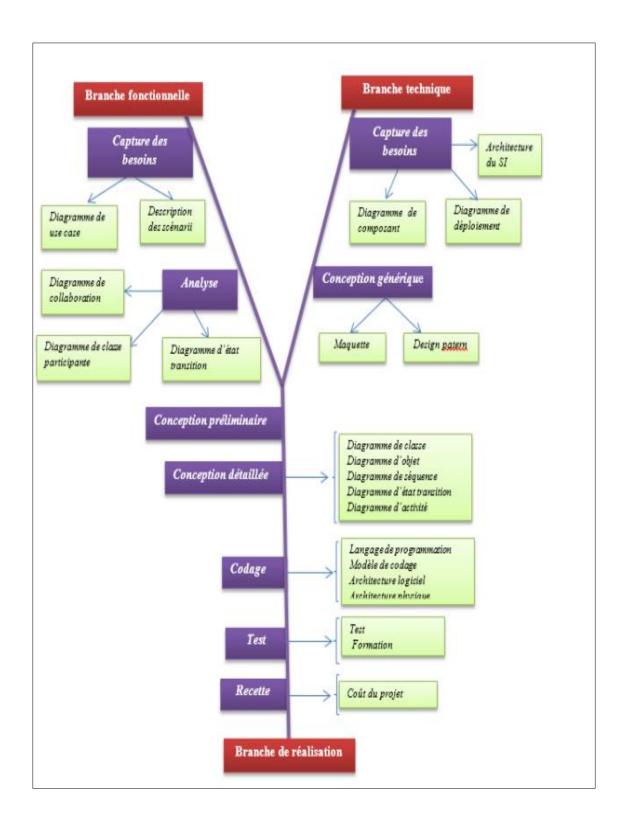
2- Présentation de la méthode d'analyse 2TUP

Notre choix est porté sur la méthode 2TUP (Two Track Unified Process) qui est un processus unifié. Du fait de son approche nouvelle et originale. Il cible les projets de toute taille, il est itératif dans ce sens qu'il permet de maitriser les inconnus et les incertitudes qui caractérises le système. Notre projet est basé sur un processus bien définit qui va de la détermination des besoins fonctionnels attendus du système jusqu'à la conception et le codage final.

Le processus unifié (PU) en anglais Unified Process (UP) est une méthode de développement logiciel conçue sur UML ; elle est itérative et incrémentale, centrée sur l'architecture, conduite par les cas d'utilisations et pilotée par les risques. On dit de la méthode UP qu'elle est générique, c'est-à-dire qu'elle décrit certains nombres de développements, que chaque société peut par la suite personnaliser afin de créer son propre processus plus adapté à ses besoins. C'est dans ce cadre que la société Val Techa a créé la méthode 2TUP. C'est un processus qui répond aux caractéristiques du processus unifié. La méthode 2TUP apporte une réponse aux contraintes de changement continuel imposé au système d'information de l'entreprise. En ce sens, il renforce le contrôle sur les capacités d'évolution et de correction de tel système. « 2 Track » signifie littéralement que le processus suit de chemins. Il s'agit « des chemins fonctionnels » et « d'architecture technique », qui correspondent aux deux axes de changement imposés au système d'information.











<u>La branche gauche (fonctionnelle)</u> : capitalise la connaissance du métier de l'entreprise. Elle constitue généralement un investissement pour le moyen au long terme. Les fonctions du système d'information sont en effet indépendantes des technologies utilisées. Cette branche comporte les étapes suivantes :

- La capture des besoins fonctionnels, qui produit un modèle des besoins focalisés sur le métier utilisateur;
- L'analyse.

La branche droite (architecture technique) : capitalise un savoir-faire technique. Elle constitue un investissement pour le court et le moyen terme. Les techniques développées pour le système peuvent l'être en effet indépendamment des fonctions à réaliser. Cette branche comporte les étapes suivantes :

- · La capture des besoins techniques ;
- La conception générique.

<u>La branche du milieu</u>: à l'issue des évolutions du modèle fonctionnel et de l'architecture technique, la réalisation du système consiste à fusionner les résultats des 2 branches. Cette fusion conduit à l'obtention d'un processus en forme de Y. cette branche comporte les étapes suivantes :

- La conception des besoins préliminaires ;
- La conception détaillée ;
- Le codage ;
- Recette.

Voici une présentation rapide des différents diagrammes UML qui vont être utilisés pour l'analyse et la conception de notre projet :

- Le diagramme des cas d'utilisation : représente la structure des fonctionnalités nécessaire aux utilisateurs du système. Il est normalement utilisé lors des étapes de capture des besoins fonctionnels et techniques ;
- Le diagramme d'activité : représente les règles d'enchainement des activités et actions dans le système. Il peut être assimilé comme un algorithme mais schématisé.
- Le diagramme de classe : surement l'un des digrammes les plus importants dans un développement orienté objet. Sur la branche fonctionnelle, ce diagramme est prévu pour développer la structure des entités manipulées par les utilisateurs. En conception, le diagramme de classe représente la structure d'un code orienté objet.





- Le diagramme de séquence : représente les échanges de messages entre objets, dans le cadre d'un fonctionnement particulier du système.
- ➤ Le diagramme de déploiement : est un diagramme permettant la représentation des nœuds ou des instances de nœuds sur lesquels s'exécute le système.
- Le diagramme de composants : est un diagramme qui décrit l'organisation du système du point de vue des éléments du logiciel comme les modules (fichiers sources, bibliothèques, exécutables). Ce diagramme permet de mettre en évidence les dépendances entre les composants.

III- Capture des besoins fonctionnels

La capture des besoins permet d'identifier les acteurs et les fonctionnalités réalisées par le système. Elle produit un modèle de besoin focalisé sur le rôle des acteurs.la technique des cas d'utilisation d'UML est au centre de cette phase du processus 2TUP. La démarche ici consiste en :

- ✓ L'identification des acteurs :
- ✓ L'identification des cas d'utilisation ;
- ✓ La description des cas d'utilisation.

1- Diagramme des cas d'utilisation

Il permet de modéliser un système sur le point de vue des utilisateurs ; c'est le tout premier diagramme à construire ; il permet de recueillir, d'analyser et d'organiser des besoins ; avec lui débute l'étape d'analyse du système.

a- Formalisme

Les composants de base des diagrammes de cas d'utilisation sont : l'acteur, le cas d'utilisation et l'association.

L'acteur : Un acteur est l'idéalisation d'un rôle joué par une personne externe, un processus ou une chose qui interagit avec un système. Il se représente par un petit bonhomme :





Les cas d'utilisation : les cas d'utilisations représentent des fonctionnalités fournies par le système. Ils sont décrits sous forme de verbes à l'infinitif. Un cas d'utilisation correspond à un objectif du système motivé par un besoin d'un ou de plusieurs utilisateurs. L'ensemble des cas d'utilisation décrit le but du système. Ainsi un cas d'utilisation permet de déterminer les besoins du système.

> **Les relations** : on distingue cinq types de relations dans le diagramme des cas d'utilisation, qui sont :

Cas d'utilisation*

L'association, qui exprime la participation d'un acteur à un cas d'utilisation ;

 L'héritage d'un acteur A vers un acteur B, indiquant qu'une instance de A peut communiquer avec les mêmes cas d'utilisation que les instances de B;

Acteur B

- La généralisation d'un cas d'utilisation A vers un cas d'utilisation B, indiquant que A est une spécification de B ;

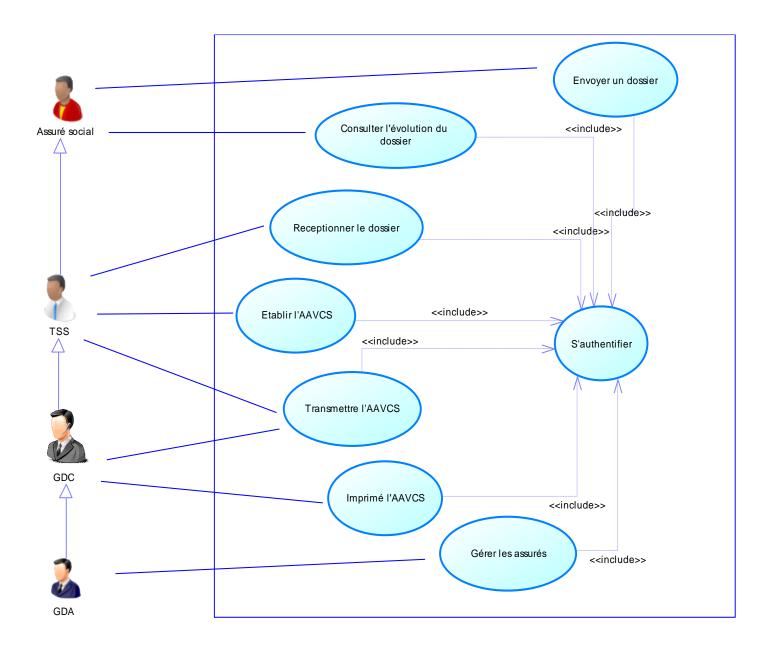
 L'extension d'un cas d'utilisation A vers un cas d'utilisation B, indiquant qu'une instance de B peut être augmentée par le comportement spécifié dans A, à l'endroit défini par le point d'extension;

- L'inclusion d'un cas d'utilisation A vers un cas d'utilisation B, indiquant qu'une instance de A contiendra le comportement spécifié dans B à un endroit défini.





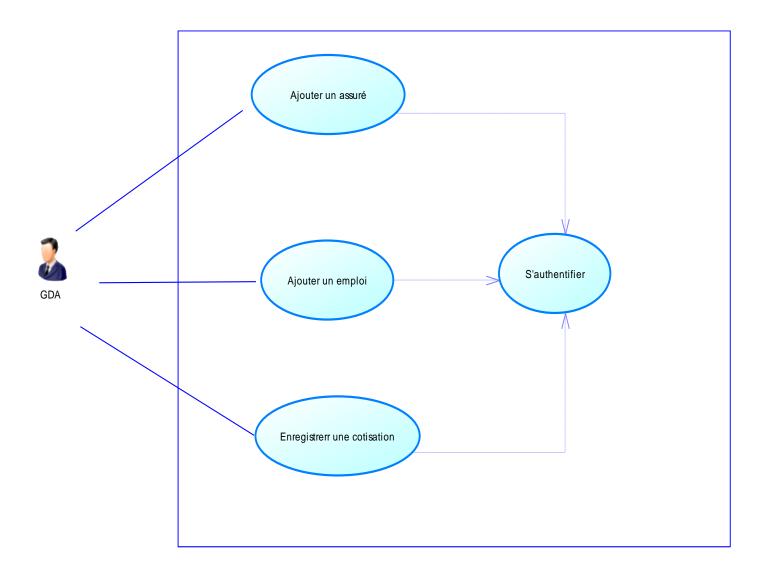
b- Diagramme des cas d'utilisation globale







c- Diagramme de cas d'utilisation : Gérer les assurés







d- Description textuelle de quelques scenarii

La description textuelle est indispensable, parce qu'elle seule permet de communiquer facilement et précisément avec les utilisateurs, mais également d'identifier le contexte d'exécution de l'un ou de l'autre des enchainements. Un scénario est une instance d'un cas d'utilisation. On distingue trois types de scénarios :

- ✓ Le scénario nominal : il décrit le déroulement normal du cas d'utilisation ;
- ✓ Le scénario alternatif : il est une variante du scénario nominal ;
- ✓ Le scénario d'exception : il illustre un déroulement anormal du cas d'utilisation.

Pour notre part, nous avons choisi d'expliciter les cas d'utilisation de notre système à travers le formalisme suivant :

Cas d'utilisation No* : « cas d'utilisation »

Description : « décrit de manière explicite et spécifique le déroulement de chaque cas d'utilisation

Acteur: « acteur participant au cas d'utilisation »

Présupposé: « situation préliminaire pour remplir une condition »

Précondition : « condition devant être remplis avant le d »roulement d'un cas d'utilisation

Scénario nominal : « séquence d'actions normales associées à un cas d'utilisation »

Scénario alternatif : « séquence d'action pouvant également conduire à un succès »

Post condition du succès : « ce que le cas d'utilisation doit faire après que les conditions soient remplies »

Post condition de l'échec : « le cas d'utilisation ne fonctionne pas malgré que les conditions sont bonne »





Cas d'utilisation : s'authentifier

Cas d'utilisation : S'authentifier

<u>Description</u>: ce cas d'utilisation illustre la connexion et l'authentification d'un utilisateur à l'application

Acteur : Assuré social, TSS, gestionnaire de compte

<u>Présupposé</u> : l'utilisateur a un compte donc dispose donc d'un nom d'utilisateur et d'un mot de passe

<u>Précondition</u>: l'utilisateur lance l'application

Scénario nominal:

- 1- Le système lui affiche le formulaire de connexion
- 2- L'utilisateur saisie son login mot de passe
- 3- Le système vérifie si les informations fournies sont correctes
- 4- Le système ouvre la session de l'utilisateur

<u>Scénario alternatif</u>: Si à l'étape 3 du scénario principal l'une ou la totalité des informations fournies n'est pas correcte,

- 1- Le système notifie (login ou mot de passe incorrect)
- 2- le système renvoie à l'étape 1 du scénario principal

Post condition du succès : l'utilisateur s'est authentifié

Post condition échec : l'utilisateur ne s'est pas authentifié





♣ Cas d'utilisation : Envoyer un dossier

Cas d'utilisation : Envoyer un dossier

<u>Description</u>: ce cas d'utilisation permet de décrire le dépôt d'un dossier d'AAVCS en ligne

Acteur : Assuré social

Présupposé : l'assuré sélectionne le menu « Déposer un dossier d'AAVCS

Précondition :» l'assuré s'authentifie en se connectant à son compte assuré

Scénario nominal :

- 1- Le système lui affiche un formulaire contenant deux champs (l'un pour l'insertion de sa lettre adressée au chef centre et l'autre pour le scan de sa CNI)
- 2- L'usager insère les pièces demandées puis clique sur le bouton « Envoyer le dossier »
- 3- Le système vérifie si tous les champs sont renseignés
- 4- Le système notifie que le dossier a été envoyé avec succès.

<u>Scénario alternatif</u>: Si à l'étape 3 du scénario principal l'une ou la totalité des pièces à joindre n'est pas insérée,

- 1- Le système notifie (Veuillez renseigner tous les champs)
- 2- Le retourne à l'étape 1 du scénario nominal

Post condition du succès : le dossier a été envoyé

Post condition échec : le dossier n'a pas été envoyé





♣ Cas d'utilisation : Réceptionner un dossier

Cas d'utilisation : Réceptionner un dossier

<u>Description</u>: ce cas d'utilisation permet de décrire les différentes actions pouvant être menées lors de la réception d'un dossier d'AAVCS

Acteur: le TSS

Présupposé : le TSS s'authentifie

Précondition : le TSS sélectionne le menu « Réceptionner un dossier d'AAVCS »

Scénario nominal :

- 1- Le système lui affiche un formulaire contenant deux champs (l'un contenant la lettre adressée au chef centre et l'autre contenant le scan de sa CNI)
- 2- Le TSS extrait les fichiers et vérifie leurs conformités
- 3- Le TSS clique sur valider le dossier
- 4- Le système notifie que le dossier a été réceptionné avec succès.

<u>Scénario alternatif</u>: Si à l'étape 2 du scénario principal l'une ou la totalité des pièces jointes n'est pas conforme,

- 1- Le TSS clique sur rejeter le dossier
- 2- Le système notifie que le dossier a été rejeté.

Post condition du succès : le dossier a été réceptionné

Post condition échec : le dossier n'a pas été réceptionné





♣ Cas d'utilisation : Etablir l'AAVCS

Cas d'utilisation : Etablir l'AAVCS

Description : ce cas d'utilisation décrit l'établissement des AAVCS

Acteur : le TSS

Présupposé : le TSS sélectionne le menu « Etablir

Précondition : l'AAVCS » le TSS s'authentifie

Scénario nominal :

1- Le système lui affiche le prototype de l'AAVCS à établir

- 2- Le TSS vérifie l'intégrité du prototype
- 3- Le TSS clique sur établir l'AAVCS
- 4- Le système notifie que l'AACVS a été établie avec succès

Scénario alternatif: Si à l'étape 2 du scénario principal le,

1- Le TSS clique sur annuler

2- Le système notifie que l'établissement de l'AAVCS a été annulé.

Post condition du succès : l'AAVCS a été établie avec succès

Post condition échec : l'AAVCS n'a pas été établie





2- Le diagramme de séquences

Il documente les interactions à mettre en œuvre entre les classes pour réaliser un résultat tel qu'un cas d'utilisation.

a- Caractéristiques

Dans le diagramme de séquences :

- Les communications entre les classes sont reconnues comme des messages ;
- Il énumère les objets horizontalement et le temps verticalement ;
- Il modélise l'exécution des différents messages en fonction du temps.
- b- Formalise

Le digramme de séquences est constitué des éléments suivants :

- Les classes, les objets, les acteurs ;

La ligne de vie : elle identifie l'existence de l'objet par rapport au temps. Elle est représentée par une ligne interrompue :

- L'activation : elle indique quand l'objet effectue l'action et se situe sur sa ligne de vie :

- Les messages, ils indiquent les communications entre les objets. Nous distinguons deux types de messages :
 - ✓ Les messages synchrones : qui se représentent par une flèche à l'extrémité pleine pointant vers le destinataire du message.

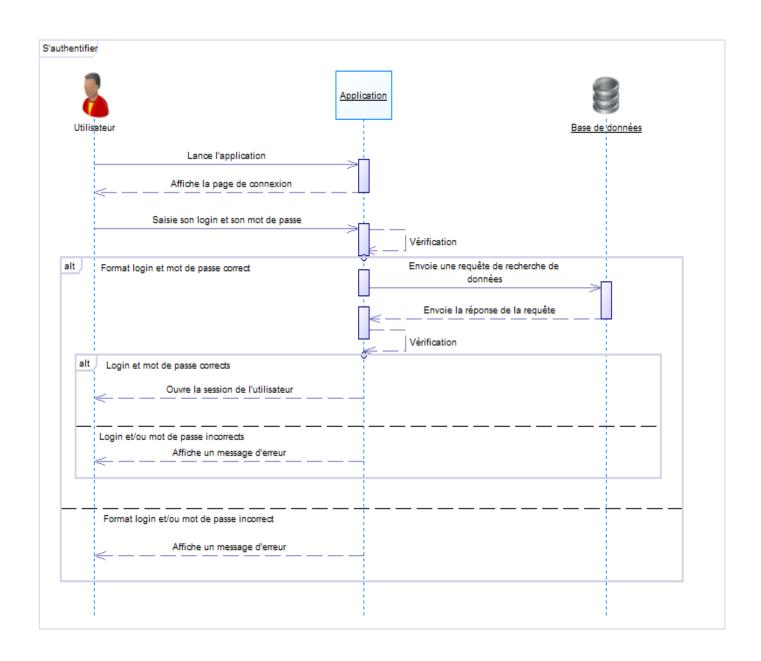




✓ Les messages asynchrones : qui se représentent par une flèche à l'extrémité ouverte.

Les messages peuvent être suivis de réponses qui se représentent par une flèche en pointillé :

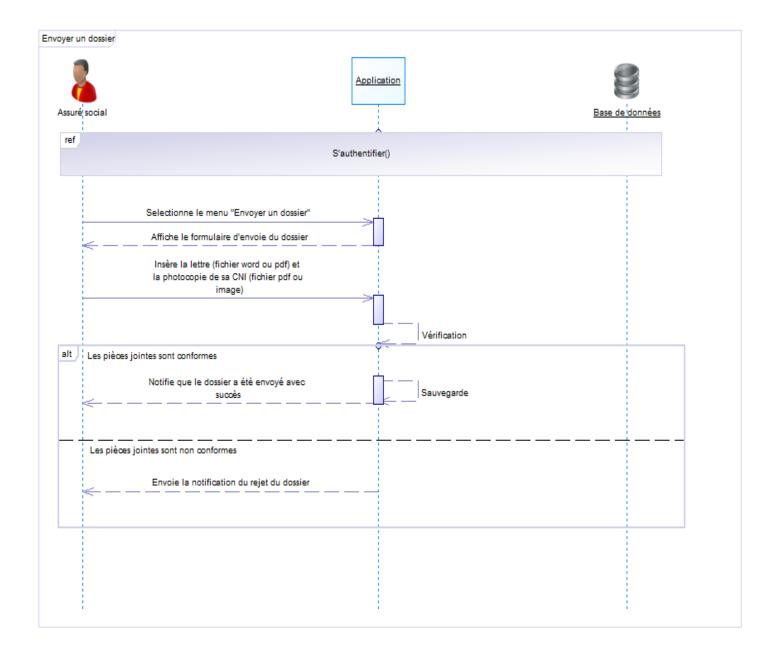
- c- Les diagrammes de séquences
- S'authentifier







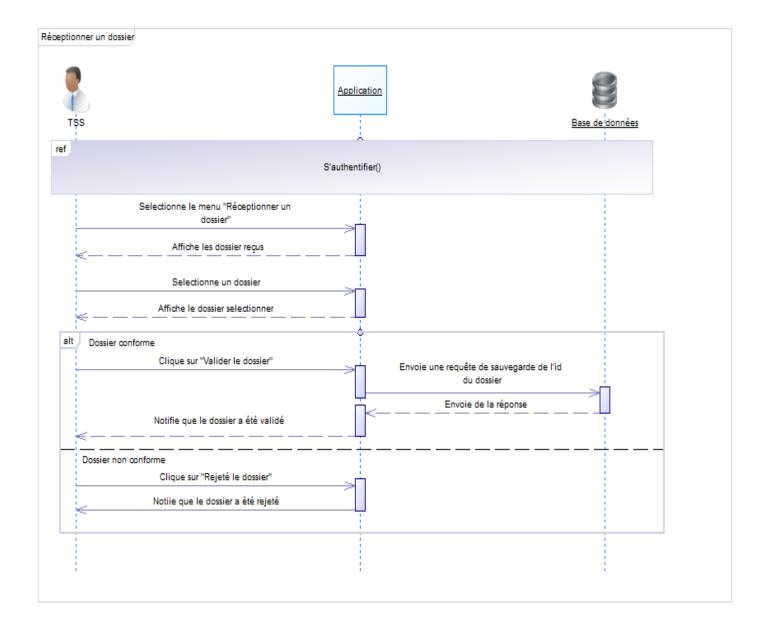
Envoyer un dossier







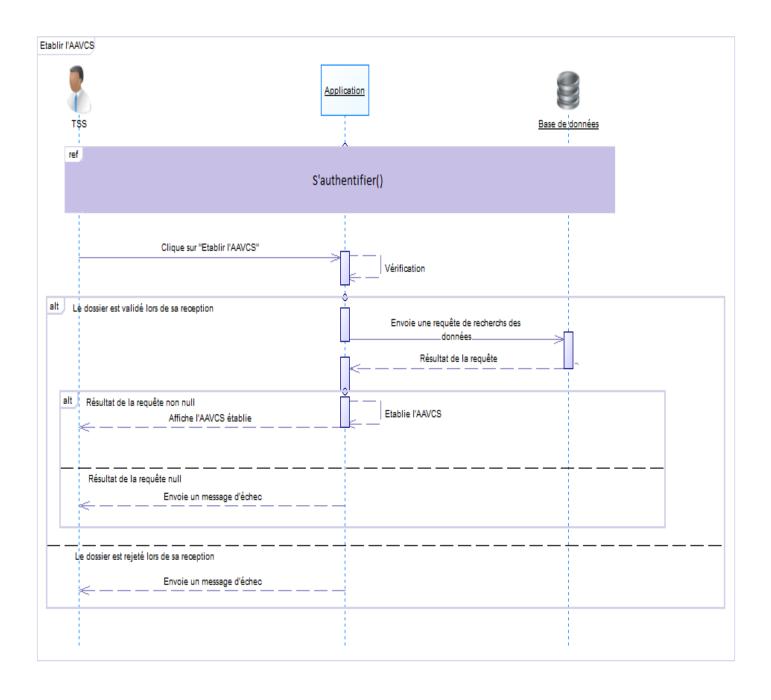
Réceptionner un dossier







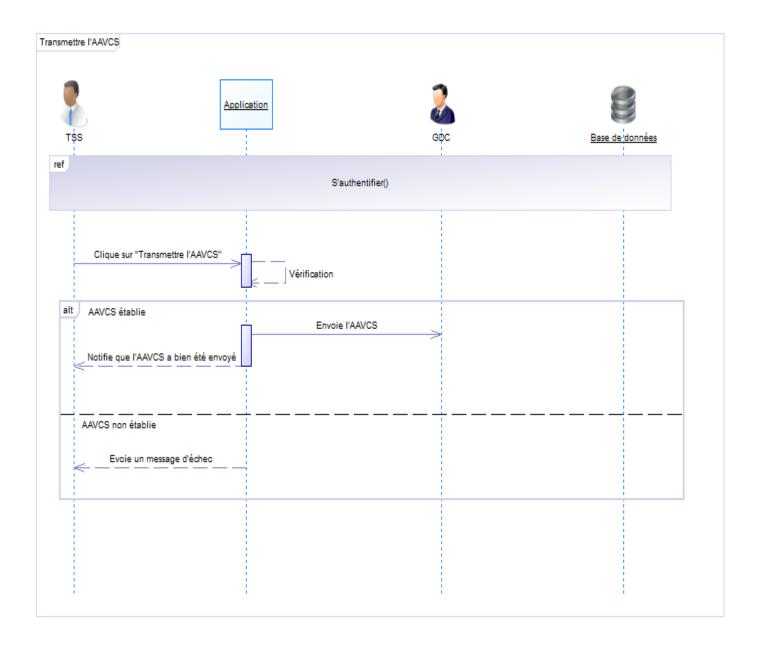
♣ Etablir l'AAVCS







♣ Transmettre l'AAVCS







CONCLUSION

Le présent dossier d'analyse nous a permis de faire une spécification de base du projet, une présentation détaillée des différents cas d'utilisation qui seront pris en compte dans notre application. Le recensement des besoins, le langage de modélisation ainsi que les différentes fonctionnalités pour l'implémentation. La suite de notre travail consistera à la modélisation et la mise sur pied de l'application de par le dossier de conception et de réalisation.





PARTIE IV: DOSSIER DE CONCEPTION





Résumé:

Le dossier de conception ici présent est une partie permettant de présenter l'architecture choisie pour l'application à réaliser; la conception détaillée et aussi l'ensemble des composants nécessaires pour le bon fonctionnement de l'application.

INTRODUCTION

- I- DIAGRAMME DE CLASSES
- II- DIAGRAMME D'ETAT TRANSITION CONCLUSION





INTRODUCTION

Concevoir un logiciel ne s'apprend pas dans un livre, mais par la pratique et l'étude des systèmes existant. Un système bien conçu est facile à réaliser, à maintenir, à comprendre et est fiable. La phase de conception est la plus cruciale du processus de développement d'un logiciel. Après avoir réalisé le cahier de charges de l'application, il sera question dans cette partie de réaliser le dossier de conception de ladite application. Le modèle de conception présente de façon détaillée le fonctionnement du futur système d'information. Il est donc question dans ce document de présenter le déroulement de la conception, des besoins utilisateurs aux codes en utilisant la méthode 2TUP. Pour se faire, nous aurons à concevoir dans cette étape : le diagramme de classe et le diagramme d'état transition





I- Diagramme de classe

Le diagramme de classe exprime la structure statique du système en terme de classes et de relations entre ces classes. L'intérêt du diagramme de classe est de modéliser les entités du SI. Le diagramme se de classes permet de représenter l'ensemble des informations qui sont gérées par le domaine.

1- Formalisme

Le diagramme de classe met en évidence d'éventuelle relation entre ces classes, il comporte quelques concepts : la classe, les attributs, l'identifiant, les opérations (méthodes) et les relations.

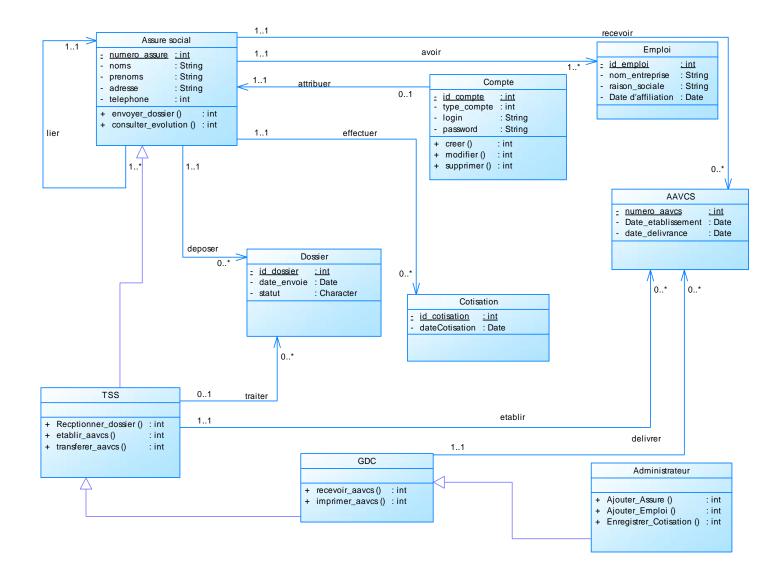
Un diagramme de classe est constitué des éléments suivants :

- Classe : on appelle classe la structure d'un objet, c'est-à-dire la déclaration de l'ensemble des entités qui compose un objet
- Associations : c'est la relation générique entre deux classes ; elle peut être surmontée d'une étiquette et bordée par des multiplicités
- Composition :si une classe ne peut exister elle-même, mais être membre d'une autre classe alors, elle possède une classe composition avec la classe contenante
- Agrégation : type d'association mettant en évidence une classe agrégée et une classe agrégat ; chaque objet de la classe agrégée est associé à un ou plusieurs objets de la classe agrégat
- Attribut : information élémentaire composant une classe. Un attribut peut permettre d'identifier la classe
- Généralisation : permet d'identifier parmi les objets d'une classe (générique) des sous ensemble d'objet (des classes spécialisées) ayant des définitions spécifiques
- Multiplicité : elle définit le nombre d'instance de l'association pour une instance de la classe. Elle est définie par un nombre entier ou un intervalle de valeur





2- Diagramme de classe du système







II- Diagramme d'état transition

Il est utilisé pour documenter divers modes (états) qu'une classe peut prendre ainsi que les événements qui causent une transition d'état.

1- Formalise

Il a pour rôle de représenter les traitements qui vont gérer le domaine étudié. Les éléments constitutifs d'un digramme d'état transition sont : l'état, la transition (ou évènement), l'état initial et l'état final.

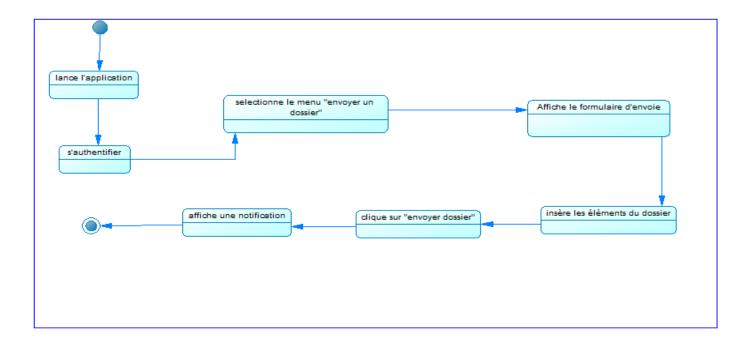
Un digramme d'état transition est constitué des éléments suivants :

- L'état ou activité : un état correspond à une situation durable dans laquelle se trouve les objets d'une classe. Il est représenté par un rectangle aux coins arrondis contenant le nom de l'état
- La transition ou évènement : décrit le changement de l'état d'un objet provoqué par un évènement. Une transition est une relation entre deux états représentée par une flèche orientée sur laquelle est noté l'évènement qui concours au changement d'état
- L'état initial : c'est l'état de l'objet avant toute transition ; un seul état initial est autorisé dans un diagramme d'état transition et est représenté par un cercle plein
- L'état final : représente la destruction de l'objet qu'on modélise. Selon les évènements, il peut exister plusieurs états finaux dans un diagramme d'état transition. Cet état est représenté d'un cercle plein entouré d'un autre cercle





- 2- Diagrammes d'état transition du système
- Envoyer un dossier



Réceptionner un dossier





Conclusion

Le dossier de conception nous a permis de ressortir les données et les processus nécessaires pour la création de notre base de données et l'implémentation de notre application. Les différents éléments modélisés dans cette partie nous ont permis d'avoir une vue globale sur les différents modules de notre application, des lors, l'étape suivante de notre projet sera la rédaction du dossier de réalisation tenant compte des différents éléments modélisés plus haut.





PARTIE IV: DOSSIER DE REALISATION





Résumé:

Le dossier de réalisation est un document qui présente l'implémentation ou l'élaboration de l'application étudiée dans les dossiers d'analyse et de conception.



- I- EQUIPE DE DEVELOPPEMENT ET CHOIX DES OUTILS
- II- PRESENTATION DES ARCHITECTURES
- III- MODELE PHYSIQUE DE DONNEES
- IV- PRESENTATION DES DIAGRAMMES DES INTERVENANTS

CONCLUSION





INTRODUCTION

La conception et la réalisation étant des maillons importants dans le processus de développement d'une application car elles permettent de définir et spécifier les différents éléments constituant l'application ou le système étudié, mais la réalisation est l'étape qui permet de reproduire concrètement les différents objets issus de la conception et de donner vie au système automatique. Il sera question pour nous tout au long de cette partie de présenter les technologies choisies, les architectures et le modèle physique de données de l'application.





- I- Equipe de développement et choix des outils
- a- Présentation de l'équipe technique

Noms et prénoms	Fonctions	Rôles
M. NDJENG NKOLO Éric	Chef du service d'étude et développement de la CNPS	Encadreur professionnel
M. ASSAWOGA Nicolas	Enseignant à l'IAI Cameroun	Encadreur académique
M. NGOLLE DIN Charles	Etudiant à l'IAI Cameroun	Maitre d'œuvre

b- Outils matériels

Pour le développement de notre application nous avons besoin de :

- ✓ Laptop (dual-core 2.4 2.4GHz, RAM 4Go, disque dur 320Go)
- ✓ Imprimante
- ✓ Clé USB
- ✓ Onduleur

c- Outils logiciels

Logiciels	Rôles
Windows 8	Système d'exploitation de l'ordinateur
Wamp server	Serveur web et SGBD
Microsoft office 2016	Editeur de texte
Bi defender	Antivirus
Mozilla Firefox, opéra mini, chrome	Navigateur
Power AMC	Modélisation de la solution
Macromedia	Environnement de développement
Netbeans	intégré
Sublime text	Editeur de code

d- Présentation du langage de programmation

Dans le cadre de la mise sur pieds de notre application, le langage de programmation choisi est le JAVA EE, à qui s'ajoute le HTML5, le CSS3, le SQL et le JavaScript.

- ➤ JAVA EE (Java Enterprise Edition) : langage de programmation orienté objet utilisé pour développer des application web.
- HTML5 : langage de balise utilisé pour structurer les pages web.
- CSS3 : langage de style utilisé pour le design de l'application.

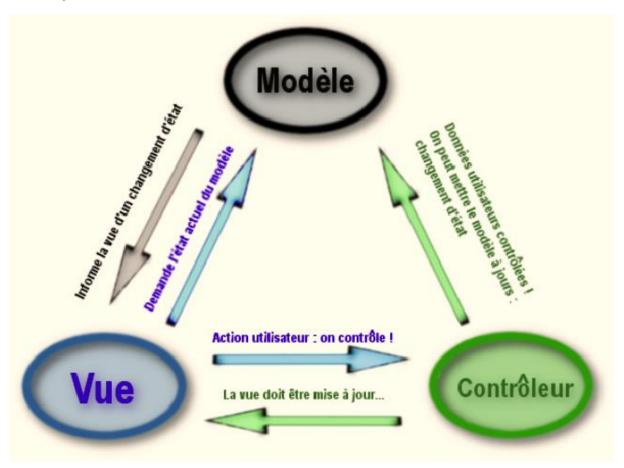




- JavaScript : langage de programmation des scripts orienté objet utilisé pour dynamiser les pages web.
- > SQL : langage de requêtes utilisé pour communiquer avec le SGBD.
 - II- Présentation des architectures
- 1- Présentation de l'architecture logicielle

La logique applicative de notre projet est basée sur l'architecture MVC qui est très pratique dans les projet web.

Le modèle Vue Contrôleur (MVC) est une architecture et une méthode de conception qui organise l'interface homme-machine (IHM) d'une application logicielle. Ce paradigme divise l'IHM en un modèle (modèle de données), une vue (interface utilisateur) et un contrôleur (logique de contrôle, gestion des évènements, synchronisation), chacun ayant un rôle précis dans l'interface. Ce modèle d'architecture impose la séparation entre les données, la présentation et les traitements, ce qui donne trois parties fondamentales dans l'application finale : le modèle, la vue et le contrôleur.







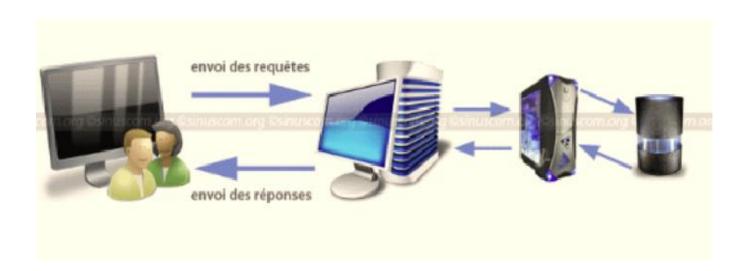
2- Présentation de l'architecture physique de déploiement

Dans le cadre de notre travail, nous avons choisie l'architecture 3-tiers encore appelé architecture trois niveaux ou architecture à trois couches.

Cette architecture est l'application du modèle plus général qu'est le multi-tiers. Il s'agit d'un modèle logique d'architecture applicative qui vise à modéliser une application comme un empilement de trois couches logicielles dont les tâches sont clairement définies :

- La présentation des données : correspondant à l'affichage, la restitution sur le poste de travail, le dialogue avec l'utilisateur.
- Le traitement métier des données : correspondant à la mise en œuvre de l'ensemble des règles de gestion et de la logique applicative.
- L'accès aux données persistantes : correspondant aux données qui sont destinées à être conservées sur la durée, voire de manière définitive.

Dans cette approche, les couches communiquent entre elles au travers d'un modèle d'échange, et chacune d'entre elles propose un ensemble de services rendus. Les services d'une couche sont mis à disposition de la couche supérieure. On s'interdit qu'une couche invoque les services d'une couche plus basse que la couche immédiatement inférieure ou plus haute que la couche immédiatement supérieure. Le rôle de chacune des couches et leurs interfaces étant bien définis, les fonctionnalités de chacune d'entre elles peuvent évoluer sans induire de changement dans les autres couches.







a) Avantages de l'architecture 3-tiers

Les avantages de l'architecture 3-tiers sont principalement au nombre de quatre :

- Les requêtes clients vers le serveur sont d'une plus grande flexibilité que dans celles de l'architecture 2-tiers basées sur le langage SQL; en effet les appels clients ne spécifient que des paramètres et des structures de données pour les valeurs de retour.
- ➤ Faisant le pendant avec la première remarque, l'utilisateur n'est pas supposé connaître le langage SQL, qui ne sera pas implémenté dans la partie client qui ne s'occupe que de fonctions d'affichage. De fait des modifications peuvent être faites au niveau du SGBD sans que cela impacte la couche client. Par ailleurs et bien que nous ayons mentionné le langage SQL au niveau des bases de données, on peut très bien envisager une organisation des données sans présupposition quant au langage lui-même et à leur organisation (relationnelle, hiérarchique...). Cette flexibilité permet à une entreprise d'envisager dans le cadre d'une architecture 3-tiers une grande souplesse pour l'introduction de toutes nouvelles technologies.
- D'un point de vue développement, la séparation qui existe entre le client, le serveur et le SGBD permet une spécialisation des développeurs sur chaque tiers de l'architecture.
- Plus de flexibilité dans l'allocation des ressources ; la portabilité du tiers serveur permet d'envisager une allocation et ou modification dynamique au gré des besoins évolutifs au sein d'une entreprise.

b) Limites de l'architecture 3tiers

L'architecture trois-tiers a corrigé les excès du client lourd en centralisant une grande partie de la logique applicative sur un serveur HTTP. Le poste client, qui ne prend à sa charge que la présentation et les contrôles de saisie, s'est trouvé soulagé et plus simple à gérer. En revanche, le serveur HTTP constitue la pierre angulaire de l'architecture et se trouve souvent fortement sollicité : il est difficile de répartir la charge entre client et serveur. On se retrouve confronté aux épineux problèmes de dimensionnement serveur et de gestion de la montée en charge rappelant l'époque des mainframes. De plus, les solutions mises en œuvre sont relativement complexes à maintenir et la gestion des sessions est compliquée, mais reste possible. Les contraintes semblent inversées par rapport à celles rencontrées avec les architectures deux tiers : le client est soulagé, mais le serveur est fortement sollicité. Le juste équilibre de la charge entre client et serveur semble atteint avec la génération suivante : les architectures n-tiers.



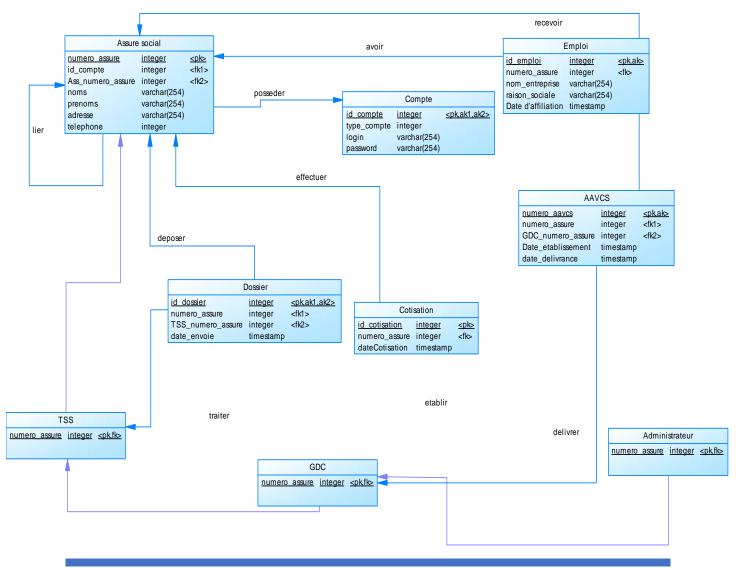


III- Modèle physique de données

a- Présentation du MPD

Un modèle physique de données est une définition des données à l'intérieur de la structure physique de l'ordinateur, c'est-à-dire le résultat de la décision technique qui a été prise en fonction des objets et des contraintes techniques. Un modèle physique de données est un formalisme qui permet de préciser le système de stockage employé pour un système de gestion de base de données, c'est-à-dire traduire dans un langage de définition des données, généralement le SQL.

b- MPD du système étudié







c- Script de la base de données

/*==========*/
/* Nom de SGBD : Sybase SQL Anywhere 11 */
/* Date de création : 02/10/2017 12:55:27 */
/*===========*/
create aavcs;
use aavcs;
if exists(select 1 from sys.sysforeignkey where role='FK_AAVCS_ASSOCIATI_TSS') then
alter table Aavcs
delete foreign key FK_AAVCS_ASSOCIATI_TSS
end if;
if exists(select 1 from sys.sysforeignkey where role='FK_AAVCS_ASSOCIATI_ASSURESO') then
alter table Aavcs
delete foreign key FK_AAVCS_ASSOCIATI_ASSURESO
end if;
if exists(select 1 from sys.sysforeignkey where role='FK_AAVCS_ASSOCIATI_GDC') then
alter table Aavcs
delete foreign key FK_AAVCS_ASSOCIATI_GDC
end if;
if exists(select 1 from sys.sysforeignkey where role='FK_ADMINIST_GENERALIS_GDC') then





alter table Administrateur delete foreign key FK_ADMINIST_GENERALIS_GDC end if; if exists(select 1 from sys.sysforeignkey where role='FK_ASSURESO_ASSOCIATI_ASSURESO') then alter table AssureSocial delete foreign key FK_ASSURESO_ASSOCIATI_ASSURESO end if: if exists(select 1 from sys.sysforeignkey where role='FK COMPTE ASSOCIATI ASSURESO') then alter table Compte delete foreign key FK_COMPTE_ASSOCIATI_ASSURESO end if: if exists(select 1 from sys.sysforeignkey where role='FK_COTISATI_ASSOCIATI_ASSURESO') then alter table Cotisation delete foreign key FK COTISATI ASSOCIATI ASSURESO end if; if exists(select 1 from sys.sysforeignkey where role='FK DOSSIER ASSOCIATI TSS') then alter table Dossier delete foreign key FK_DOSSIER_ASSOCIATI_TSS end if;

if exists(select 1 from sys.sysforeignkey where role='FK_DOSSIER_ASSOCIATI_ASSURESO') then alter table Dossier delete foreign key FK_DOSSIER_ASSOCIATI_ASSURESO





end if;

```
if exists(select 1 from sys.sysforeignkey where
role='FK_EMPLOI_ASSOCIATI_ASSURESO') then
  alter table Emploi
    delete foreign key FK_EMPLOI_ASSOCIATI_ASSURESO
end if;
if exists(select 1 from sys.sysforeignkey where role='FK_GDC_GENERALIS_TSS')
then
  alter table Gdc
    delete foreign key FK_GDC_GENERALIS_TSS
end if;
if exists(select 1 from sys.sysforeignkey where
role='FK_TSS_GENERALIS_ASSURESO') then
  alter table Tss
    delete foreign key FK_TSS_GENERALIS_ASSURESO
end if:
if exists(
 select 1 from sys.sysindex i, sys.systable t
 where i.table_id=t.table_id
   and i.index_name='ASSOCIATION12_FK'
   and t.table_name='Aavcs'
) then
 drop index Aavcs.ASSOCIATION12_FK
end if:
if exists(
 select 1 from sys.sysindex i, sys.systable t
 where i.table id=t.table id
```





```
and i.index_name='ASSOCIATION8_FK'
   and t.table_name='Aavcs'
) then
 drop index Aavcs.ASSOCIATION8_FK
end if;
if exists(
  select 1 from sys.sysindex i, sys.systable t
 where i.table_id=t.table_id
   and i.index_name='ASSOCIATION3_FK'
   and t.table_name='Aavcs'
) then
 drop index Aavcs.ASSOCIATION3_FK
end if;
if exists(
 select 1 from sys.sysindex i, sys.systable t
 where i.table_id=t.table_id
   and i.index_name='AAVCS_PK'
   and t.table_name='Aavcs'
) then
  drop index Aavcs.AAVCS_PK
end if;
if exists(
  select 1 from sys.systable
 where table_name='Aavcs'
   and table_type in ('BASE', 'GBL TEMP')
) then
  drop table Aavcs
```





```
end if;
if exists(
 select 1 from sys.sysindex i, sys.systable t
 where i.table_id=t.table_id
   and i.index_name='ADMINISTRATEUR_PK'
   and t.table_name='Administrateur'
) then
 drop index Administrateur.ADMINISTRATEUR_PK
end if;
if exists(
 select 1 from sys.systable
 where table_name='Administrateur'
   and table_type in ('BASE', 'GBL TEMP')
) then
  drop table Administrateur
end if;
if exists(
 select 1 from sys.sysindex i, sys.systable t
 where i.table_id=t.table_id
   and i.index_name='ASSOCIATION10_FK'
   and t.table_name='AssureSocial'
) then
  drop index AssureSocial.ASSOCIATION10_FK
end if;
if exists(
  select 1 from sys.sysindex i, sys.systable t
```





```
where i.table_id=t.table_id
   and i.index_name='ASSURESOCIAL_PK'
   and t.table_name='AssureSocial'
) then
 drop index AssureSocial.ASSURESOCIAL_PK
end if;
if exists(
  select 1 from sys.systable
 where table_name='AssureSocial'
   and table_type in ('BASE', 'GBL TEMP')
) then
  drop table AssureSocial
end if;
if exists(
 select 1 from sys.sysindex i, sys.systable t
 where i.table_id=t.table_id
   and i.index_name='ASSOCIATION13_FK'
   and t.table_name='Compte'
) then
  drop index Compte.ASSOCIATION13_FK
end if;
if exists(
  select 1 from sys.sysindex i, sys.systable t
 where i.table_id=t.table_id
   and i.index_name='COMPTE_PK'
   and t.table_name='Compte'
) then
```





```
drop index Compte.COMPTE_PK
end if;
if exists(
 select 1 from sys.systable
 where table_name='Compte'
   and table_type in ('BASE', 'GBL TEMP')
) then
  drop table Compte
end if;
if exists(
 select 1 from sys.sysindex i, sys.systable t
 where i.table_id=t.table_id
   and i.index_name='ASSOCIATION9_FK'
   and t.table_name='Cotisation'
) then
 drop index Cotisation.ASSOCIATION9_FK
end if;
if exists(
 select 1 from sys.sysindex i, sys.systable t
 where i.table_id=t.table_id
   and i.index_name='COTISATION_PK'
   and t.table_name='Cotisation'
) then
 drop index Cotisation.COTISATION_PK
end if;
if exists(
```





```
select 1 from sys.systable
 where table_name='Cotisation'
   and table_type in ('BASE', 'GBL TEMP')
) then
  drop table Cotisation
end if;
if exists(
  select 1 from sys.sysindex i, sys.systable t
 where i.table_id=t.table_id
   and i.index_name='ASSOCIATION11_FK'
   and t.table_name='Dossier'
) then
 drop index Dossier.ASSOCIATION11_FK
end if;
if exists(
 select 1 from sys.sysindex i, sys.systable t
 where i.table_id=t.table_id
   and i.index_name='ASSOCIATION2_FK'
   and t.table_name='Dossier'
) then
 drop index Dossier.ASSOCIATION2_FK
end if;
if exists(
  select 1 from sys.sysindex i, sys.systable t
 where i.table_id=t.table_id
   and i.index_name='DOSSIER_PK'
   and t.table_name='Dossier'
```





```
) then
 drop index Dossier.DOSSIER_PK
end if;
if exists(
 select 1 from sys.systable
 where table_name='Dossier'
   and table_type in ('BASE', 'GBL TEMP')
) then
  drop table Dossier
end if;
if exists(
 select 1 from sys.sysindex i, sys.systable t
 where i.table_id=t.table_id
   and i.index_name='ASSOCIATION4_FK'
   and t.table_name='Emploi'
) then
 drop index Emploi.ASSOCIATION4_FK
end if;
if exists(
 select 1 from sys.sysindex i, sys.systable t
 where i.table_id=t.table_id
   and i.index_name='EMPLOI_PK'
   and t.table_name='Emploi'
) then
 drop index Emploi.EMPLOI_PK
end if;
```





```
if exists(
 select 1 from sys.systable
 where table_name='Emploi'
   and table_type in ('BASE', 'GBL TEMP')
) then
  drop table Emploi
end if;
if exists(
 select 1 from sys.sysindex i, sys.systable t
 where i.table_id=t.table_id
   and i.index_name='GDC_PK'
   and t.table_name='Gdc'
) then
 drop index Gdc.GDC_PK
end if;
if exists(
 select 1 from sys.systable
 where table_name='Gdc'
   and table_type in ('BASE', 'GBL TEMP')
) then
  drop table Gdc
end if;
if exists(
  select 1 from sys.sysindex i, sys.systable t
 where i.table_id=t.table_id
   and i.index_name='TSS_PK'
   and t.table_name='Tss'
```





```
) then
 drop index Tss.TSS_PK
end if;
if exists(
 select 1 from sys.systable
 where table_name='Tss'
   and table_type in ('BASE', 'GBL TEMP')
) then
  drop table Tss
end if;
/* Table : Aavcs
                                            */
create table Aavcs
  numeroAavcs
                     integer
                                          not null,
 numeroAssure
                     integer
                                          not null,
 Tss_numeroAssure
                       integer
                                            not null,
  Gdc_numeroAssure
                        integer
                                             not null,
 dateEtablissement
                      timestamp
                                             null,
 dateDelivrance
                    timestamp
                                           null,
 constraint PK_AAVCS primary key (numeroAavcs),
 constraint AK_IDENTIFIANT_1_AAVCS unique (numeroAavcs)
);
/* Index : AAVCS PK
                                               */
```





	*/
*/	
	*/
	*/
*/	
	*/
	*/
*/	
	======*/
	-=====*/
*/	
	*/
	*/ */ */ */ */ */ */ */ */ */ */ */ */ *





numeroAssure	integer	not null,	
constraint PK_A	DMINISTRATEUR p	orimary key cluste	red (numeroAssure)
);			
/*======	:========		*/
/* Index : ADMINIS	STRATEUR_PK		*/
/*======	:========		*/
create unique clus	tered index ADMINI	STRATEUR_PK	on Administrateur (
numeroAssure AS	С		
);			
/*=======	:========		*/
/* Table : AssureS	ocial	*/	
/*=======	:========		*/
create table Assur	eSocial		
(
numeroAssure	integer	not null,	
Ass_numeroAss	ure integer	not null,	
noms	varchar(254)	null,	
prenoms	varchar(254)	null,	
adresse	varchar(254)	null,	
telephone	integer	null,	
constraint PK_A	SSURESOCIAL prir	mary key (numero	Assure)
);			
/*=======	:========		*/
/* Index : ASSURE	SOCIAL_PK		*/
/*=======	:=======		:=======*/
create unique inde	x ASSURESOCIAL	_PK on AssureSc	ocial (
numeroAssure AS	С		





```
);
                               */
/* Index : ASSOCIATION10_FK
create index ASSOCIATION10_FK on AssureSocial (
Ass numeroAssure ASC
);
/* Table : Compte
                           */
create table Compte
 idCompte
           integer
                        not null,
 numeroAssure
            integer
                         not null,
 typeCompte
            integer
                        null,
 "login"
         varchar(254)
                        null,
           varchar(254)
 password
                         null,
 constraint PK_COMPTE primary key (idCompte),
 constraint AK_IDENTIFIANT_1_COMPTE unique (idCompte),
 constraint AK_IDENTIFIANT_2_COMPTE unique (idCompte)
);
*/
/* Index : COMPTE PK
create unique index COMPTE_PK on Compte (
idCompte ASC
);
```





/*=====================================	===*/
/* Index : ASSOCIATION13_FK */ /*==================================	
create index ASSOCIATION13_FK on Compte (numeroAssure ASC);	·
/*=====================================	===*/
/* Table : Cotisation */	
/*====================================	:==*/
/*====================================	===*/
/*====================================	
/*====================================	
/*====================================	-== /





```
);
                             */
/* Table : Dossier
create table Dossier
 idDossier
            integer
                          not null,
 numeroAssure
              integer
                            not null,
 Tss_numeroAssure
               integer
                              null,
 dateEnvoie
             timestamp
                            null,
 statut
           char(1)
                         null,
 constraint PK_DOSSIER primary key (idDossier),
 constraint AK_IDENTIFIANT_1_DOSSIER unique (idDossier),
 constraint AK_IDENTIFIANT_2_DOSSIER unique (idDossier)
);
/* Index : DOSSIER PK
                                */
create unique index DOSSIER_PK on Dossier (
idDossier ASC
);
/* Index : ASSOCIATION2_FK
create index ASSOCIATION2_FK on Dossier (
numeroAssure ASC
);
```





/*=====================================		=======*/
/* Index : ASSOCIATION11_FK /*====================================	*/	
create index ASSOCIATION11_FK of		·
Tss_numeroAssure ASC	,	
);		
/*===========		*/
/* Table : Emploi	*/	
/*==========		-======*/
create table Emploi		
(
idEmploi integer	not null,	
numeroAssure integer	not null,	
nomEntreprise varchar(254)	null,	
raisonSociale varchar(254)	null,	
dateD_affiliation timestamp	null,	
constraint PK_EMPLOI primary ke	y (idEmploi),	
constraint AK_IDENTIFIANT_1_EI	MPLOI unique (idEmploi)	
);		
/*==========		-=====*/
/* Index : EMPLOI_PK	*/	
/*=====================================		=======*/
create unique index EMPLOI_PK on	Emploi (
idEmploi ASC		
);		
/*=====================================		-======*/
/* Index : ASSOCIATION4_FK	*/	





/*========	========		-======*/
create index ASSO	CIATION4_FK	on Emploi (
numeroAssure ASC			
);			
/*========	=======		=======*/
/* Table : Gdc		*/	
/*=======	=======		-======*/
create table Gdc			
(
	J	not null,	
	C primary key	clustered (numeroAssure)	
);			
/*			* .
	========	*/	-======= <i>"/</i>
/* Index : GDC_PK		/ ====================================	*/
create unique cluste			/
numeroAssure ASC		<u></u>	
);			
,,			
/*=======	=======	=======================================	-=====*/
/* Table : Tss		*/	
/*=======			-=====*/
create table Tss			
(
numeroAssure	integer	not null,	
constraint PK_TS	S primary key o	clustered (numeroAssure)	
);			





```
/* Index : TSS PK
                                        */
create unique clustered index TSS_PK on Tss (
numeroAssure ASC
);
alter table Aavcs
 add constraint FK_AAVCS_ASSOCIATI_TSS foreign key (Tss_numeroAssure)
   references Tss (numeroAssure)
   on update restrict
   on delete restrict;
alter table Aavcs
 add constraint FK_AAVCS_ASSOCIATI_ASSURESO foreign key (numeroAssure)
   references AssureSocial (numeroAssure)
   on update restrict
   on delete restrict;
alter table Aavcs
 add constraint FK_AAVCS_ASSOCIATI_GDC foreign key (Gdc_numeroAssure)
   references Gdc (numeroAssure)
   on update restrict
   on delete restrict;
alter table Administrateur
 add constraint FK_ADMINIST_GENERALIS_GDC foreign key (numeroAssure)
   references Gdc (numeroAssure)
   on update restrict
   on delete restrict;
```





```
alter table AssureSocial
 add constraint FK_ASSURESO_ASSOCIATI_ASSURESO foreign key
(Ass_numeroAssure)
   references AssureSocial (numeroAssure)
   on update restrict
   on delete restrict;
alter table Compte
  add constraint FK_COMPTE_ASSOCIATI_ASSURESO foreign key
(numeroAssure)
   references AssureSocial (numeroAssure)
   on update restrict
   on delete restrict:
alter table Cotisation
 add constraint FK_COTISATI_ASSOCIATI_ASSURESO foreign key
(numeroAssure)
   references AssureSocial (numeroAssure)
   on update restrict
   on delete restrict;
alter table Dossier
 add constraint FK_DOSSIER_ASSOCIATI_TSS foreign key (Tss_numeroAssure)
   references Tss (numeroAssure)
   on update restrict
   on delete restrict;
alter table Dossier
 add constraint FK_DOSSIER_ASSOCIATI_ASSURESO foreign key
(numeroAssure)
   references AssureSocial (numeroAssure)
```





```
on update restrict
   on delete restrict;
alter table Emploi
 add constraint FK_EMPLOI_ASSOCIATI_ASSURESO foreign key (numeroAssure)
   references AssureSocial (numeroAssure)
   on update restrict
   on delete restrict;
alter table Gdc
 add constraint FK_GDC_GENERALIS_TSS foreign key (numeroAssure)
   references Tss (numeroAssure)
   on update restrict
   on delete restrict;
alter table Tss
 add constraint FK_TSS_GENERALIS_ASSURESO foreign key (numeroAssure)
   references AssureSocial (numeroAssure)
   on update restrict
   on delete restrict;
```





IV- Présentation des diagrammes intervenants

1- Diagramme de composants

Le de diagramme de composant est principalement employé pour décrire les dépendances entre différents ou divers composants logiciels. Ce diagramme décrit les composants et leurs dépendances dans l'environnement de réalisation.

a- Formalisme

Les principaux éléments du diagramme de composants sont :

Le composant : qui représente une entité logicielle d'un système. Il est représenté par une boite rectangulaire avec deux rectangles dépassant du côté gauche.

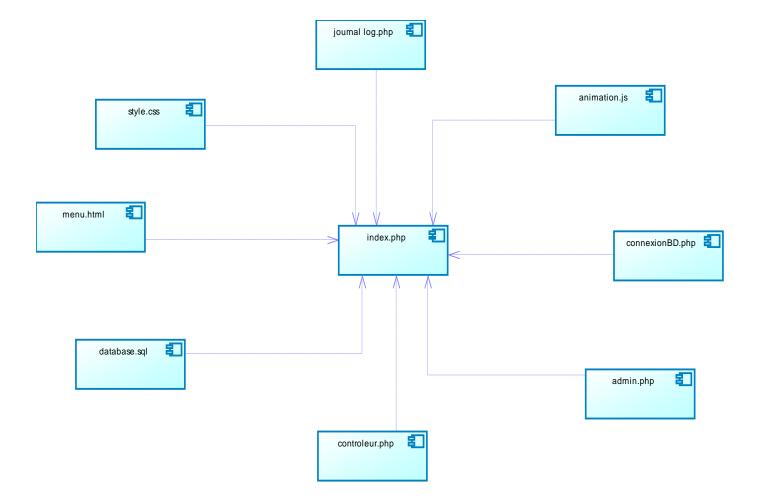
Composant_10

Les dépendances : elle est utilisée pour modéliser ou matérialiser la relation entre deux composants. L'annotation utilisé pour matérialiser la relation dépendance est la flèche en pointillée se dirigeant d'un composant donné vers le composant dont il dépend.





a- Diagramme de composants du système







2- Diagramme de déploiement

Un diagramme de déploiement est une vue statique qui sert à représenter l'utilisation de l'infrastructure physique par le système et la manière dont les composants du système sont répartis ainsi que leurs relations entre eux.

a- Formalisme

Les éléments utilisés par un diagramme de déploiement sont principalement les nœuds, les composants et les associations.

➤ Le nœud : il représente un ensemble d'éléments matériel du système. Il est représenté par un cube tridimensionnel contenant un nom ainsi que les composants associés.



Le composant : qui représente une entité logicielle d'un système. Sur un diagramme de déploiement, les composants sont placés dans des nœuds pour identifier ou spécifier l'endroit de leur déploiement.



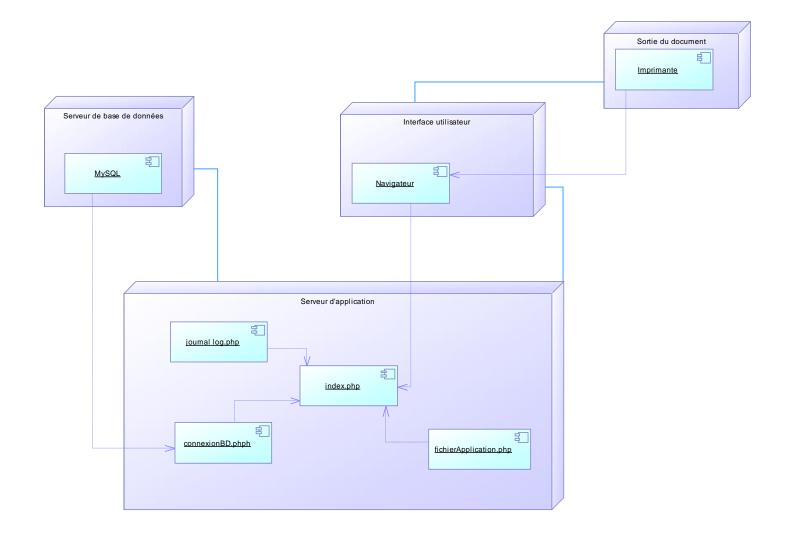
➤ L'association : elle représente une relation entre deux nœuds, elle est représentée par une ligne joignant les deux composants.

·····





b- Diagramme de déploiement du système







CONCLUSION

Le dossier de réalisation nous a permis de présenter de façon générale les outils utilisés pour la réalisation de notre application. Cependant, savoir comment l'installer et accéder à tous ses menus pour y effectuer toutes les tâches pour lesquelles elle a été conçue devient d'une importance capitale. Le prochain chapitre sera donc consacré à la présentation de l'utilisation de l'outil.





PARTIE V : GUIDE D'UTILISATEUR





Résumé:

Le présent document est un guide pour une utilisation aisée, rapide et fiable de l'application. Il sert de guide d'installation et de manuel d'utilisation pour tous les utilisateurs de l'application.

INTRODUCTION GUIDE D'INSTALLATION

II- MANUEL D'UTILISATION
CONCLUSION

I-





INTRODUCTION

Le guide d'utilisateur est un document établi après la réalisation de l'application et qui renseigne sur les différentes manières d'utiliser l'application. C'est un document important qui aide à maitriser le fonctionnement de l'application aisément de son installation jusqu'à l'utilisation des différents modules. Nous y présenterons d'abord le guide d'installation sur une machine grâce au web WampServer et enfin le manuel d'utilisation.





I- GUIDE D'INSTALLATION

1- Prérequis pour le fonctionnement de l'application

Pour déployer l'application, il faut un serveur de base de données sur un système d'exploitation (Windows, Linux, Mac OS...) et un navigateur (Mozilla Firefox, Google Chrome, Safari...).

- 2- Installation du serveur web : WampServer
- Configuration minimale :

Avoir un ordinateur ayant au moins 20Go de disque dur, un processeur de 1.5Ghz, et une RAM de 512Mo

Installation de WampServer

Pour installer cet outil, il faut double clic sur son setup, la page principale de l'installation s'ouvre, on doit installer le logiciel avec les paramètres par défauts, pour cela, il faudra faire un clic sur le bouton « suivant » jusqu'à la page ou figure le bouton « terminer ».