

Ministère des Enseignements Secondaire Supérieur
et de la Recherche Scientifique
(MESSR)

Université Polytechnique de Bobo-Dioulasso
(UPB)

Ecole Supérieure d'Informatique
(ESI)

01 BP 1091 Bobo-Dioulasso 01
Tel : 20 97 27 64
Email : esi@bf.refer.org

Cycle des Ingénieurs de Travaux Informatiques
Option : Analyse et Programmation

Burkina Faso
Unité-Progrès-Justice



Caisse Nationale de Sécurité Sociale
(CNSS)

01BP 562 Ouagadougou 01
Tel : 50 33 02 29
www.cnss.bf

Projet de fin de cycle

Année académique 2006-2007

THÈME :

*Gestion des prestations du Centre de
Production et de Formation en
Audiovisuel et Communication en
Entreprise*

Le groupe de projet :

*Abdou KABRÉ
Saïbou PARE*

Le Maître de stage :

*M. Vital DJIGMA
Informaticien à la Caisse
Nationale de Sécurité Sociale*

Le Superviseur :

*Dr. Michel SOME
Enseignant chercheur à
L'Ecole Supérieure d'Informatique*

Remerciements

Nous remercions :

Le Directeur Informatique et Statistique pour nous avoir octroyé ce stage combien important pour nous.

✚ Notre maitre de stage M. DJIGMA Vital informaticien à la CNSS : Nous avons toujours trouvé auprès de vous un accueil bienveillant. Dévouement, simplicité, amour du travail bien fait et esprit de compréhension sont autant de souvenirs que nous emportons de vous : Profonde reconnaissance ;

✚ Le responsable du CPFACE M. Raphael S. DAKISSAGA et à tout le personnel pour toutes les informations mises à notre disposition durant le stage:

✚ Tous les enseignants de l'Ecole Supérieure d'Informatique qui par une conscience professionnelle nous ont conduit, nous joignons à notre profonde reconnaissance l'espoir de demeurer digne d'eux.

✚ Tous nos Parents, Amis et Tuteurs pour leur soutien indéfectible ;

Que tous ceux qui d'une manière ou d'une autre ont contribué à la réussite de ce travail, trouvent dans ce présent rapport l'expression de notre profonde gratitude.

Sommaire

SOMMAIRE.....	3
LISTE DES ABBREVIATIONS	5
TABLE DES ILLUSTRATIONS	6
AVANT PROPOS.....	8
CHAPITRE 1 POSITIONNEMENT DU PROBLEME.....	9
1.1-PRESENTATION DES STRUCTURES	9
1.2-PRESENTATION DU PROBLEME.....	14
1.3-RESULTATS ATTENDUS.....	14
1.4-METHODES ET DEMARCHE D'ANALYSE	14
1.5-LA CONDUITE DU PROJET	21
CHAPITRE 2 ETUDE DE L'EXISTANT.....	22
2.1-CAPTURE DES BESOINS FONCTIONNELS.....	22
2.2-ANALYSE	25
2.3-BILAN CRITIQUE	45
CHAPITRE 3 ETUDE DES SCENARI II	46
3.1-OBJECTIF DU FUTUR SYSTEME D'INFORMATION	46
3.2-RECONFIGURATION DU SYSTEME D'INFORMATION.....	46
3.3-ETUDE COMPARATIVES DES OUTILS LOGICIELS ET LANGAGES.....	47
3.4-ARCHITECTURE RESEAU	56
3.5-MODELE DE CALCUL DES COUTS DE REALISATION	57
3.6-PRESENTATIONS DES SCENARI II.....	58
3.7-SCENARIO RETENU	62
CHAPITRE 4 ETUDE DU FUTUR SYSTEME D'INFORMATION.....	63
4.1-PHASE D'ELABORATION	63
4.2-CONCEPTION GENERALE.....	100
4.3-CONCEPTION DETAILLEE	109
4.4-PROCEDURES TRANSITOIRE.....	137
4.5-POLITIQUE DE SECURITE.....	137
4.6-PROCEDURE DE SECOURS	138
CHAPITRE 5 REALISATION	140
5.1-INFRASTRUCTURE APPLICATIVE	140
5.2-PRESENTATION DE QUELQUES MAQUETTES GRAPHIQUES.....	142
ANNEXES	145
1-ORGANIGRAMME DE LA CASSE NATIONALE DE SECURITE SOCIALE.....	145
2-DIAGRAMME DES CAS D'UTILISATION	146
3-DIAGRAMME DE SEQUENCE.....	148
4-DIAGRAMME DE CLASSE	152

5-DIAGRAMME D'ACTIVITES	155
6-DIAGRAMME DE DEPLOIEMENT	157
CONCLUSION.....	158
BIBLIOGRAPHIE	159

Liste des abréviations

CNSS : Caisse Nationale de Sécurité Sociale ;
IAPRP : Interafricaine de la Prévention des Risques Professionnels ;
S.IAPRP : Structure d'un pays membre de l'IAPRP
CPFACE : Centre de Production et de Formation en Audiovisuel et Communication en Entreprise ;
SYAPRO : Symposium sur l'Audiovisuel et la Prévention des Risques professionnels de Ouagadougou ;
SST : Santé et Sécurité au Travail ;
FESPACO : Festival Panafricain du Cinéma et de la télévision de Ouagadougou ;
DPASS : Direction de Prévention de l'Action Sanitaire et Sociale ;
DG : Direction Générale ;
CA : Conseil d'Administration ;
DIS : Direction Informatique et Statistique
DBA : DataBase Administration
INRS : Institut National de Recherche et de Sécurité ;
R.CPFACE : Responsable CPFACE ;
CGM : Chargé de Gestion de la médiathèque ;
AS : Administrateur Système
CPF : Caisse de Compensation des Prestations Familiales ;
OMT : Object Modeling Technic
UML : Unified Modeling Language ;
UP : Unified Process ;
HTML :Hyper Text Mark up Language
CU : Cas d'Utilisation

Table des illustrations

Tableau 1.1 Parc informatique du CPFACE	13
Tableau 1.2 Plan prévisionnel.....	21
Tableau 3.1 comparatif MySQL/PostgreSQL/Oracle.....	47
Tableau 3.2 Comparatif JBuilder/Eclipse/NetBeans	53
Tableau 3. 3 les coûts du modèle de base	58
Tableau 3. 4 besoins matériels et logiciels du premier scénario	59
Tableau 3. 5 Coût de la formation du premier scénario	59
Tableau 3. 6 Coût total de la mise en œuvre.....	59
Tableau 3.7 besoins matériels et logiciels du premier scénario	61
Tableau 3.8 coût de la formation du premier scénario	61
Tableau 3.9 coût total de la mise en œuvre.....	61
Tableau 3.10 résumé comparatif des caractéristiques des scénarii	62
Tableau 4.1Comparatif JDBC / Hibernate / EJBs	100
Tableau 4.2 Classes importantes de l'API JDBC.....	101
Tableau 4.3 Comparatif Struts /JSF.....	106
Figure 3.1 les symboles utilisés.....	56
Figure3.2 Ressources existantes	56
Figure 3.3 Architecture réseau du premier scénario	58
Figure 3.4 Architecture réseau du deuxième scénario.	60
Figure 4.1Mapping avec Hibernate.....	103
Figure 4.2Environnement d'exécution d'EJB	104
Figure 4. 3 Cycle de vie d'une requête http.	107
Figure5. 1Environnement de travail de Sybase Power AMC	140
Figure 5.2 Environnement de développement NetBeans 5.5.1.....	141
Figure5.3 Environnement web : Serveur web Tomcat	142
Diagramme 2.1 Diagramme de cas d'utilisation du système existant	25
Diagramme2.2 Diagramme de séquence décrivant l'inscription au centre	35
Diagramme2.3 Diagramme de séquence décrivant la collection d'une œuvre.....	36
Diagramme2.4 Diagramme de séquence décrivant la procédure d'emprunt d'une œuvre	37
Diagramme2.5 Diagramme de séquence décrivant la procédure de production d'une œuvre.....	38
Diagramme2.6 Diagramme de séquence décrivant la procédure de coproduction d'une œuvre	39
Diagramme2.7 Diagramme de séquence décrivant la procédure de choix d'un thème.....	40
Diagramme2.8 Diagramme de séquence décrivant la procédure d'établissement d'un programme d'activité du SYAPRO	41
Diagramme2.9 Diagramme de séquence décrivant la procédure d'organisation d'un prix « SST »	43
Diagramme2.10 Diagramme de séquence décrivant la procédure de gestion de la formation continue.....	44
Diagramme4.1 Diagramme de cas d'utilisation du futur système.....	63
Diagramme4.2 Diagramme de séquence décrivant la procédure d'authentification.....	81
Diagramme4.3 Diagramme de séquence décrivant la procédure de Gestion des informations sur le prix « SST »	81
Diagramme4.4 Diagramme de séquence décrivant la procédure de suppression ou de modification d'information sur le prix « SST »	82
Diagramme4.5 Diagramme de séquence décrivant la procédure d'affichage d'informations sur le prix « SST »	83
Diagramme4.6 Diagramme de séquence décrivant la procédure d'enregistrement d'un nouveau partenaire	84
Diagramme4.7 Diagramme de séquence décrivant la procédure de suppression/ modification d'information sur un partenaire	85
Diagramme4.8 Diagramme de séquence décrivant la procédure de consultation d'informations sur un partenaire	86
Diagramme4.9 Diagramme de séquence décrivant la procédure d'enregistrement d'information sur une édition du SYAPRO	87

Diagramme4.10 Diagramme de séquence décrivant la procédure de Modification/Suppression des informations d'une édition	88
Diagramme4.11 Diagramme de séquence décrivant la procédure de consultation d'une formation sur une édition du SYAPRO	89
Diagramme4.12 Diagramme de séquence décrivant la procédure d'inscription en ligne	90
Diagramme4.13 Diagramme de séquence décrivant la procédure d'enregistrement d'une œuvre.....	91
Diagramme4.14 Diagramme de séquence décrivant la procédure de suppression/modification des informations descriptives d'une œuvre.....	92
Diagramme4.15 Diagramme de séquence décrivant la procédure de consultation d'une œuvre.....	93
Diagramme4.16 Diagramme de séquence décrivant la procédure d'emprunt d'une œuvre	94
Diagramme4.17 Diagramme de séquence décrivant la procédure d'envoi d'un message	95
Diagramme4.18 Diagramme de séquence décrivant la procédure de gestion de la formation continue.....	96
Diagramme4.19 Diagramme de séquence décrivant la procédure de gestion des notes.....	97
Diagramme4.20 Diagramme de séquence décrivant la procédure de gestion des enseignants.....	98
Diagramme4.21 Diagramme de séquence décrivant la procédure de gestion des étudiants.....	101
Diagramme4.22 Diagramme de paquetage du futur système4.3-Conception détaillée	108
Diagramme4.23 Diagramme de classe du paquetage Médiathèque.....	109
Diagramme4.24 Diagramme de classe du paquetage Production	110
Diagramme4.25 Diagramme de classe du paquetage SYAPRO	112
Diagramme4.26 Diagramme de classe du paquetage Formation.....	114
Diagramme4.26 Diagramme de classe du paquetage Formation prix « Santé et sécurité u Travail »	115
Diagramme4.27 Diagramme d'activité décrivant la procédure d'authentification.....	128
Diagramme4.28 Diagramme d'activité décrivant la procédure d'inscription en ligne	129
Diagramme4.29 Diagramme d'activité décrivant la procédure d'envoi de message	130
Diagramme4.30 Diagramme d'activité décrivant la procédure de gestion de la médiathèque	131
Diagramme4.31 Diagramme d'activité décrivant la procédure d'emprunt des œuvres	132
Diagramme4.32 Diagramme d'activité décrivant la procédure de gestion des archives du SYAPRO.....	133
Diagramme4.33 Diagramme d'activité décrivant la procédure de gestion des archives du prix «santé et sécurité au travail » du FESPACO	134
Diagramme4.34 Diagramme d'activité décrivant la procédure de gestion des formations	135
Diagramme4.35 Diagramme d'activité décrivant la procédure de gestion des partenaires	136
Maquette 5.1 la vue de la page d'accueil.....	142
Maquette 5.2 la vue de la page médiathèque	143
Maquette 5.3 la vue de la liste des films par thème	143
Maquette 5.4 la vue de la page de visualisation d'une séquence vidéo.....	144

Avant propos

L'informatisation est le phénomène le plus important de notre époque, elle s'immisce dans la plus part des objets de la vie courante et ce, que ce soit dans l'objet proprement dit ou bien dans le processus de conception ou de fabrication de cet objet. Actuellement elle est au cœur des entreprises et de défis nouveaux se présentent. Pour outiller les futurs diplômés de compétences à relever ces défis, il est institué à l'Ecole Supérieure d'Informatique (ESI) dans le cursus de formation des ingénieurs de travaux informatiques (option Analyse et programmation) deux stages pratiques en entreprise.

Le premier stage se déroulant en fin de deuxième année est axé sur la programmation et a pour objectif d'initier les étudiants aux réalités professionnelles.

Le second quant à lui tend à garantir aux futurs employeurs des diplômés une intégration rapide et efficace en milieu professionnel. En effet, l'occasion est donnée aux étudiants de traiter d'une problématique dans le domaine de gestion en y apportant des solutions informatiques. Ce stage, d'une durée de trois (03) mois, fera l'objet d'une soutenance publique. Il constitue pour les étudiants le cadre idéal d'application puis d'évaluation des différentes connaissances acquises tout au long du cycle.

C'est dans cette perspective que nous avons été reçus à la Caisse Nationale de Sécurité Sociale (CNSS) où il nous a été soumis la problématique de la **Gestion des prestation du Centre de Production et Formation en Audiovisuel et Communication en Entreprise (CPFACE)**.

Le présent document comprend cinq (5) chapitres. Le premier présente la structure d'accueil, le problème dans toutes ses dimensions et définit la démarche d'analyse et langage adoptés. Le deuxième chapitre quant à lui décrit le fonctionnement actuel du système d'information tout en présentant ses forces et dysfonctionnements. Dans le troisième chapitre il est question de reconfigurer le système d'information actuel et de déterminer des scénarii pour la réalisation du futur système d'information. Le chapitre quatre aborde la conception générale et détaillée du futur système d'information, les procédures transitoires, la politique de sécurité et la procédure de secours à adopter. Le chapitre cinq présente l'environnement de développement et fournit une description des fonctionnalités de l'application à travers quelques maquettes.

Chapitre 1 Positionnement du problème

Ce chapitre pose le problème de la Gestion des prestations du CPFACE et permet d'avoir une idée du fonctionnement actuel du CPFACE. Il présente aussi la méthode d'analyse et de conception considérée pour l'étude.

1.1-Présentation des structures

1.1.1- Structures d'accueil

1.1.1.1- Historique et domaines d'activités

L'institution publique de l'Etat qu'est la Caisse Nationale de Sécurité Sociale (CNSS) telle que connue aujourd'hui provient de plusieurs restructurations et d'une nouvelle dénomination de l'ex Caisse de Compensation des Prestations Familiales (CPF). En effet, suite à l'application du Code du travail d'Outre-mer, une petite caisse dite de compensation des prestations familiales fut créée le 6 décembre 1955 et installée au départ à Bobo-Dioulasso avec une vingtaine d'agents. Cette caisse passera par plusieurs étapes de restructuration pour donner la CNSS. C'est ainsi que, ayant à l'origine pour mission de ne s'occuper que d'un seul domaine de la sécurité sociale – à savoir celui de la prestation familiale, elle verra son champ d'action élargi au traitement des accidents de travail et des maladies professionnelles deux ans plus tard, puis à celui du traitement des pensions en 1960. Le siège fut transféré à Ouagadougou le 18 octobre 1963 et la loi N° 13-72 du 28-12-1972 viendra créer la Caisse Nationale de Sécurité Sociale en même temps qu'elle refondait les textes pour mieux satisfaire les nouveaux besoins d'efficacité et de rapprochement des assurés sociaux de leur structure.

La CNSS a pour mission de gérer le régime de sécurité sociale institué au Burkina Faso en faveur des travailleurs déclarés. Un régime de sécurité sociale est un système de protection sociale obligatoire institué par l'Etat dans le but de protéger le travailleur et sa famille contre les risques pouvant provenir du travail, de la maladie, du décès, ou de la vieillesse.

Le régime de sécurité sociale du Burkina Faso comprend 3 branches, à savoir :

- La branche des prestations familiales, chargée du service des prestations familiales et des prestations de maternité.
- La branche des risques professionnels, chargée des prestations en cas d'accident du travail ou de maladies professionnelles.
- La branche des pensions, chargée du service des prestations de vieillesse, d'invalidité et de décès.

1.1.1.2-Structure et organisation¹

Aujourd'hui, avec plus de neuf cent (900) agents répartis sur l'ensemble du territoire national, la CNSS est dotée d'une direction générale, d'un secrétariat général, de cinq (5) directions régionales (dont les sièges sont à Bobo-Dioulasso, Ouagadougou, Ouahigouya, Dédougou, et Fada N'Gourma), de neuf (9) services provinciaux et de quatorze (14) guichets de paiement dans la plupart des localités. La CNSS est placée sous la tutelle de trois ministères :

¹ L'organigramme de la CNSS est représenté en annexes partie 1

- le ministère du travail et de la sécurité sociale, qui assure la tutelle technique.
- Le ministère du commerce, de la promotion de l'industrie et de l'artisanat, qui assure la tutelle gestion.

Les directions régionales, au nombre de 5, sont chargées d'assurer la gestion des services communs décentralisés (recouvrer les cotisations, assurer les services financiers et comptables, l'action sanitaire et sociale, payer les prestations...), de gérer les ressources humaines et le patrimoine mobilier et immobilier, d'assurer la défense des intérêts de CNSS dans la limite de leur ressort territorial.

La gestion de la Caisse Nationale de Sécurité Sociale est confiée à un Conseil d'Administration tripartite comprenant des représentants des employeurs, des représentants des travailleurs, des représentants de l'Etat. Y siègent également des représentants de l'Association des Retraités Burkinabé.

1.1.1.3- Environnement informatique : la Direction Informatique et Statistique (DIS)

La Direction Informatique et Statistique assure tous les travaux relevant du domaine informatique aussi bien que du domaine des statistiques, et comprend trois structures, à savoir :

- La cellule DBA (Data Base Administration) et système d'information
- Le service informatique
- Le service statistique.

Le service statistique a pour rôle la collecte des données primaires, l'élaboration de l'annuaire statistique, l'analyse des données statistiques et toute étude d'ordre statistique.

Le domaine informatique est assuré par :

- La cellule DBA et système d'information : Elle s'occupe de l'administration de la base de données, participe à la conception des schémas directeurs informatiques, travaille à la conception et supervise la mise en œuvre du système d'information. La base de données de la Caisse Nationale de Sécurité Sociale est gérée par le système de gestion de base de données (SGBD) Oracle 9.i avec comme outils de développement Oracle Designer et Oracle Developer.
- Le service informatique : composé aussi bien d'informaticiens analystes-programmeurs que d'informaticiens réseau ou professionnels de la maintenance. Il est chargé de développer les nouvelles applications et d'assurer leur maintenance, de participer à la formation et à la sensibilisation du personnel sur le plan informatique, d'exploiter les applications sur les ordinateurs de la CNSS, de veiller à la sécurité et à la confidentialité des informations traitées. C'est également ce service qui veille au bon fonctionnement des équipements informatiques, et de tout l'environnement informatique. Il assure par ailleurs l'administration et le bon fonctionnement des réseaux locaux des différentes directions et de tout le réseau national de la CNSS.

Aussi, c'est ce service qui nous a accueilli pour le stage, afin de nous permettre de développer notre thème d'étude qui n'est autre que **la gestion des prestations du CPFACE**.

• Présentation de l'environnement matériel

Au niveau de l'environnement matériel nous avons :

- **Les postes de travail.** La CNSS compte en son sein environ trois cent (300) micro-ordinateurs équipés pour la plupart de processeurs de type Pentium (Pentium II à Pentium IV), de marque COMPAQ et HEWLETT PACKARD (HP) essentiellement. La vitesse des processeurs de ces ordinateurs se situe entre 133 MHz et 2,99 GHz et la capacité de leurs disques varie entre 3 Go et 80 Go. Ces postes de travail sont munis de 32 Mo à 512 Mo de mémoire vive.
- **Les serveurs.** La CNSS dispose en son sein de huit (8) serveurs, dont les caractéristiques sont les suivantes :

Serveur 1	Type de serveur Marque et modèle Système d'exploitation Mémoire RAM Processeurs Disques durs	Serveur de données. SGBD Oracle 9i ALPHA DS20E HP TRU64 UNIX V5.B 4 GB 2 Alpha 64 bits de 2 GHz et 4 MB cache chacun 06 disques ultra SCSI de 72,3 GB chacun
Serveur 2	Type de serveur Marque et modèle Système d'exploitation Mémoire RAM Processeurs Disques durs	Serveur de données. SGBD Oracle 9i ALPHA DS25 HP TRU64 UNIX V5.B 4 GB 2 Alpha 64 bits de 2 GHz et 4 MB cache chacun 06 disques ultra SCSI de 72,3 GB chacun
Serveur 3	Type de serveur Marque et modèle Système d'exploitation Mémoire RAM Processeur Disques durs	Serveur de données. SGBD Oracle 9i ALPHA DS15L HP TRU64 UNIX V5.B 2 GB 1 Alpha 64 bits de 2 GHz et 4 MB cache 02 disques ultra SCSI de 72,3 GB chacun
Serveur 4	Type de serveur Marque et modèle Système d'exploitation Mémoire RAM Processeur Disques durs	Serveur de données. SGBD Oracle 9i ALPHA DS15L HP TRU64 UNIX V5.B 2 GB 1 Alpha 64 bits de 2 GHz et 4 MB cache 02 disques ultra SCSI de 72,3 GB chacun
Serveur 5	Type de serveur Marque et modèle Système d'exploitation Mémoire RAM Processeur Disques durs	Serveur de données. SGBD Oracle 9i ALPHA DS15L HP TRU64 UNIX V5.B 2 GB 1 Alpha 64 bits de 2 GHz et 4 MB cache 02 disques ultra SCSI de 72,3 GB chacun
Serveur 6	Type de serveur Marque et modèle Système d'exploitation Mémoire RAM Processeurs Disques durs	Serveur de données. SGBD Oracle 9i ALPHA ES40 HP TRU64 UNIX V5.B 1 GB 2 Alpha 64 bits de 667 MHz 02 disques ultra SCSI de 72,3 GB chacun
Serveur 7	Type de serveur Marque et modèle Système d'exploitation Mémoire RAM Processeur Disques durs	Serveur de données. SGBD Oracle 9i ALPHA SERVER 2100 DIGITAL UNIX 2 GB 1 processeur de 833 MHz 10 disques ultra SCSI de 8 GB chacun

Serveur 8	Type de serveur	Serveur de développement. SGBD Oracle 9i
	Marque et modèle	ALPHA SERVER 1100
	Système d'exploitation	DIGITAL UNIX
	Mémoire RAM	512 MB
	Processeur	1 processeur de 833 MHz
	Disques durs	06 disques ultra SCSI de 8 GB chacun

– **Les imprimantes.** La grande majorité des imprimantes utilisées à la CNSS sont de marque HP, et bon nombre d'ordinateurs en sont équipés. Outre ces imprimantes connectées aux différents postes de travail, il existe des imprimantes réseaux utilisées par le service statistique et le service de la comptabilité. Celles-ci ont de grandes vitesses d'impression permettant ainsi aux agents de ces différents services d'imprimer très rapidement leurs listings. Ces imprimantes réseaux sont surtout de marque EPSON, DIGITAL et GENICOM.

– **Les onduleurs.** Pour permettre à ses équipements informatiques et électroniques de façon générale, de fonctionner dans un environnement stable, la CNSS s'est doté de plusieurs onduleurs, dont un onduleur central de marque LIEBERT, de puissance 80 KVA et de type online, avec une autonomie d'environ 30 minutes. Il fonctionne rarement en mode batterie car en cas de coupure du courant secteur, un groupe électrogène prend la relève.

Cet onduleur central, de même que les différents serveurs ci-dessus cités, sont bien gardés dans une salle machine à air conditionné, dotée d'un important dispositif de sécurité, et à accès très limité.

• Présentation de l'environnement logiciel

L'environnement logiciel est constitué essentiellement par les systèmes d'exploitation WINDOWS et UNIX. Windows est installé sur les postes de travail et on retrouve presque toute la gamme des produits Microsoft (Windows 95, Windows 98, Windows NT, Windows 2000, Windows XP) excepté les versions trop anciennes (Windows 3.X). Les produits Microsoft Office installés vont de la version 97 à la version 2003. Les serveurs quant à eux tournent tous sous Unix, les versions HP TRU64 UNIX V5.B et DIGITAL UNIX.

Pour gérer son importante base de données, la CNSS a choisi le système de gestion de bases de données (SGBD) Oracle 9.i avec pour outils de développement Oracle Designer et Oracle Developer.

Outre ces différents logiciels, il y'a les applications métiers, qui sont des applications spécialement développées par le service informatique dans le cadre des activités menées au sein de la structure. Ces applications métier constituent le principal outil de travail des agents, car elles sont celles qui leur permettent de pouvoir mener les différentes prestations.

• Présentation de l'environnement réseau

Pour une meilleure exploitation de ses ressources, la Caisse Nationale de Sécurité Sociale a mis ses équipements informatiques en réseau de type client-serveur, architecture 2-tiers. Au niveau du réseau local de son siège, elle dispose de huit (8) serveurs (1 serveur de développement et 7 serveurs de base de données) tournant tous sous le système d'exploitation UNIX, les versions HP TRU64 UNIX V5.B et DIGITAL UNIX. Mais cette architecture client-serveur à deux niveaux est en voie de modification pour passer à l'architecture 3-tiers. La différence entre ces deux types d'architecture est qu'au niveau de l'architecture 2-tiers (qui est l'architecture client-serveur classique), le client demande une ressource et le serveur la lui fournit directement, sans faire appel à une autre application afin de fournir le service.

Au niveau de l'architecture 3-tiers par contre, il existe un troisième niveau qui n'existe pas au niveau de l'architecture 2-tiers. Nous avons :

Le client qui est le demandeur de ressources

Le serveur d'application (aussi appelé middleware) qui est chargé de fournir la ressource, mais en faisant appel à un autre serveur

Le serveur secondaire (généralement un serveur de base de données) fournissant un service au premier serveur.

1.1.2-Le Centre de production et de Formation en Audiovisuel et Communication en Entreprise (CPFACE)

1.1.2.1-Les Missions du CPFACE

Le CPFACE, créé par la CNSS à la demande des institutions membres de l'Interafricaine de la Prévention des Risques Professionnels(IAPRP) c'est-à-dire les 18 pays francophones d'Afrique est une structure de formation aux métiers de l'audiovisuel et de la communication et de production des supports audiovisuels d'information et de sensibilisation à la prévention des risques professionnels. Pour assurer efficacement les tâches qui lui ont été assignées il s'est fixé pour objectif :

La promotion de la coproduction des supports audiovisuels de sensibilisation entre institutions membres de l'Interafricaine de Prévention des Risques Professionnels

La formation de techniciens et des acteurs de la santé et sécurité au travail en audiovisuel et communication en entreprise.

Au delà de la production des supports audiovisuels et de la promotion des coproductions entre structures de santé et de sécurité au travail, le CPFACE a pour autre mission de constituer une « banque de données » pour la collecte et la diffusion des productions audiovisuelles de la sous-région.

C'est dans ce sens qu'il faut comprendre le rôle combien important que va jouer la Bibliothèque/Médiathèque du CPFACE en tant que point repère de l'IAPRP dans le domaine du recueil et de l'archivage des productions audiovisuelles

1.1.2.2-Ressource informatiques du CPFACE

Les ressources informatiques existantes du CPFACE sont consignées dans le tableau suivant :

Poste de travail	matériel	Logiciel	Réseau
Bureau du responsable CPFACE	-un micro-ordinateur PIV -une imprimante HP -un onduleur	XP et Office 2003	Néant
Bureau du coordonnateur	-un micro-ordinateur PIII		
Secrétariat	-un micro-ordinateur PIV -une imprimante HP -une photocopieuse		

Tableau 1.1Parc informatique du CPFACE

1.2-Présentation du problème

Le CPFACE a été institué par la caisse à la demande des institutions des pays membres de l'IAPRP pour la promotion des supports audiovisuels de sensibilisation des risques professionnels et la formation des agents en audiovisuel et communication en entreprise. Le CPFACE a une vocation sous régionale, cependant au stade actuel il ne dispose pas d'une grande visibilité au sein des Etats membres de l'IAPRP.

De plus, le Symposium sur l'Audiovisuel et la Prévention des Risques Professionnels de Ouagadougou (SYAPRO) qui se tient tous les trois (3) ans est un pur produit du CPFACE et toute la documentation sur cette activité régionale qui intéresse tout le monde et de plus en plus d'étudiants chercheurs n'est pas mise à la disposition du public. Actuellement la gestion et le suivi de cette activité sont réalisés sur la base de rapports, ce qui rend difficile la recherche d'information.

Aussi, il a été confié au CPFACE lors du dernier SYAPRO la gestion de la médiathèque africaine en matière de films et multimédia sur la prévention des risques professionnels et il doit alors tenir à jour et publier la liste des œuvres disponible dans la sous région.

Le CPFACE gère également le prix « Santé et Sécurité au Travail » du FESPACO, une autre activité de l'IAPRP, ce qui nécessite un suivi des différents lauréats et la publication de leurs œuvres collectionnées à cet effet.

Par ailleurs, le CPFACE en tant que structure à vocation régionale a besoins de coller au plus près aux attentes de ses clients et les inscriptions pour la formation au stade actuel se font par voie de courrier avec la lenteur liées à ce mode de contact.

1.3-Résultats attendus

Le système à mettre en place devra permettre de résoudre les problèmes soulevés dans la gestion des activités assignées au CPFACE. Pour cela le système doit offrir principalement les fonctionnalités suivantes :

- la mise à la disposition de toutes les informations sur le CPFACE ;
- la possibilité d'échange interactif entre le CPFACE et ses partenaires ;
- la possibilité d'inscriptions en ligne pour la formation en audiovisuel et communication en entreprise ;
- la gestion de la médiathèque africaine en matière de films et de multimédia sur la prévention des risques professionnels ;
- la mise à la disposition de toute la documentation sur le SYAPRO et du prix « Santé et Sécurité au Travail » du FESPACO qui intéresse tout le monde et de plus en plus d'étudiants chercheurs.

1.4-Méthodes et démarche d'analyse

Modéliser un système avant sa réalisation permet de mieux comprendre le fonctionnement du système. C'est également un bon moyen de maîtriser sa complexité et d'assurer sa cohérence.

La construction de modèles² (appelée modélisation) a quatre objectifs principaux :

- les modèles aident à visualiser un système tel qu'il est ou tel que nous voudrions qu'il soit
- les modèles permettent de préciser la structure ou le comportement d'un système ;
- les modèles fournissent un canevas qui guide la construction d'un système ;
- les modèles permettent de documenter les décisions prises.

Les méthodes d'analyse et de conception quant à elles fournissent une méthodologie et des notations standards (modèles) qui aident à concevoir des logiciels de qualité. On note principalement deux approches : l'approche fonctionnelle et l'approche orientée objet.

Les méthodes fonctionnelles (structurées) qui trouvent leur origine dans les langages procéduraux mettent en évidence les fonctions à assurer et proposent une approche hiérarchique descendante et modulaire. L'approche orientée objet en revanche considère le logiciel comme une collection d'objet dissocié, et identifié, définis par des propriétés.

En approche objet, l'évolution des besoins se présente comme un changement de l'interaction des objets. Ainsi pour apporter une modification aux données seul l'objet incriminé sera modifié. Dans une approche structurée, l'évolution des besoins entraîne souvent une dégénérescence, ou une profonde remise en question même de la structure du logiciel. L'approche objet rapproche les données et les traitements et la structure d'un logiciel issue d'une approche objet est beaucoup plus malléable, adaptable que celle issue d'une approche structurée. Tels sont les raisons de l'adoption d'une approche objet même si l'approche objet n'avait jamais été expérimentée à la CNSS.

1.4.1-Le langage UML

1.4.2.1-Présentation

Dans le cadre de la conception orientée objet, un langage unifié pour la modélisation a été développé : UML (« Unified Modeling Language »). Il s'agit d'un langage graphique de modélisation objet permettant de spécifier, de construire, de visualiser et de décrire les détails d'un système logiciel. Il est issu de la fusion de plusieurs méthodes dont « Booch » et « OMT » et est adapté à la modélisation de tous types de systèmes. La modélisation d'un système s'effectue indépendamment de toute méthode ou de tout langage de programmation.

UML est un langage : il comprend un vocabulaire et un ensemble de règles centrés sur la représentation conceptuelle et physique d'un système logiciel. Ses domaines d'utilisation sont :

- visualisation d'un système ;
- spécification d'un système ;
- construction d'un système ;
- documentation d'un système.

1.4.2.1- Le modèle conceptuel d'UML

Le modèle conceptuel d'UML comprend les notions de base génériques du langage. Il définit trois sortes de « briques » de base :

- les éléments, qui sont les abstractions essentielles à un modèle ;
- les relations, qui constituent des liens entre ces éléments ;

² Modèle: simplification de la réalité permettant de mieux comprendre le système que l'on développe.

- les diagrammes, qui regroupent des éléments et des liens au sein de divers ensembles.

1.4.2.1.1 -Les éléments

Il existe quatre types d'éléments dans UML :

- les éléments structurels (classe, interface, collaboration,...) ;
- les éléments comportementaux (interaction, automate à états finis) ;
- les éléments de regroupement (package) ;
- les éléments d'annotation (note).

1.4.2.1.2-Les relations

Il existe quatre types de relations dans UML :

- la dépendance ;
- l'association ;
- la généralisation ;
- la réalisation.

1.4.2.1.3-Les diagrammes³

UML2.0 comporte treize types de diagrammes représentant autant de vues distinctes pour représenter des concepts particuliers du système d'information. Ils se répartissent en deux grands groupes :

- Diagrammes structurels ou diagrammes statique (UML structure)

Diagramme de classe (Class Diagram) : exprime de manière générale la structure statique d'un système, en termes de classes et de relations entre ces classes.

Diagramme d'objets (Object Diagram) : permet de représenter les relations existant entre les différentes instances des classes à un instant donné de la vie du système.

Diagramme de composants (Component Diagram) : permet de décrire les composants et leurs dépendances dans leur environnement d'implémentation.

Diagramme de déploiement (Deployment Diagram) : permet de montrer la disposition physique des matériels qui composent le système, ainsi que la répartition des composants sur ces matériels représentés par des nœuds.

Diagramme de paquetages (Package Diagram) : permet de définir des sous-systèmes formés d'éléments ayant entre eux une certaine logique.

³ Diagramme: représentation graphique d'un ensemble d'éléments et de relations qui constituent un système.

Diagramme de structures composites (Composite Structure Diagram) : pour explorer les instances des classificateurs collaborant à travers des liens de communication.

- **Diagrammes comportementaux ou diagramme dynamiques (UML Behaviour)**

Diagramme de cas d'utilisation (Use Case Diagram) : représente la structure des grande fonctionnalités nécessaires aux utilisateurs du système.

Diagramme d'activités (Activity Diagram) : c'est une forme spatiale du diagramme d'état-transition utilisé pour modéliser l'état d'un contrôle.

Diagramme d'états transitions (State Machine Diagram) : permet de décrire les changements d'états d'un objet ou d'un composant, en réponse aux interactions avec d'autres objets, composants ou acteurs.

Diagrammes d'interactions (Interaction Diagram) :

Diagramme de séquences (Sequence Diagram) : représente la succession chronologique des opérations réalisées par un acteur. Il indique les objets que l'acteur va manipuler et les opérations qui font passer d'un objet à l'autre.

Diagramme de communication/collaboration (Communication Diagram) : montrent des interactions entre des objets qui peuvent être des instances de classes ou des acteurs.

Diagramme global d'interaction (Interaction Overview Diagram) : est une variante du diagramme d'activité qui donne une vue globale d'un flot de contrôle.

Diagramme de temps (Timing Diagram) : utilisé pour explorer le comportement d'un ou plusieurs objets pendant une période donnée

1.4.2.2-Avantages et limites d'UML

Plusieurs raisons conduisent à l'utilisation d'UML. Il présente l'avantage d'être un langage formel et normalisé ce qui procure un gain de précision et constitue de ce fait un gage de stabilité. UML est un support de communication performant puisqu'il cadre l'analyse et facilite la compréhension de représentations abstraites complexes. Son caractère polyvalent et sa souplesse en font un langage universel. Notons néanmoins que la mise en pratique d'UML nécessite un apprentissage et passe par une période d'adaptation. En effet le processus de développement (non couvert par UML) est une autre clé de la réussite d'un projet. Or, l'intégration d'UML dans un processus n'est pas triviale et améliorer un processus est une tâche complexe et longue.

1.4.3-Le Processus Unifié (UP)

Le processus unifié est un processus de développement logiciel entièrement basé sur UML : il regroupe les activités à mener pour transformer les besoins d'un utilisateur en système logiciel. C'est un patron de processus pouvant être adaptés à une large classe de systèmes logiciels, différents domaines d'application, à différents types d'entreprises, à différents niveaux de compétences et à différentes tailles de l'entreprise.

1.4.3.1-Caractéristiques du processus unifié

- **Le processus unifié est piloté par les cas d'utilisation**

L'objectif principal d'un système logiciel est de rendre service à ses utilisateurs ; il faut par conséquent bien comprendre les désirs et les besoins des futurs utilisateurs. Le processus de développement sera donc centré sur l'utilisateur. Le terme utilisateur ne désigne pas seulement les utilisateurs humains mais également les autres systèmes. L'utilisateur représente donc une personne ou une chose dialoguant avec le système en cours de développement. Les cas d'utilisation font apparaître les besoins fonctionnels et leur ensemble constitue le modèle des cas d'utilisation qui décrit les fonctionnalités complètes du système.

Les cas d'utilisation ne sont pas un simple outil de spécification des besoins du système. Ils vont complètement guider le processus de développement à travers l'utilisation de modèles basés sur l'utilisation du langage UML.

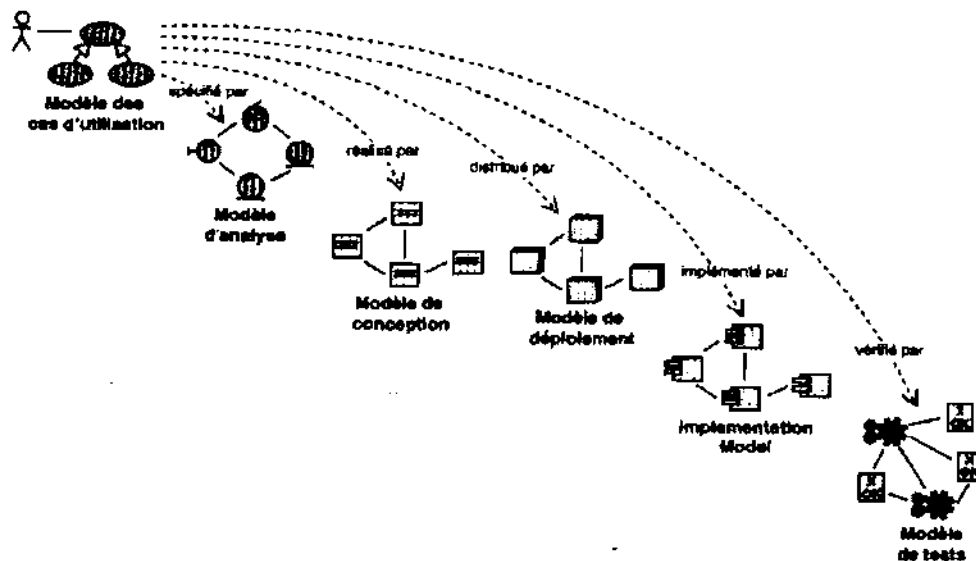


Figure 1.1 Stratégie des cas d'utilisation

- **Le processus unifié est centré sur l'architecture**

Dès le démarrage du processus, on aura une vue sur l'architecture à mettre en place.

L'architecture d'un système logiciel peut être décrite comme les différentes vues du système qui doit être construit. L'architecture logicielle équivaut aux aspects statiques et dynamiques les plus significatifs du système. L'architecture émerge des besoins de l'entreprise, tels qu'ils sont exprimés par les utilisateurs et autres intervenants et tels qu'ils sont reflétés par les cas d'utilisation.

Elle subit également l'influence d'autres facteurs :

- la plate-forme sur laquelle devra s'exécuter le système ;
- les briques de bases réutilisables disponibles pour le développement ;
- les considérations de déploiement, les systèmes existants et les besoins non fonctionnels (performance, fiabilité...)

- **Le processus unifié est itératif et incrémental**

Le développement d'un produit logiciel destiné à la commercialisation est une vaste entreprise qui peut s'étendre sur plusieurs mois. On ne va pas tout développer d'un coup. On peut découper le travail en plusieurs parties qui sont autant de mini projets. Chacun d'entre eux représentant une itération qui donne lieu à un incrément.

Une itération désigne la succession des étapes de l'enchaînement d'activités, tandis qu'un incrément correspond à une avancée dans les différents stades de développement.

1.4.3.2- le cycle de vie du processus unifié

L'objectif d'un processus unifié est de maîtriser la complexité des projets informatiques en diminuant les risques. UP est un ensemble de principes génériques adapté en fonction des spécificités des projets. UP répond aux préoccupations suivantes :

- QUI participe au projet ?
- QUOI, qu'est-ce qui est produit durant le projet ?
- COMMENT doit-il être réalisé ?
- QUAND est réalisé chaque livrable ?

1.4.3.3- L'architecture bidirectionnelle

UP gère le processus de développement par deux axes :

➤ **L'axe vertical** représente les principaux enchaînements d'activités, qui regroupent les activités selon leur nature. Cette dimension rend compte l'aspect statique du processus qui s'exprime en termes de composants, de processus, d'activités, d'enchaînements, d'artefacts et de travailleurs.

➤ **L'axe horizontal** représente le temps et montre le déroulement du cycle de vie du processus; cette dimension rend compte de l'aspect dynamique du processus qui s'exprime en terme de cycles, de phases, d'itérations et de jalons.

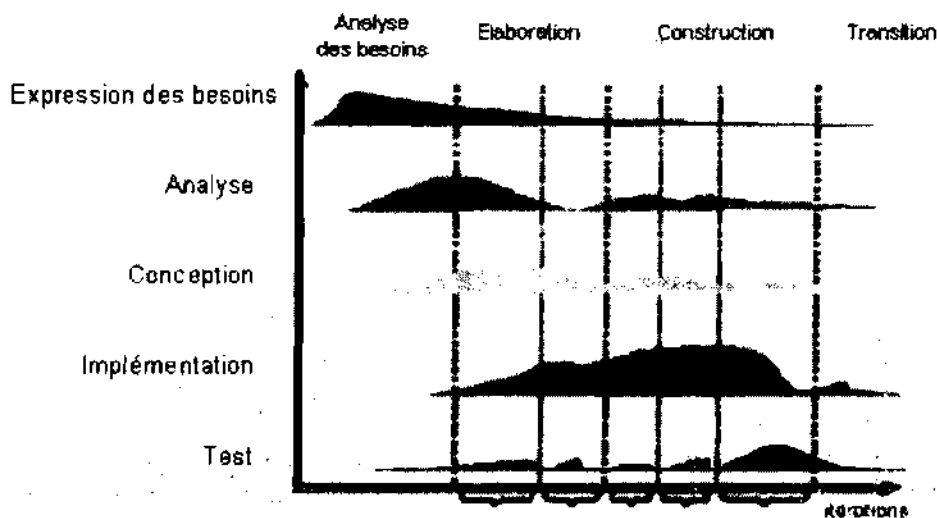


Figure 1.2 cycle de vie du processus unifié

UP répète un certain nombre de fois une série de cycle qui s'articule autour de 4 phases

- analyse des besoins
- élaboration
- construction
- transition

Pour mener efficacement un tel cycle, les développeurs ont besoins de toutes les représentations du produit logiciel

- un modèle de cas d'utilisation
- un modèle d'analyse : détailler les cas d'utilisation et procéder à une première répartition du comportement
- un modèle de conception : finissant la structure statique du système sous forme de sous systèmes, de classes et interfaces.
- un modèle d'implémentation : intégrant les composants
- un modèle de déploiement : définissant les nœuds physiques des ordinateurs
- un modèle de test : décrivant les cas de test vérifiant les cas d'utilisation
- une représentation de l'architecture

1.4.4-démarche d'analyse : les phases du processus unifié

1.4 .4.1- Analyse des besoins

L'analyse des besoins donne une vue du projet sous forme de produit fini. Cette phase porte essentiellement sur les besoins principaux (du point de vue de l'utilisateur), l'architecture générale du système, les risques majeurs, les délais et les coûts. On met en place le projet.

Elle répond aux questions suivantes :

- que va faire le système ? Par rapport aux utilisateurs principaux, quels services va-t-il rendre?
- quelle va être l'architecture générale (cible) de ce système
- quels vont être : les délais, les coûts, les ressources, les moyens à déployer?

1.4 .4.2-Elaboration

L'élaboration reprend les éléments de la phase d'analyse des besoins et les précise pour arriver à une spécification détaillée de la solution à mettre en œuvre. Elle permet de préciser la plupart des cas d'utilisation, de concevoir l'architecture du système et surtout de déterminer l'architecture de référence.

Au terme de cette phase, les chefs de projet doivent être en mesure de prévoir les activités et d'estimer les ressources nécessaires à l'achèvement du projet. Les tâches à effectuer dans la phase élaboration sont les suivantes :

- créer une architecture de référence
- identifier les risques, ceux qui sont de nature à bouleverser le plan, le coût et le calendrier
- définir les niveaux de qualité à atteindre
- formuler les cas d'utilisation pour couvrir les besoins fonctionnels et planifier la phase de construction
- élaborer une offre abordant les questions de calendrier, de personnel et de budget

1.4.4.3-Construction

La construction est le moment où l'on construit le produit. L'architecture de référence se métamorphose en produit complet. Le produit contient tous les cas d'utilisation que les chefs de projet, en accord avec les utilisateurs ont décidé de mettre au point pour cette version.

1.4 .4.4-Transition

Le produit est en version bêta. Un groupe d'utilisateurs essaye le produit et détecte les anomalies et défauts.

Cette phase suppose des activités comme la formation des utilisateurs clients, la mise en œuvre d'un service d'assistance et la correction des anomalies constatées.

1.5-La conduite du projet

1.5.1-Les acteurs du projet

- **Le groupe de supervision**

Il est responsable de l'ensemble du projet, à la fois au niveau de la prise des décisions relatives aux objectifs recherchés et des délais à respecter. Il est également responsable de la rédaction, du suivi du plan projet et de la définition des moyens à mettre en place pour sa réalisation. Le groupe est constitué de :

- **M. Raphael S. DAKISSAGA**, responsable du CPFACE ;
- **M. Vital DJIGMA**, ingénieur informaticien à la CNSS ;
- **Dr. Michel SOME**, enseignant à l'ESI, notre superviseur.

- **Le groupe de projet**

Il est chargé de l'exécution du plan de projet c'est-à-dire la définition et l'analyse des besoins la spécification fonctionnelle, la conception et la programmation. Il comprend :

- **Abdou KABRE** ;
- **Saïbou PARE**.

- **Le groupe d'utilisateur**

Il a un rôle consultatif. Il fournit également toutes les informations nécessaires à la bonne conduite du projet. Il intervient dans la validation des dossiers d'étude et des prototypes réalisés par le groupe de projet.

1.5.2-Le planning

Le groupe de projet détermine, en accord avec le groupe de supervision, les temps de développement associés à toutes les activités. Pour le faire, on doit prendre en compte des imprévus inhérents au développement. Pour prévoir la durée d'une activité, on calculera sa durée dans le meilleur des cas, puis on y ajoutera un délai de garantie des problèmes envisagés, et enfin on pondère le tout pour tenir compte de l'imprévu.

Mois	Aout	Septembre	Octobre	Novembre	Décembre	Janvier	Février	Mars
Phases								
Analyse des besoins		■						
Elaboration			■					
Construction				■	■	■	■	■
Transition								■

Tableau 1.2 Plan prévisionnel

Légende : une cellule correspond à deux semaines

Chapitre 2 Etude de l'existant

Le chapitre 1 nous a permis de cerner la problématique de **Gestion des prestations du CPFACE** ainsi que les résultats attendus de notre étude. Quant au deuxième chapitre, il vise à étudier le système actuel de la gestion des prestations du CPFACE tout en mettant en exergue les points de dysfonctionnement. Il s'agira de faire une analyse critique de la situation actuelle. Ainsi ce chapitre se subdivise en trois grandes parties. La première partie traite de la capture des besoins fonctionnels qui est la première activité de la phase d'analyse des besoins. La seconde partie développe l'analyse, deuxième activité de la phase d'analyse des besoins. Enfin, la dernière partie fait un diagnostic de l'étude de l'existant.

2.1-Capture des besoins fonctionnels

L'objectif de cette activité est de définir les besoins principaux, de recenser les besoins fonctionnels (du point de vue de l'utilisateur) qui conduisent à l'élaboration des modèles de cas d'utilisation et d'appréhender les besoins non fonctionnels (technique).

Elle a été réalisée suite à des interviews effectués avec le responsable et le coordinateur des cours du Centre de Production et de Formation en Audiovisuel et Communication en Entreprise (CPFACE). L'objectif des interviews est de s'imprégner des objectifs du projet et de comprendre le fonctionnement actuel des services. Les informations recueillies sont décrites à travers le compte rendu suivant :

2.1.1- Compte rendu des interviews

Compte rendu d'interview	Folio : 1/4
Personne rencontrée : responsable CPFACE	
<p><u>Procédure d'inscription au centre</u></p> <p>L'une des missions du CPFACE est la formation des techniciens et des acteurs de la santé et sécurité au travail en audiovisuel et communication en entreprise. La procédure d'inscription au centre se décrit comme suit :</p> <ul style="list-style-type: none"> ▪ L'édition des conditions d'entrée par le Responsable CPFACE ▪ le Responsable CPFACE lance des appels à candidature auprès des institutions des pays membres de l'APRP. ▪ Les institutions et structures proposent des candidatures ; ▪ le Responsable CPFACE reçoit et étudie les dossiers de candidature et établit une liste des candidats retenus puis informe leurs structures ; ▪ Le Responsable CPFACE informe ensuite les structures dont les candidats n'ont pas été retenus ▪ Les structures confirment le besoin de formation des candidats retenus et s'acquittent alors des frais de formations de leurs candidats ; ▪ Les structures peuvent maintenant s'adresser au responsable du CPFACE pour des questions spécifiques. 	

Compte rendu d'interview	Folio : 2/4
Personne rencontrée : responsable CPFACE	
<p style="text-align: center;"><u>L'organisation des formations continues</u></p> <p>La formation continue consiste en l'organisation dans les locaux du Centre de séminaires/ateliers intensifs sur des thèmes se rapportant à l'audiovisuel, la communication ou la santé et sécurité au travail. La procédure d'organisation se présente comme suit :</p> <ul style="list-style-type: none"> ▪ Le responsable du CPFACE élabore un programme de formation continue (séminaires/ateliers) en lien avec les compétences du centre ; ▪ Il lance alors des appels à candidature auprès des institutions membres de l'JAPRP ; ▪ Le responsable du CPFACE identifie des formateurs et fixe les dates et les coûts puis informe le public cible ; ▪ Les formateurs sont chargés maintenant de l'exécution du programme de formation ; ▪ Le responsable du CPFACE s'occupe de la validation de la formation et délivrance des attestations ; ▪ Le responsable du CPFACE présente un rapport financier et d'exécution de l'activité à la direction générale ; ▪ La gestion des questions pendantes est également assurées par le responsable du CPFACE ; <p style="text-align: center;"><u>Procédure de collecte d'œuvre</u></p> <p>Au delà de la production et de la promotion de la coproduction entre structures de santé et de sécurité au travail des supports audiovisuels, le CPFACE a pour autre mission de constituer « une banque de données » des productions audiovisuelles en santé et sécurité au travail de la sous-région et de leurs diffusions.</p> <p>La procédure de collecte des œuvres est la suivante :</p> <ul style="list-style-type: none"> ▪ Le chargé de la gestion informe aux producteurs que le CPFACE est la médiathèque africaine en matière de prévention de risques professionnels ; ▪ Le chargé de la gestion crée des fiches d'inscription à envoyer à chaque producteur ; ▪ les producteurs définissent des modalités techniques et conditions de cession, remplissent les fiches d'inscription et leurs envoient par courrier ; ▪ Le centre retourne un accusé de réception par courrier ; ▪ Le chargé de la gestion se charge alors de classer les œuvres par catégorie et par thème traité ; ▪ L'œuvre reçue est ensuite consignée (enregistrement dans la collection) ; ▪ Le chargé de la gestion rend l'œuvre disponible (pour une éventuelle consultation) ; 	

Compte rendu d'interview	Folio : 3/4
Personne rencontrée : responsable CPFACE	
<p align="center">Procédure d'emprunt d'une œuvre</p> <ul style="list-style-type: none"> ▪ Une fois collectées et consignées, les œuvres peuvent faire l'objet d'emprunt. La procédure d'emprunt des œuvres se résume comme suit : ▪ L'entreprise, le partenaire, l'ONG, ou le ministère sollicite une œuvre en donnant ses descriptions. ▪ Le chargé de la gestion recherche dans la collection une œuvre répondant aux descriptions ▪ L'intéressé consulte l'œuvre et confirme sa demande ▪ Le chargé de la gestion lui rappelle les conditions d'exploitation ▪ L'intéressé peut maintenant rentrer en possession de l'œuvre ▪ L'organisation d'une édition du SYAPRO ▪ L'organisation du prix se résume en trois points essentiels à savoir : <ul style="list-style-type: none"> ▪ Le choix un thème pour le SYAPRO, ▪ L'établissement du programme d'activité du SYAPRO ; ▪ L'exécution du programme d'activité du SYAPRO ; <p align="center">L'organisation d'une édition du prix « Santé et Sécurité au Travail » du FESPACO</p> <p>Le prix « santé et sécurité au travail » est une activité promotionnelle initiée par l'Interafricaine de la prévention des risques professionnels (IAPRP) auprès du Festival panafricain du cinéma et de la télévision de Ouagadougou (FESPACO) et visant à donner de la visibilité aux actions de lutte contre les risques professionnels entreprises en Afrique. La Caisse Nationale de Sécurité Sociale du Burkina assure la coordination et l'organisation du prix, au nom de l'IAPRP.</p> <ul style="list-style-type: none"> ▪ Le responsable du CPFACE informe l'IAPRP sur la tenue du FESPACO et du prix « santé et sécurité au travail » et procède à l'élaboration d'un projet de budget ▪ Le Directeur Générale de la CNSSS sollicite alors de l'IAPRP le financement du prix ; ▪ La confirmation de l'offre du prix auprès du FESPACO est fait par le responsable du CPFACE ; ▪ Le DG procède à la mise en place d'un jury international dénommé jury prix spécial et d'un comité d'organisation qui aura pour charge le suivi des correspondances liées à l'organisation du prix; ▪ Le comité d'organisation assure également la confection des affiches, trophées, diplômes du prix, la coordination des activités des jurys avec la commission des prix spéciaux du FESPACO, l'accueil des membres des jurys venant de l'extérieur du Burkina, leur hébergement dans les hôtels et leur déplacement (moyens de transport) dans les différents centres d'intérêts du FESPACO, dans le cadre de leurs activités de visionnage des films ▪ L'activité se termine par la présentation par le comité d'organisation d'un rapport financier et d'exécution du prix à la direction générale du CNSS. 	

Compte rendu d'interview	Folio : 4/4
Personne rencontrée : coordonnateur des cours du CPFACE	
<p align="center">Procédure de programmation des cours</p> <p>La programmation des cours au centre s'effectue par semaine : Le coordonnateur établit un contact avec les enseignants en vue d'obtenir leurs heures vacantes. Les enseignants à leur tour, fournissent leurs heures vacantes au coordonnateur du cours du centre. Une fois les heures vacantes connues, le coordonnateur des cours procède à l'établissement du programme de cours hebdomadaire. Le programme de cours ainsi établi est publié par le coordonnateur à l'endroit des enseignants et des étudiants.</p>	

2.2.1.2- Fiches descriptives des cas d'utilisations

- **Formalisme adopté**

Pour une meilleure lisibilité le formalisme suivant sera adopté :

Cas d'utilisation i : « Nom du cas d'utilisation »
Résumé du Cas d'utilisation i :
Les acteurs du Cas d'utilisation i
Action
Action1 Action2 Action N
Règles de gestion et d'organisation
Règle 1 Règle 2 Règle N

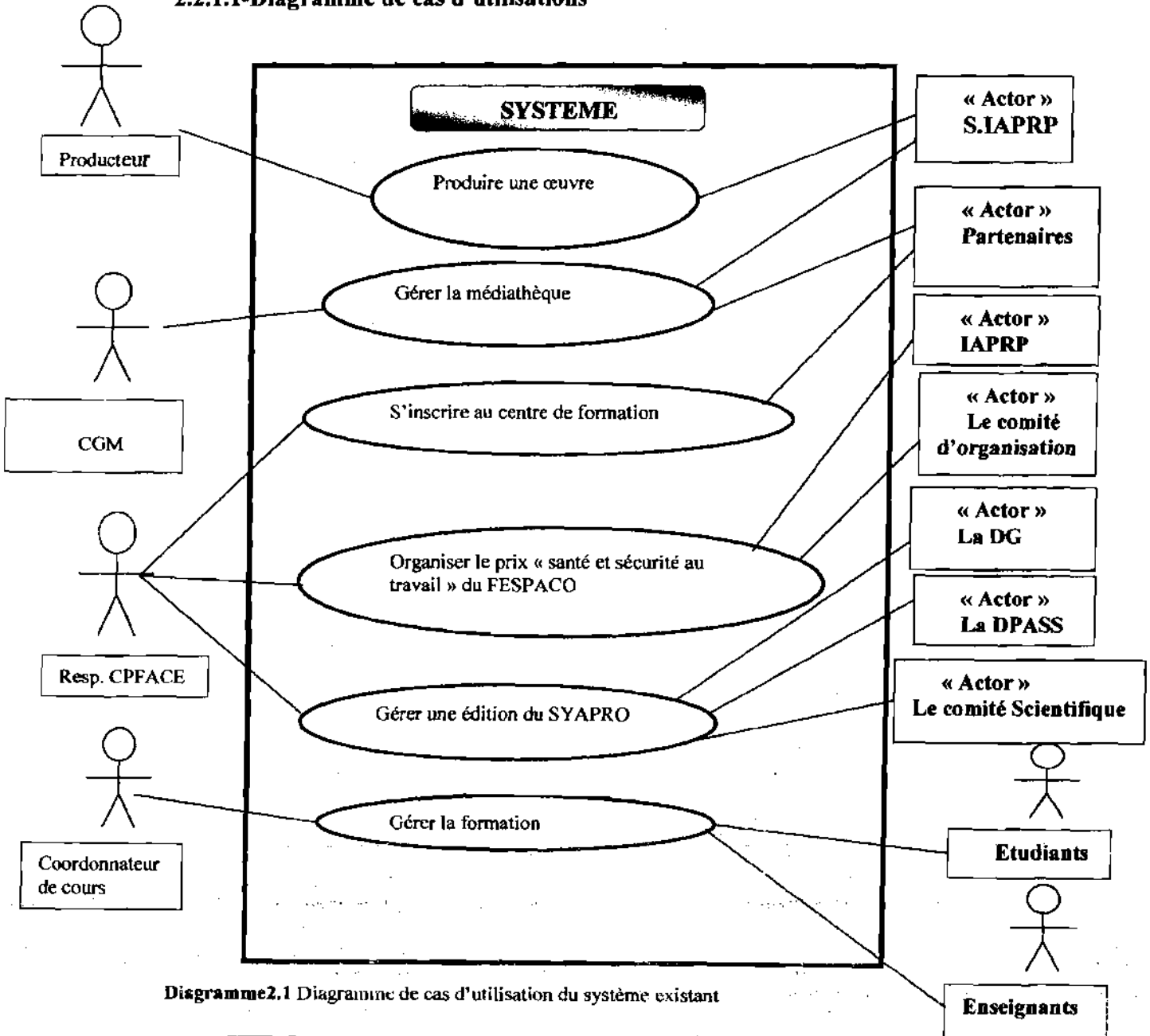
2.2-Analyse

Cette phase donne une vue du projet sous forme de produit fini. Elle porte essentiellement sur les besoins principaux (du point de vue de l'utilisateur) et l'architecture générale.

Elle décrit pour ce fait le fonctionnement actuel du système d'information tout en mettant en évidence les besoins fonctionnels et non fonctionnels.

2.2.1-Elaboration du diagramme de cas d'utilisation⁴

2.2.1.1-Diagramme de cas d'utilisations



⁴ Le diagramme de cas d'utilisation et : les concepts et ses formalismes sont décrits en Annexes partie 2

- **Description des cas d'utilisation**

Inscription au centre

L'inscription au CPFACE est réalisée par le cas d'utilisation « S'inscrire au CPFACE »

Cas d'utilisation 1 : S'inscrire au CPFACE
Résumé : décrit la procédure d'inscription par les institutions des pays membres de l'IAPRP de leurs agents (techniciens et acteurs de la santé et sécurité au travail) à la formation en audiovisuel et communication en entreprise
Acteurs : le responsable du CPFACE, les institutions et structures des pays membres de l'IAPRP
Actions
<ul style="list-style-type: none"> ▪ Ediction des conditions d'entrée ▪ Appel à candidature ▪ Les institutions et structures proposent des candidatures ▪ Réception des dossiers de candidature par le CPFACE ▪ Etudier les dossiers ▪ Etablir une liste des candidats retenus et informer leurs structures ▪ Informer les structures dont les candidats n'ont pas été retenus ▪ Les structures confirment le besoin de formation des candidats retenus ▪ Acquiescement des frais de formations par les structures ▪ Les structures peuvent s'adresser au responsable du CPFACE pour des questions spécifiques
Règles de gestion et d'organisation
<ul style="list-style-type: none"> ▪ Peuvent inscrire un agent, une institution ou structure d'un pays membre de l'IAPRP ▪ Chaque structure peut présenter plus d'un candidat ▪ Ces candidats doivent être titulaire du Baccalauréat toutes séries confondues ▪ Ou avoir participé à trois sessions de formation continue organisées par le Centre

Gestion de la formation continue

Cas d'utilisation 2 : Gérer la formation continue (séminaires/ateliers en liens avec les compétences du centre

Résumé : décrit la procédure de gestion de la formation continue c'est-à-dire la formation à travers des séminaires/ateliers en liens avec les compétences du centre

Acteurs : les institutions et structures des pays membres de l'IAPRP, les Formateurs, le responsable du CPFACE, la direction générale

Actions

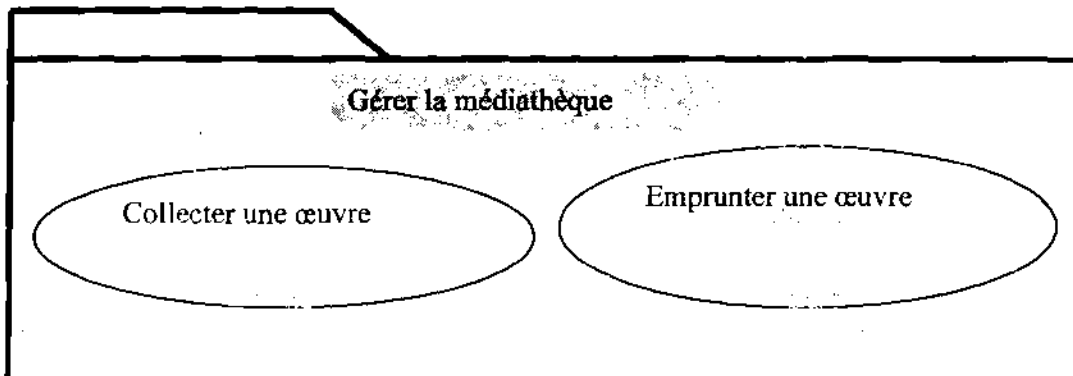
- Elaborez un programme de formation continue (séminaires/ateliers) en lien avec les compétences du centre
- Lancement d'appel à candidature par le responsable du CPFACE
- Identification des formateurs par le responsable du CPFACE
- Le responsable du CPFACE fixe les dates et les coûts et informe le public cible
- Choix du lieu par le responsable du CPFACE
- Exécuter le programme de formation
- Valider formation et délivrer les attestations
- Le responsable du CPFACE présenter un rapport financier et d'exécution de l'activité à la direction générale
- La gestion des questions pendantes également assurées par le responsable du CPFACE

Règles de gestion et d'organisation

- Peuvent s'inscrire un agent à un séminaire ou un atelier, une institution ou structure d'un pays membre de l'IAPRP
- Chaque structure peut présenter plus d'un agent pour une formation

Gestion de la médiathèque

Le cas d'utilisation gérer la médiathèque est un paquetage regroupant les cas d'utilisation «Collectionner une œuvre» et «Emprunter une œuvre»

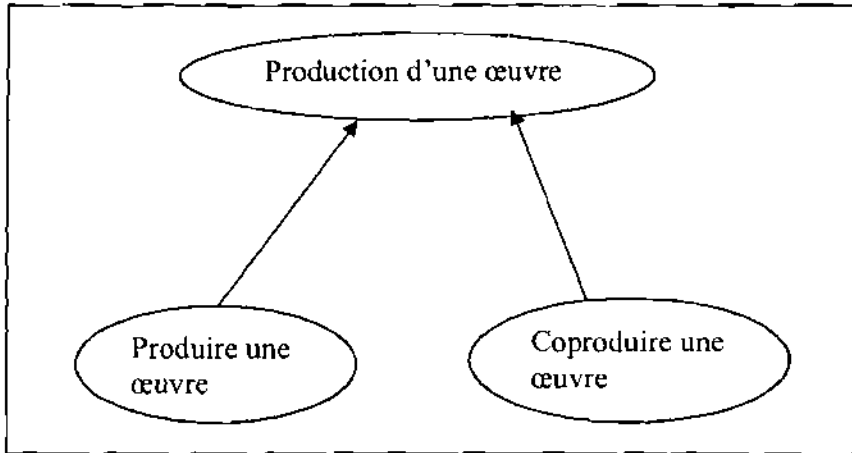


Cas d'utilisation 3 : Emprunter une œuvre
Résumé : décrit la procédure d'emprunt d'une œuvre par un ministère, une institution ou une entreprise ou une structure des pays membre de l'IARPR.
Acteurs : les institutions et structures des pays membres de l'IARPR, le chargé de la gestion de la bibliothèque / médiathèque, les partenaires.
Actions
<ul style="list-style-type: none"> ▪ L'entreprise, le partenaire, l'ONG, ou le ministère sollicite une œuvre en donnant ses descriptions. ▪ Le chargé de la gestion recherche dans la collection une œuvre répondant aux descriptions ▪ L'intéressé consulte l'œuvre et confirme sa demande ▪ Le chargé de la gestion lui rappelle les conditions d'exploitation ▪ L'intéressé peut maintenant rentrer en possession de l'œuvre
<ul style="list-style-type: none"> ▪ Règles de gestion et d'organisation
<ul style="list-style-type: none"> ▪ Peut emprunter une œuvre toute entreprise, institution ou structure d'un pays membre de l'IARPR ▪ entreprise, institution ou structure d'un pays membre de l'IARPR peut emprunter plusieurs œuvres pour consultation ▪ une œuvre peut appartenir à plusieurs catégories (images, film...) ▪ un emprunt concerne une période donnée

Cas d'utilisation 11 : Collectionner une œuvre
Résumé : Ce cas d'utilisation montre les différentes étapes dans la collection des œuvres de la bibliothèque médiathèque
Acteurs : les producteurs d'œuvre, le chargé de la gestion de la bibliothèque/médiathèque
Actions
<ul style="list-style-type: none"> ▪ Informer aux producteurs que le CPFACE est la médiathèque africaine en matière de prévention de risques professionnels ▪ Créer des fiches d'inscription à envoyer aux producteurs ▪ Définition des modalités techniques et conditions de cession ▪ Le producteur remplit la fiche d'inscription et l'envoie par courrier ▪ Le centre retourne un accusé de réception par courrier ▪ Le chargé de la gestion classe les œuvres par catégorie et par thème traité ▪ Le chargé de la gestion consigne l'œuvre reçue (enregistrement dans la collection) ▪ Le chargé de la gestion rend l'œuvre disponible (pour une éventuelle consultation)
Règles de gestion et d'organisation

Production d'une œuvre

Ce cas d'utilisation se spécialise en deux cas d'utilisation à savoir la production par le CPFACE et la coproduction avec une structure externe.

**Cas d'utilisation 4 : Produire un œuvre**

Résumé : décrit la procédure de production d'une œuvre par le Centre de Production et de Formation en Audiovisuel et Communication en Entreprise.

Acteurs : CPFACE, DG, DPASS, entreprises, partenaires

Actions

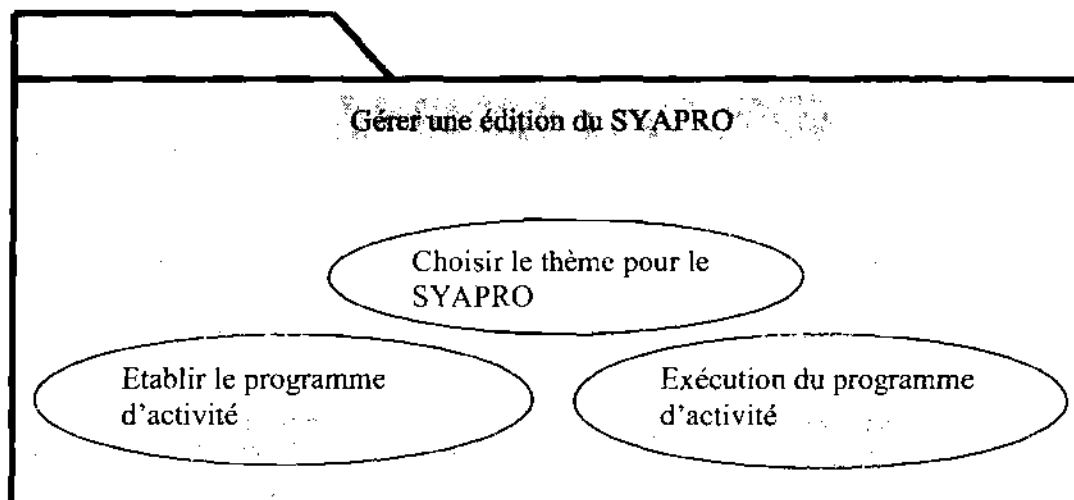
- Le besoin d'un support audiovisuel de sensibilisation est exprimé
- Elaboration d'un projet de film
- Choisir un support
- Elaboration d'un projet de budget
- Adoption du budget par la Direction Générale et mise à la disposition du CPFACE
- Mise en place d'une équipe de production (repérage, tournage, montage, finition)
- Postproduction (montage, mixage)
- Finition (Copie de l'œuvre audiovisuel dans différents supports)
- Mise à disposition auprès du commanditaire
- Gestion du patrimoine technique
- Gestion des supports audiovisuels

Règles de gestion et d'organisation

Cas d'utilisation 5 : Coproduire un œuvre
Résumé : décrit la procédure de coproduction d'une œuvre par le Centre de Production et de Formation en Audiovisuel et Communication en Entreprise en partenariat avec un ministère, une structure, une institution ou une entreprise d'un pays membre de l'IARPR.
Acteurs : CPFACE, institution ou structure d'un pays membre de l'IARPR partenaires
Actions
<ul style="list-style-type: none"> ▪ Le besoin d'un support audiovisuel de sensibilisation est exprimé ▪ Elaboration d'un projet de film ▪ Choisir un support ▪ Informer les partenaires coproducteurs et recueillir leur accord pour une coproduction ▪ Mise à disposition de la quote-part de chaque coproducteur ▪ Mise en place d'une équipe de production (repérage, tournage, montage, finition) ▪ Postproduction (montage, mixage) ▪ Finition (Copie de l'œuvre audiovisuel dans différents supports) ▪ Mise à disposition auprès du commanditaire ▪ Gestion du patrimoine technique ▪ Gestion des supports audiovisuels
Règles de gestion et d'organisation

Gérer une édition du SYAPRO

Ce cas d'utilisation regroupe les cas suivants : Choisir un thème pour le SYAPRO, Etablir le programme d'activités, Exécuter le programme d'activités.



Cas d'utilisation 6 : Choisir un thème pour le SYAPRO
Résumé : décrit la procédure de choix du thème du Symposium sur l'Audiovisuel et la Prévention des Risques professionnels de Ouagadougou (SYAPRO).
Acteurs : responsable du CPFACE, la direction générale de la caisse
Actions
<ul style="list-style-type: none"> ▪ Le responsable du CPFACE propose un thème ▪ Le DPASS approuve le thème proposé ▪ Le DG valide le thème ▪ Le DG fixe les dates
Règles de gestion et d'organisation

Cas d'utilisation 7 : Etablir le programme d'activité du SYAPRO
Résumé : décrit la procédure d'élaboration du programme d'activités liées au Symposium sur l'Audiovisuel et la Prévention des Risques professionnels de Ouagadougou (SYAPRO).
Acteurs le comité scientifique et technique, le directeur général, le conseil d'administration
Actions
<ul style="list-style-type: none"> ▪ Elaborer un projet de contenu (responsable du CPFACE) ▪ Elaborer un projet de budget (responsable du CPFACE) ▪ Adopter le projet de contenu (DG) ▪ Adopter le budget (par le Conseil d'Administration) ▪ Créer le comité scientifique et technique (DG)
Règles de gestion et d'organisation

Cas d'utilisation 8 : Exécuter le programme d'activité du SYAPRO
Résumé : décrit la procédure d'exécution du programme d'activité élaboré dans le cadre d'une édition de SYAPRO.
Acteurs : le comité scientifique et technique, le comité d'organisation
Actions
<ul style="list-style-type: none">▪ Identifier les conférenciers▪ Elaborer la liste des institutions, structures, personnes ressources et personnalités à inviter▪ Relire le programme pour préciser le contenu de chaque activité▪ Concevoir un projet de dépliant▪ Préparer les règlements intérieurs des différentes compétitions inscrites au programme▪ Analyser le projet de budget▪ Etablir les fiches d'inscription▪ Appel à participation▪ Mettre en place un comité d'organisation▪ Publier les actes▪ Relever les recommandations
Règles de gestion et d'organisation

Organiser le prix « Santé et sécurité au travail » du FESPACO

Les différentes actions entrant dans l'organisation du prix « Santé et Sécurité au Travail » du FESPACO sont décrites dans le cas d'utilisation « Organiser le prix « Santé et Sécurité au Travail » du FESPACO ».

Cas d'utilisation 10 : Organiser le prix « santé et sécurité au travail » du FESPACO
Résumé : décrit la procédure d'organisation du prix « santé et sécurité au travail » du FESPACO.
Acteurs : le comité d'organisation, le responsable du CPFACE, le comité d'organisation
Actions
<ul style="list-style-type: none"> ▪ Informer l'IAPRP sur la tenue du FESPACO et du prix « santé et sécurité au travail » ▪ Elaborer un projet de budget (le responsable du CPFACE) ▪ Solliciter de l'IAPRP le financement du prix (DG) ▪ Confirmer l'offre du prix auprès du FESPACO (le responsable du CPFACE) ▪ Mettre en place un jury international dénommé jury prix spécial par le DG ▪ Mise en place d'un comité d'organisation (DG) ▪ Suivre les correspondances liées à l'organisation du prix (le comité d'organisation) ▪ Assurer la confection des affiches, trophées et diplômes du prix ▪ Assurer la coordination des activités des jurys avec la commission des prix spéciaux du FESPACO ▪ Accueillir les membres des jurys venant de l'extérieur du Burkina et assurer leur hébergement dans les hôtels ▪ Assurer le déplacement des membres des jurys (moyens de transport) dans les différents centres d'intérêts du FESPACO, dans le cadre de leurs activités de visionnage des films ▪ Présenter un rapport financier et d'exécution de l'activité
Règles de gestion et d'organisation
▪

2.2.1- Elaboration du diagramme de séquence

2.2.1.1 -diagrammes de séquence⁵

Nous ne représentons que les diagrammes de séquences correspondant au scénarii nominaux. Les scénarii alternatifs ne sont donc pas pris en compte.

Le diagramme de séquence inscription au centre

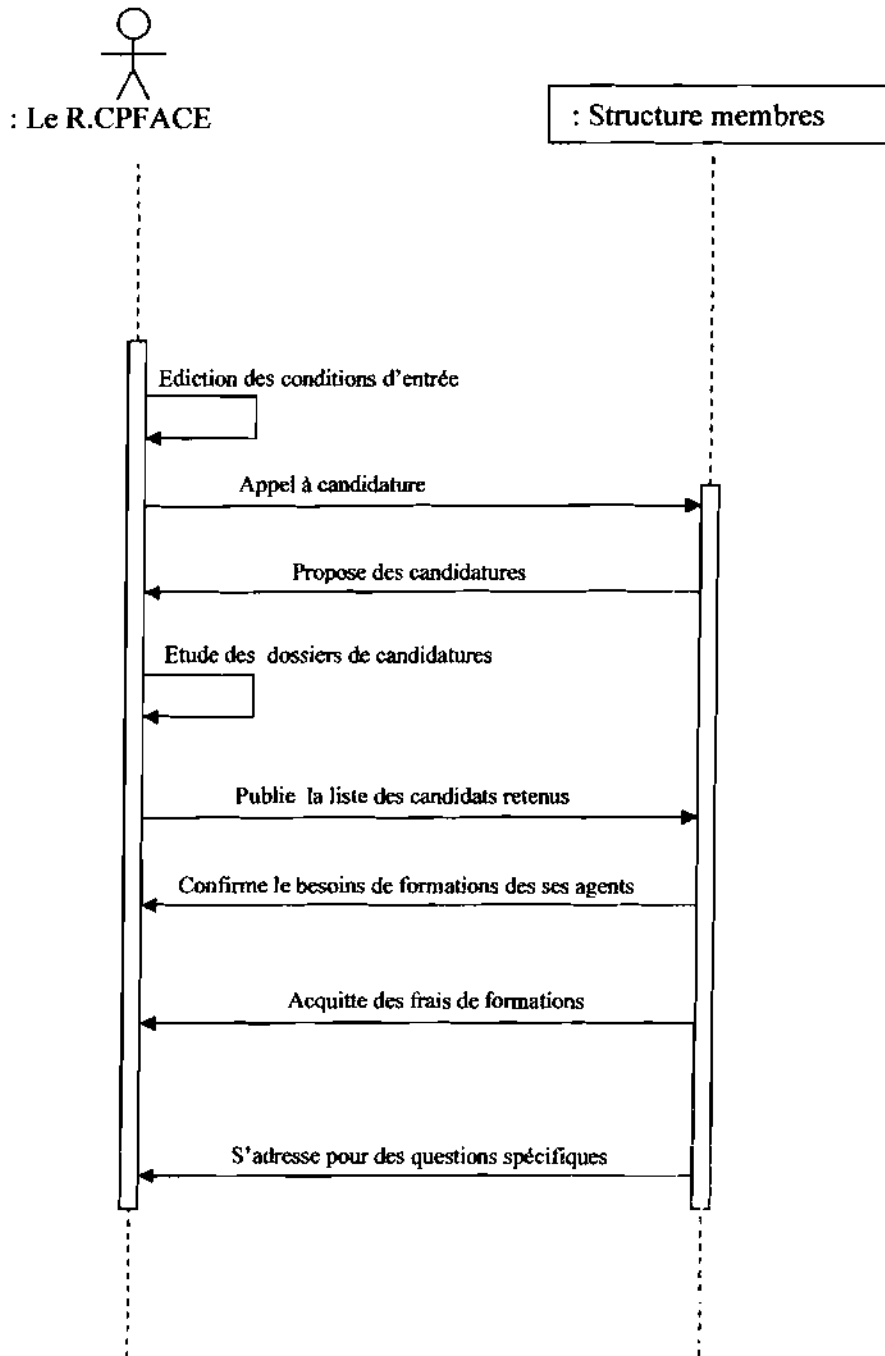


Diagramme2.2 Diagramme de séquence décrivant l'inscription au centre

⁵ Les concepts et les formalismes du diagramme de séquences sont décrits en Annexes partie 3

Le diagramme de séquence décrivant la collection d'une œuvre

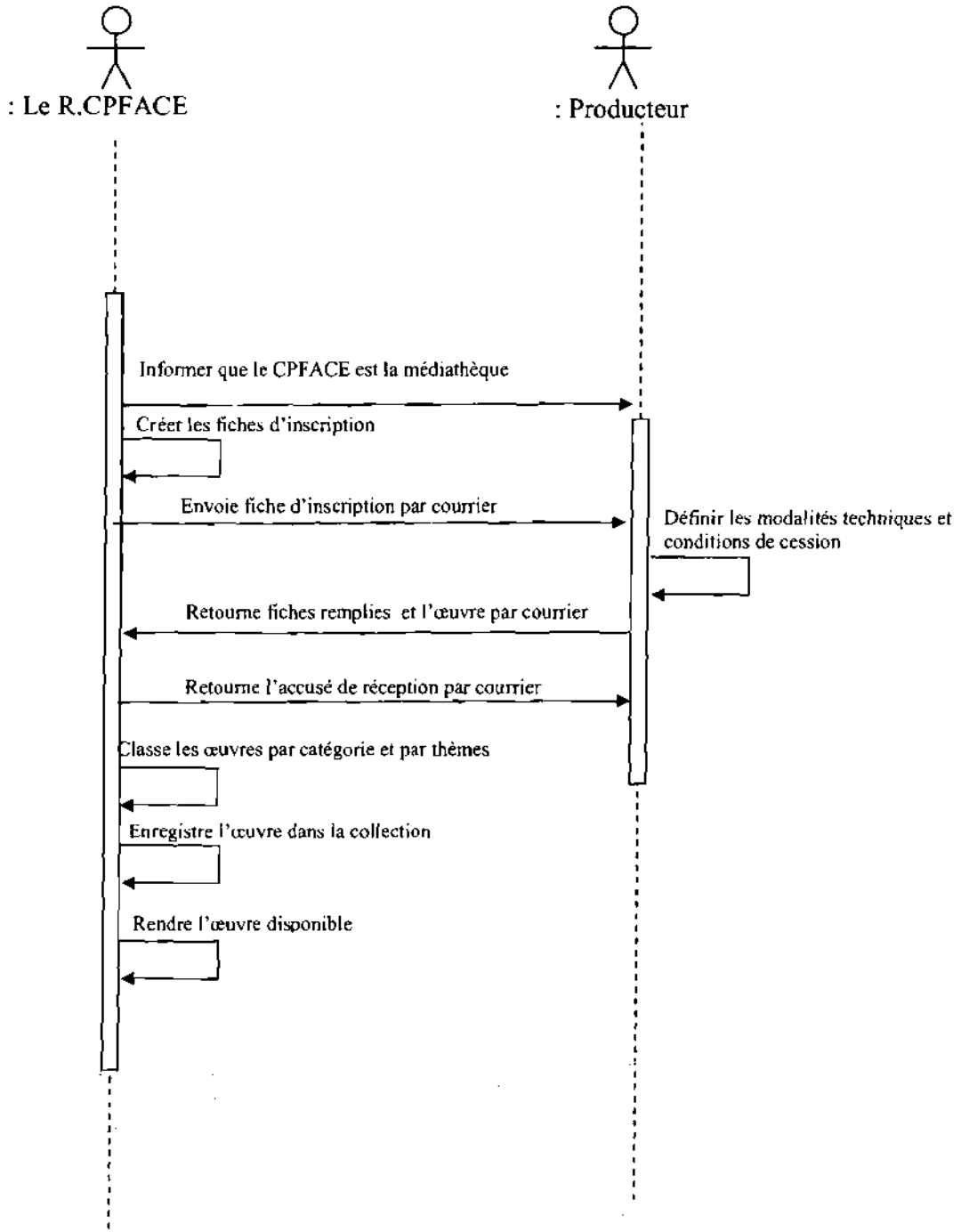


Diagramme2.3 Diagramme de séquence décrivant la collection d'une œuvre

Le diagramme de séquence décrivant l'emprunt d'une œuvre

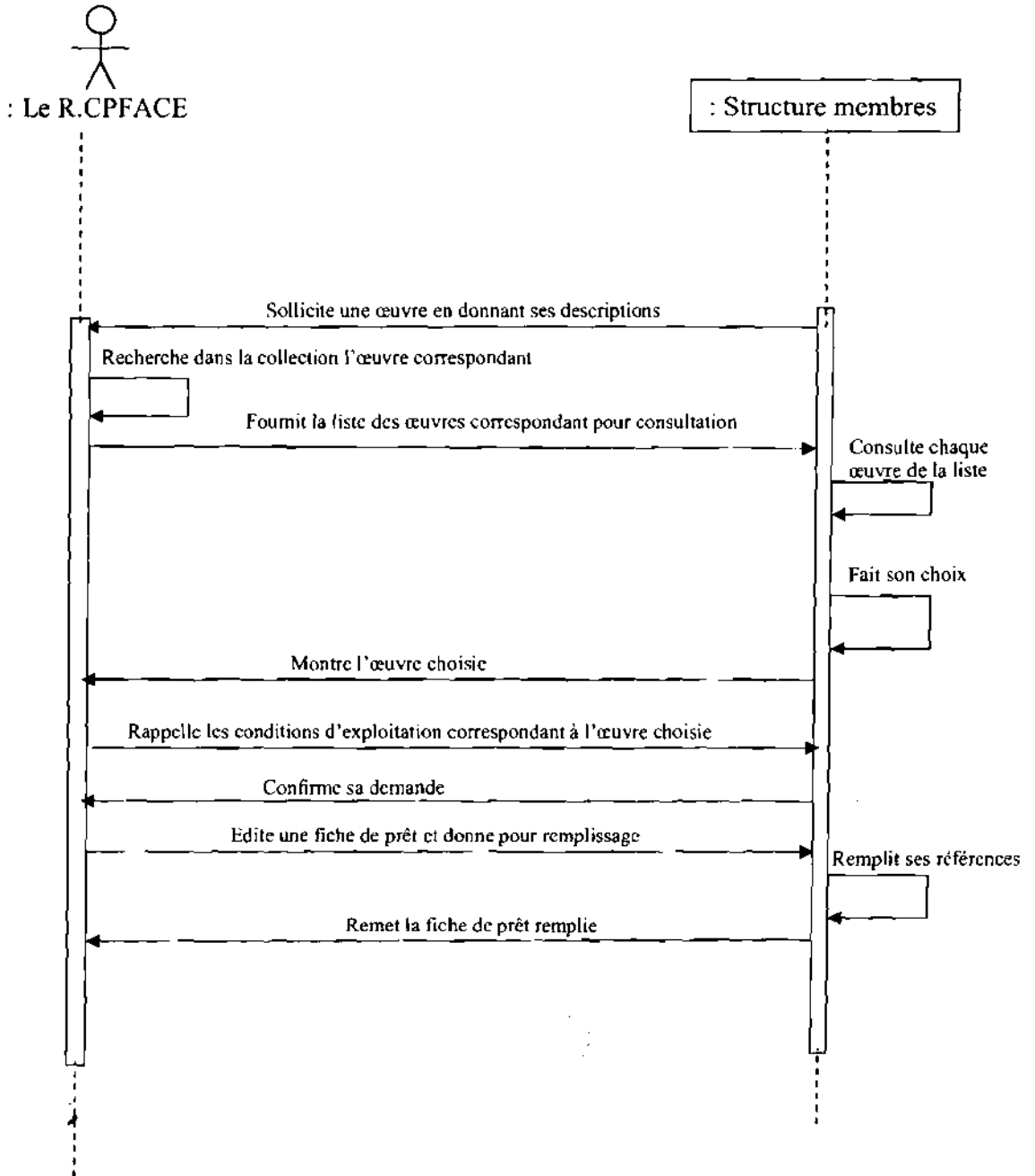


Diagramme2.4 Diagramme de séquence décrivant la procédure d'emprunt d'une œuvre

Le diagramme de séquence décrivant la production d'une œuvre
 Ce diagramme décrit la procédure de production d'œuvre.

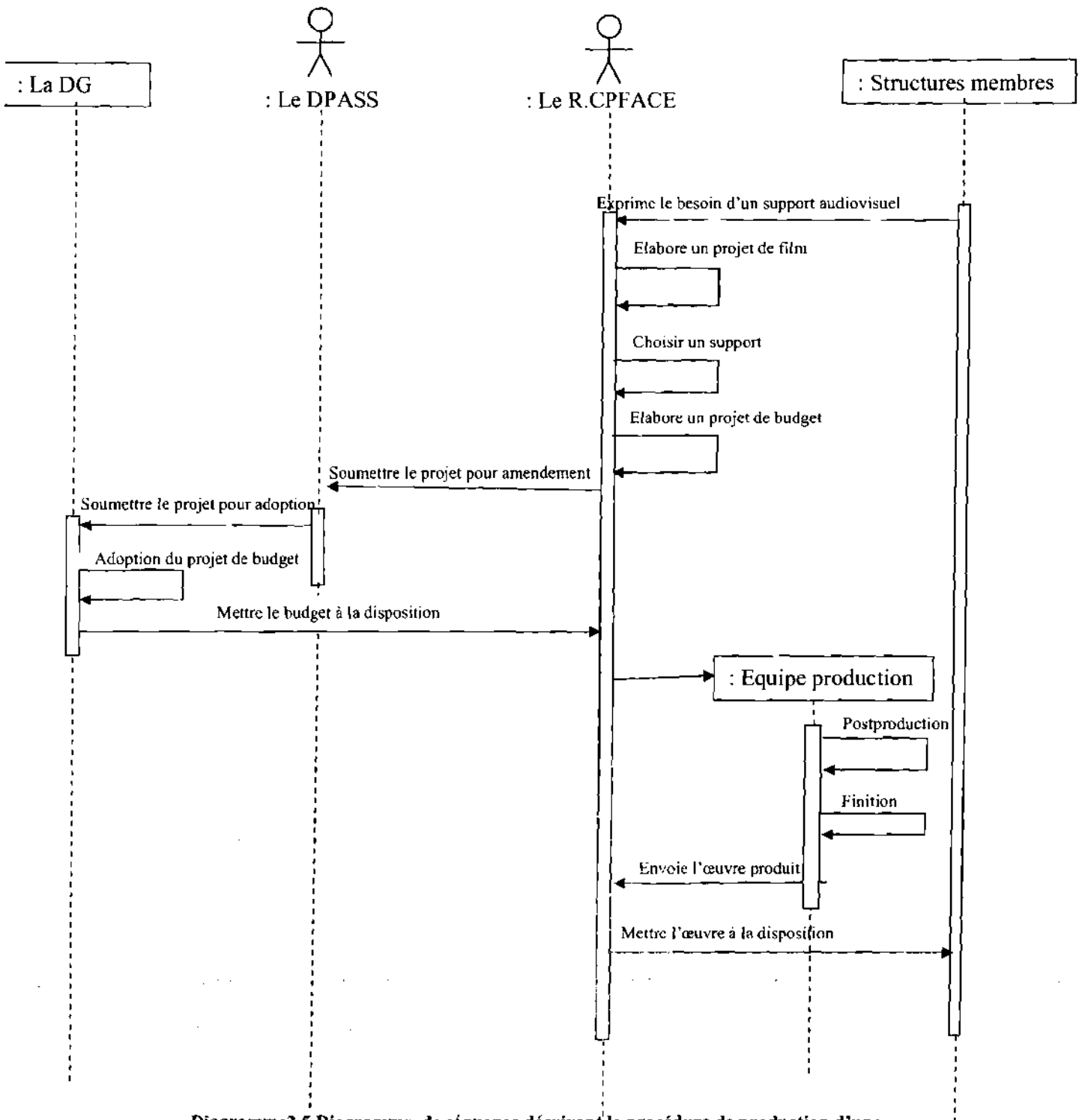


Diagramme 2.5 Diagramme de séquence décrivant la procédure de production d'une œuvre

Le diagramme de séquence du cas d'utilisation « Choisir la thème du SYAPRO »

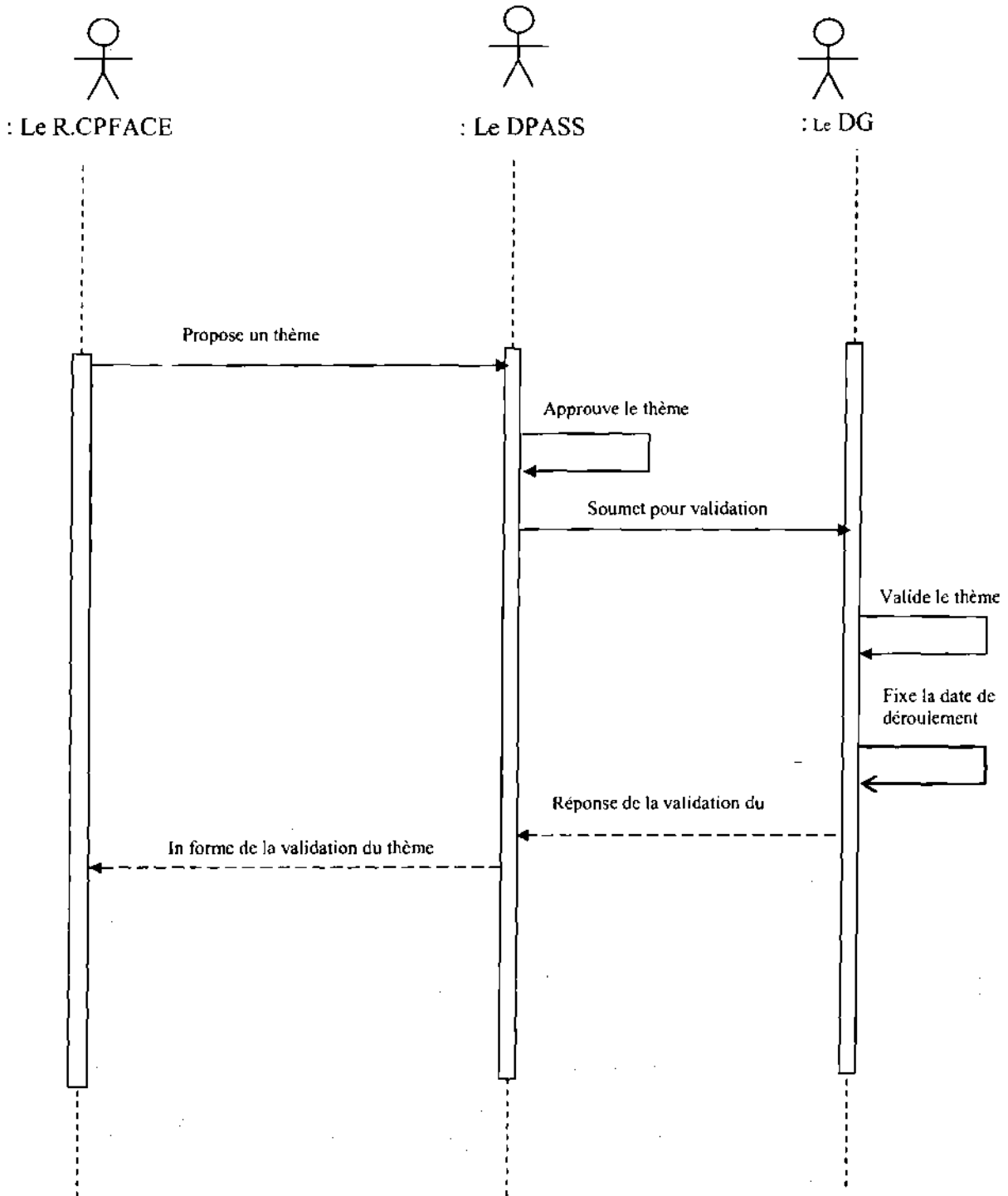


Diagramme2.7 Diagramme de séquence décrivant la procédure de choix d'un thème

Le diagramme de séquence du cas d'utilisation « Etablir le programme d'activité du SYAPRO »

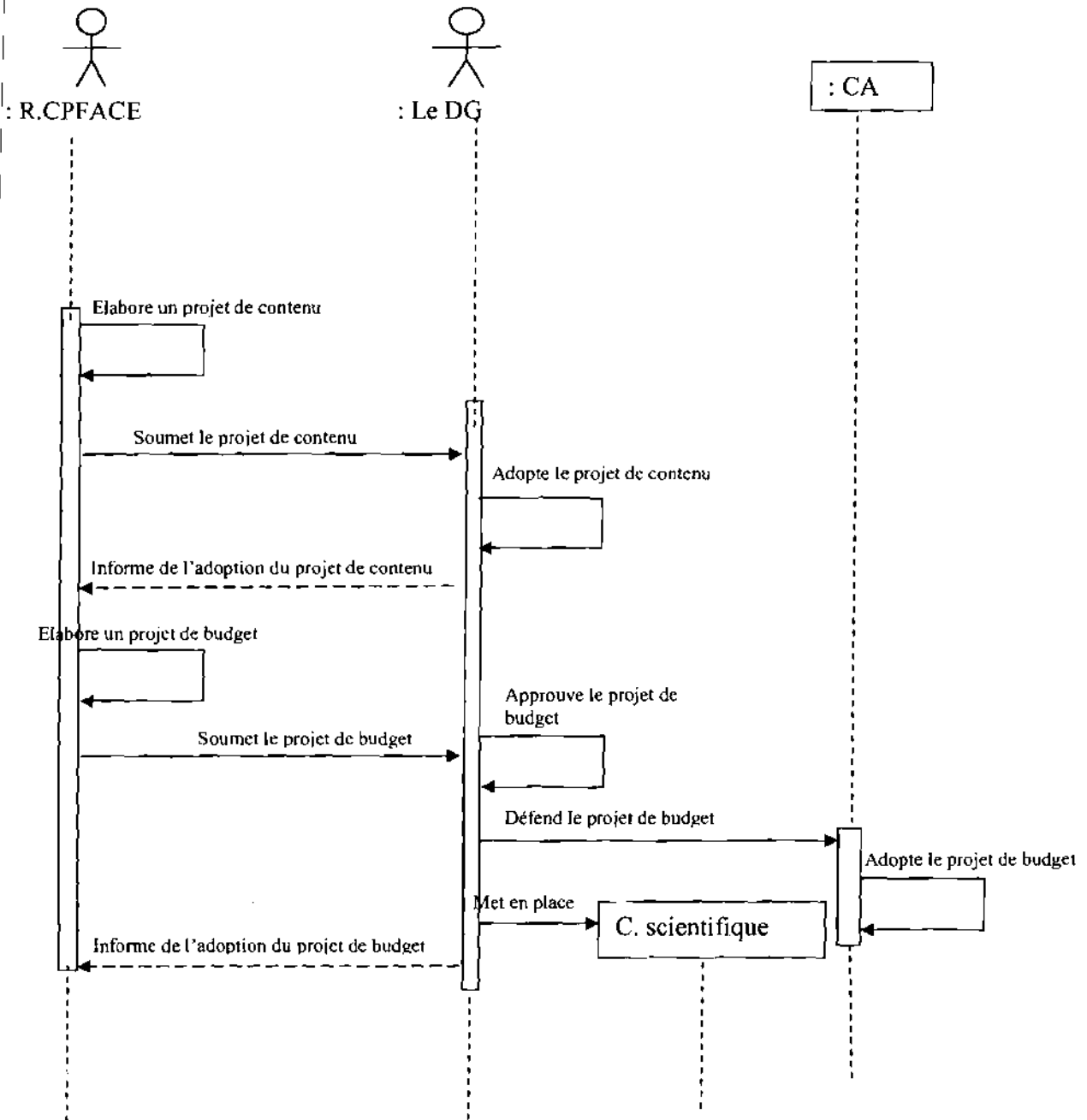


Diagramme2.8 Diagramme de séquence décrivant la procédure d'établissement d'un programme d'activité du SYAPRO

Le diagramme de séquence du cas d'utilisation « Exécution du programme d'activité du SYAPRO »

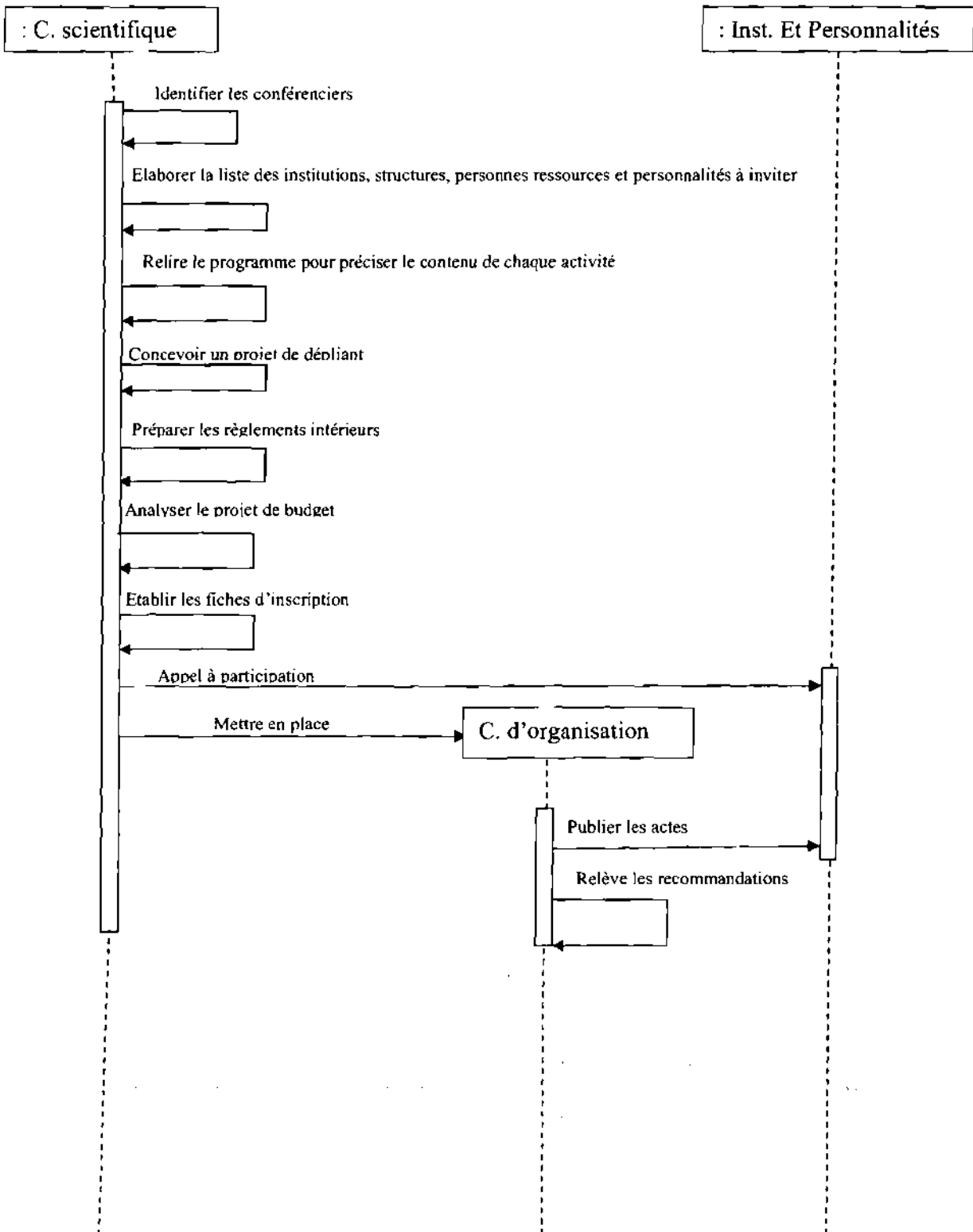


Diagramme 2.9 Diagramme de séquence décrivant la procédure d'exécution du programme d'activité du SYAPRO

Le diagramme de séquence du cas d'utilisation « Organiser le prix Santé et Sécurité au Travail du FESPACO »

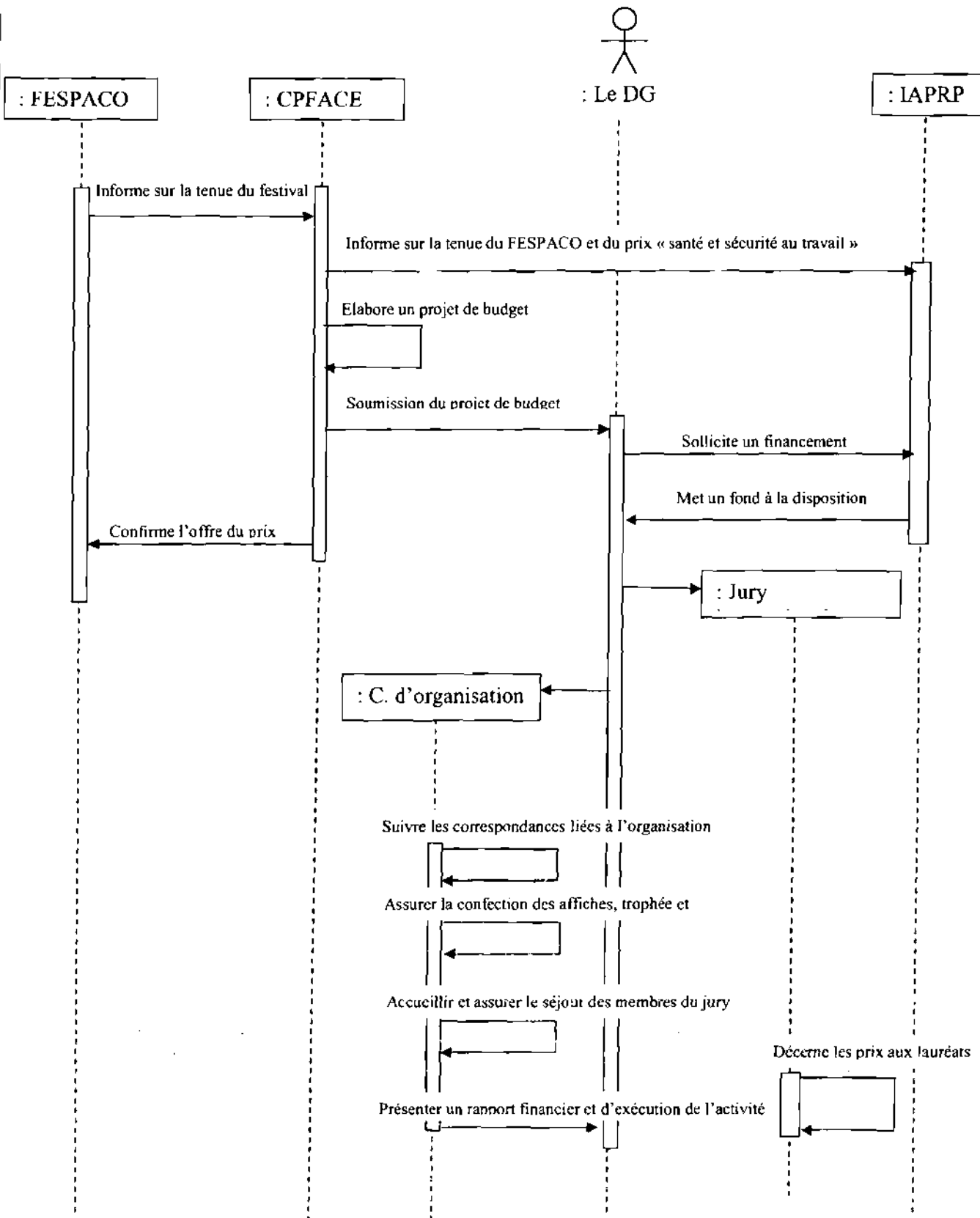


Diagramme2.10 Diagramme de séquence décrivant la procédure d'organisation d'un prix « SST »

Le diagramme de séquence décrivant la formation continue

Ce diagramme décrit la procédure de formation continue.

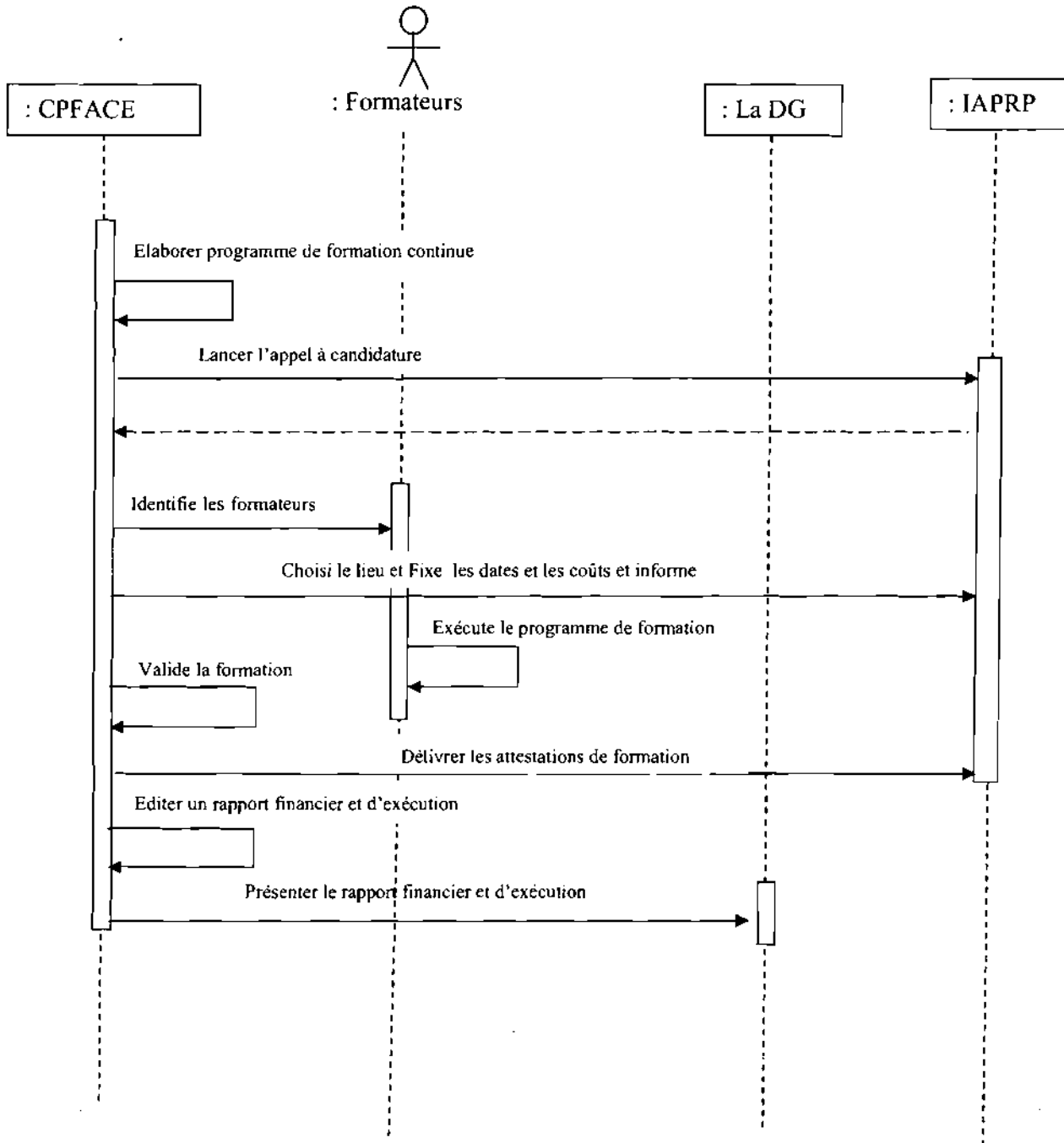


Diagramme2.11 Diagramme de séquence décrivant la procédure de gestion de la formation continue

2.3-Bilan critique

Il est question ici de faire une appréciation sur la gestion actuel des différentes activités menées dans le centre de production et de formation en audiovisuel et communication en entreprise.

Forces	<ul style="list-style-type: none"> • L'établissement des rapports à la fin de chaque SYAPRO et prix « Santé et Sécurité au Travail » du FESPACO permet d'avoir un suivi au fil des années de l'organisation des éditions du SYAPRO ; • L'établissement de recommandations après chaque édition de SYAPRO et du prix « Santé et Sécurité au Travail » permet à chaque édition de prendre des mesures pour relever les défis d'organisations déjà diagnostiqués.
faiblesse	<ul style="list-style-type: none"> • Le CPPFACE en tant que structure à vocation régionale n'a toujours pas d'ouverture pour ses partenaires ; • La laborieuse voie d'information orale est la plus utilisée jusqu'à présent ; • Les inscriptions au centre au stade actuel se font par voie de courrier, avec toute la lenteur liées à ce mode de contact ; • La gestion de la médiathèque interafricaine est confiée au CPPFACE et il ne dispose toujours pas d'un logiciel de gestion de la médiathèque ; • La documentation sur le SYAPRO et du prix « Santé et Sécurité au Travail » du FESPACO qui intéresse plus d'une personne n'est toujours pas accessible ; • Il n'y pas un suivi effectif de la formation.

Chapitre 3 Etude des scenarii

L'étude de l'existant nous a permis d'une part de nous imprégner des interactions et des flux d'informations dans le centre et d'autre part de faire ressortir certains dysfonctionnements dans certains aspects du fonctionnement du système et surtout d'appréhender les besoins et souhaits des utilisateurs du système.

Tout en rappelant les objectifs du futur système à mettre en place, nous allons, au cours de cette phase d'étude, proposer des solutions possibles, satisfaisant les objectifs et contraintes du domaine. La solution retenue, dans ce chapitre d'étude des scénarii, fera l'objet d'une étude détaillée à l'étape suivante.

3.1-Objectif du futur système d'information

Le système à mettre en place a pour but de répondre aux souhaits des intervenants du centre. Ainsi à travers l'application on devra être à mesure de :

- gérer la médiathèque africaine en matière de films et de multimédia sur la prévention des risques professionnels.
- tenir à jour et publier la liste des films disponibles dans la sous régions dans le domaine de préventions des risques professionnels
- permettre aux enseignants intervenant dans le centre d'envoyer des informations ou de la documentation
- publier toute la documentation liée au SYAPRO
- permettre aux structures des pays membres de l'IAPRP d'inscrire leurs agents en ligne.
- gérer les étudiants et agents inscrits pour les différentes formations ainsi que les enseignants.
- gérer les archives liées au prix « Santé et Sécurité au Travail » du FESPACO
- de répondre aux demandes de ses partenaires d'un site du CPFACE (notamment l'INRS de France), une demande qui paraît naturelle pour une structure à vocation sous régionale.
- d'assurer la sécurité et la confidentialité des données.

3.2-Reconfiguration du système d'information

Elle consiste à apporter des modifications et des ajouts au système d'information actuel dans le but d'améliorer son fonctionnement.

Le stockage des données relatives aux différentes manifestations à travers des rapports qui sont des supports fragiles se fera maintenant dans une base de données.

Pour faciliter l'inscription aux différentes formations qui se réalisait par voie de courriers, nous préconisons désormais l'inscription en ligne. Cela réduirait énormément les flux d'informations et de ce fait le coût et le temps de l'inscription.

Les emplois de temps, les fiches et listes des étudiants et enseignants s'éditeront automatiquement et de ce fait faciliterait le travail du coordonnateur des cours.

On notera une meilleure gestion de la médiathèque en ce sens que le nouveau système permettra l'enregistrement, la recherche des œuvres et le suivi des emprunts cela permettra un meilleur suivi des œuvres et une bonne structuration de la médiathèque.

3.3-Etude comparatives des outils logiciels et langages

3.3.1 Stockage et sauvegarde de la base de données

3.3.1.1 Le besoin

L'application doit être capable de stocker un ensemble d'informations concernant la gestion de la médiathèque, les différentes éditions du SYAPRO et du prix « Santé et sécurité au travail » etc. C'est pourquoi il est nécessaire qu'elle implémente une base de données. L'outil le plus efficace pour gérer des informations de manière structurée et centralisée reste une base de données. En effet, elle permet de stocker différents types de données qui seront accessibles de n'importe où sur un réseau. Un point très important est de choisir le système de gestion de base de données (SGBD) le plus adapté à nos besoins spécifiques. Une base de données possède des tables contenant les données que l'on souhaite stocker. Sur celles-ci on peut effectuer des requêtes, en utilisant un langage spécifique (SQL Standard Query Language), afin de gérer l'ensemble des données, c'est à dire les opérations de lecture, de suppression et de modification.

Le projet est une application multi-utilisateurs. Chacun d'entre eux pourra consulter et modifier les informations contenues dans la base en même temps. Il nous faut donc choisir un SGBD ayant une bonne gestion des accès concurrents. Les informations que nous souhaitons stocker dans cette base concernent les différents acteurs du CPFACE. Ce sont des informations concernant les enseignants, les étudiants du centre ou encore les structures membre des pays membres de l'IAPRP et qui ne doivent pouvoir être consulté seulement que par l'application. Il faut donc assurer un certain niveau de sécurité. Deux solutions gratuites: PostgreSQL et MySQL et une solution payante d'ailleurs adoptée déjà par la structure qui est ORACLE s'offrent à nous pour répondre aux différents besoins.

3.3.1.1 Les solutions proposées

❖ Comparatif MySQL / PostgreSQL /Oracle

Fonctionnalités	MySQL 5.0	PostgreSQL 8.1	Oracle 9i
Rapidité d'exécution	Rapide	Moyennement rapide	Très rapide
Requêtes Imbriquées	Oui	Oui	Oui
Transactions	Oui	Oui	Oui
Réplication de base	Oui	Oui	Oui
Support de clés étrangères	Oui	Oui	Oui
Vues	Oui	Oui	Oui
Procédures stockées	Oui	Oui	Oui
Triggers	Oui	Oui	Oui
Unions	Oui	Oui	Oui
Jointures complètes	Non	Oui	Oui
Contraintes	Non	Oui	Oui
Héritage de table	Non	Oui	Oui
Curseurs	Partiellement (seulement en lecture)	Oui	Oui
Supporte IPV6	Non	Oui	Oui
Driver JDBC	Oui	Oui	Oui

Tableau 3.1 comparatif MySQL/PostgreSQL/Oracle

❖ MySQL

Ce gestionnaire de bases de données développé par MySQL AB est aujourd'hui très répandu dans l'univers du web. La dernière version complète du SGBD est la 4.1 mais une version 5.0 beta existe déjà et complète les fonctionnalités manquantes de l'application.

Depuis la version 4.1 le MySQL intègre les requêtes imbriquées et les tables InnoDB permettant de gérer les transactions et les verrous au niveau des lignes. Les verrous ne sont cependant pas automatiques. En effet il faut utiliser une commande spécifique pour placer des verrous exclusifs (SELECT FOR UPDATE) pour ensuite lancer sa requête. Par la même occasion, le codage des caractères sur plusieurs octets (comme UTF-8) est maintenant disponible.

MySQL fonctionne sur beaucoup de plateformes différentes comme: AIX, BSDi, FreeBSD, HP-UX, Linux, Mac OS X, NetBSD, OpenBSD, OS/2, Warp, SGI Irix, Solaris, SunOS, SCO OpenServer, SCO UnixWare, Tru64, Unix, Windows 95, 98, NT, 2000 et XP. De plus, la plupart des langages utilisés aujourd'hui sont capable d'interagir avec MySQL.

MySQL offre de très bon résultat en termes de performances et rapidité d'accès. En effet, il est très rapide, multithread, robuste et multi-utilisateurs. C'est pourquoi il est très utilisé avec les technologies web.

Points forts

- solution très courante en hébergement public ;
- très bonne intégration dans l'environnement Apache/PHP ;
- open source, bien que les critères de licence soient de plus en plus difficiles à supporter ;
- version cluster depuis la version 4 ;
- facilité de déploiement et de prise en main ;
- plusieurs moteurs de stockage adaptés aux différentes problématiques.

Points faibles

- ne supporte qu'une faible partie des standards SQL-92 ;
- support incomplet des triggers et procédures stockées ;
- gestion des transactions que depuis la version 4 avec InnoDB⁶ ;
- assez peu de richesse fonctionnelle ;
- manque de robustesse avec de fortes volumétries ;
- pas d'héritage de tables.

❖ PostgreSQL

PostgreSQL est un système de gestion de base de données développé par PostgreSQL Global développement group. C'est un système très utilisé aujourd'hui qui continue à se développer pour répondre aux besoins spécifiques de nombreux utilisateurs. Ce système en est aujourd'hui à sa huitième version.

Ce système de base de données possède de nombreux avantages techniques :

• **requête imbriquée** : permet, lorsque l'on écrit une requête SQL, d'exprimer certaines de ses conditions par des sous requêtes. Cette méthode a pour avantage de simplifier et optimiser certains traitements.

• **transaction** : consiste à effectuer un ensemble de requêtes pour réaliser une opération.

Si l'une des requêtes de la transaction ne s'effectue pas correctement, la transaction est annulée.

⁶ InnoDB: c'est un moteur de stockage utilisé par MySQL pour créer, supprimer ou lire dans une table.

Ex : Les mouvements de comptes bancaires nécessitent plusieurs opérations indissociables (**débit + crédit**). On doit s'assurer qu'elles ont toutes été effectuées avant de valider la transaction. Dans le cas contraire on effectue un retour en arrière pour remettre les données dans leur état initial. Ceci est appelé **ROLLBACK**.

Une transaction commence par un « **BEGIN** » et finit par un « **COMMIT** ».

La gestion des verrous exclusifs sous PostgreSQL est automatique pour les requêtes **UPDATE** et **DELETE**. Ils peuvent être placés au niveau des tables mais aussi de leurs lignes pour permettre à d'autres transactions d'accéder aux lignes non verrouillées de la table.

- **vue** : permet de créer un schéma spécifique d'une table en sélectionnant certains champs. Cela permet de simplifier la vision que l'on peut avoir d'une table et ainsi simplifier certaines requêtes.

- **l'héritage de table** : l'idée de ce type de fonctionnalité provient des langages objets.

En effet il permet à une table d'hériter de tous les champs d'une autre table.

Exemple : si on prend une table Œuvre et une table Film. La table Film héritera de tous les champs présents dans la table Œuvre.

- **trigger** : Les triggers (aussi appelés déclencheurs en français) sont, au même titre que les contraintes, des éléments fondamentaux dans la structure des bases de données.

Alors qu'une contrainte permet de définir une règle algébrique, le trigger permet de définir une règle algorithmique. Il peut également être très utile d'un point de vue purement fonctionnel. Il correspond à une fonction affectée à une table et qui est déclenchée sur certaines opérations.

PostgreSQL propose donc de nombreuses fonctionnalités qui peuvent être très utiles pour le développement de l'application.

Points forts

- Open source et gratuit ;
- fiable et relativement performant, tout en restant simple d'utilisation ;
- supporte la majorité du standard SQL-92 et possède en plus un certain nombre d'extensions ;
- très riche fonctionnellement, notions d'héritage de tables, multitude de modules ;
- simple d'utilisation et d'administration ;
- héritage de tables.

Points faibles

- sauvegardes peu évoluées ;
- supporte les bases de moyenne importance ;
- pas de notion de rôles, pas de hiérarchisation de groupes ;
- pas de services Web ;
- pas de support XML ;
- permission seulement au niveau de la table, pas au niveau de la colonne ;
- solutions de réplication, de cluster pas encore totalement packagées

❖ Oracle

Oracle n'est pas un SGBDRO optimisé pour de petites bases de données sur de petits volumes de traitements (2 Go par exemple) et peu d'utilisateurs (une trentaine). On peut trouver des benchmark⁷ où MySQL offre des performances quasi comparables à Oracle. Si l'on monte à de plus importants volumes de donnée (>200Go) et un grand nombre

⁷ Un benchmark est un banc d'essai permettant de mesurer les performances d'un système pour le comparer à d'autres.

d'utilisateurs (>300) les écarts de performance entre un MySQL et un Oracle, Sybase, Db2 seront très visibles.

Disponibilités : Linux, Windows, Unix.

Points forts

- procédures stockés en PL/SQL⁸ (langage propriétaire Oracle, orienté ADA) ou en JAVA (depuis la 8.1.7) ce qui peut s'avérer utile pour les équipes de développement ;
- assistants performants via Oracle Manager Server, possibilité de gérer en interne des tâches et des alarmes ;
- gestion centralisée de plusieurs instances ;
- concept unique de retour arrière (Flash-back) ;
- pérennité de l'éditeur : avec plus de 40% de part de marché ;
- réglages fins : dans la mesure où l'on connaît suffisamment le moteur, presque tout est paramétrable ;
- accès aux données système via des vues, bien plus aisément manipulable que des procédures stockées ;
- interface utilisateur remaniée et extrêmement riche, permettant le tuning fin de requêtes par modification des plans d'exécution ;
- services Web ;
- support XML.

Points faibles

- prix exorbitant, tant au point de vue des licences que des composants matériels (RAM, CPU) à fournir pour de bonnes performances ;
- fort demandeur de ressources : ce qui n'arrange rien au point précité, Oracle est bien plus gourmand en ressource mémoire que ses concurrents, ce qui implique un investissement matériel non négligeable. La connexion utilisateur nécessite par exemple près de 700 Ko/utilisateur, contre une petite centaine sur des serveurs MS-SQL ou Sybase ASE ;
- gestion erratique des rôles et privilèges (pas possible de donner des droits sur des schémas particuliers sans passer par leurs objets, désactivation des rôles lors d'exécution de packages...) ;
- opacité de l'éditeur. Même au niveau des cours donnés par Oracle, il est quasi impossible d'obtenir des informations concernant les fonctionnements internes du moteur.

3.3.1.1- La solution retenue : Oracle 9.i

En termes de performances, MySQL semble avoir un atout majeur par rapport à PostgreSQL avec des temps d'accès à la base plus rapides. En contre partie, PostgreSQL offre un panel de fonctionnalités plus importantes même si l'écart entre les deux SGBDs se réduit petit à petit d'une version à l'autre. Oracle quant à lui cumule performance rapidité et robustesse. Le choix à faire se situe donc entre rapidité, fonctionnalités et robustesse même si son coût d'acquisition reste relativement élevé.

Nous avons décidé d'utiliser la base de données Oracle. En effet, nous allons utiliser l'héritage de table. Les SELECT imbriqués sont plus intuitifs à utiliser que MySQL et PostgreSQL et, de plus, la CNSS a déjà adopté Oracle.

⁸ SQL : Structured Query Language, Langage structuré de requêtes.

3.3.1- Langages de programmation

3.3.1 .1-Besoins

Le projet, en accord avec le groupe de supervision se réalisera avec des solutions orientées objets car étant plus adaptées et convenant mieux à ce genre de projet.

Plusieurs outils logiciels et langages s'offrent à nous pour sa réalisation; alors quels choix faire ?

Avant de procéder à nos choix techniques, il est important de mener une étude comparative sur ces solutions, afin de bien guider nos choix.

3.3.1 .2- Solutions proposées

❖ Java

Présentation

Java est une technologie composée d'un langage de programmation orienté objet et d'un environnement d'exécution. Préalablement nommé **Oak**, il a été créé par James Gosling et Patrick Naughton chez Sun Microsystems avec le soutien de Bill Joy.

Java est à la fois un langage de programmation et une plate-forme d'exécution. Le langage Java a la particularité principale d'être portable sur plusieurs systèmes d'exploitation tels que Windows, Mac OS ou Linux. C'est la plate-forme qui garantit la portabilité des applications développées en Java.

Java permet de développer des applications autonomes mais aussi, et surtout, des applications client-serveur. Côté client, les Applets sont à l'origine de la notoriété du langage. C'est surtout côté serveur que Java s'est imposé dans le milieu de l'entreprise grâce aux Servlets, et plus récemment les JSP (Java Server Pages) qui peuvent se substituer à PHP et ASP (Application Services Pages).

Quatre caractéristiques militent fortement en faveur de Java depuis sa création et lui confèrent une notoriété remarquable dans le monde des langages de programmation. Il :

- est orienté pur objet ;
- est indépendant de la plate-forme du client ;
- contient des aides et des bibliothèques pour le réseau informatique;
- a la capacité d'exécuter du code source extérieur de façon sécurisée.

Points forts

- simple et familier ;
- indépendant de l'architecture ;
- orienté objet ;
- distribué ;
- interprété et compilé;
- robuste et sûr ;
- portable ;
- dynamique ;
- sécurisé : les traitements de Java s'exécutent dans un espace mémoire protégé de la machine virtuelle : la SandBox ;
- mode de diffusion gratuit ;
- multithread.

Point faible

- demande beaucoup en ressources matérielles.

❖ C#

Présentation

Le C# est un langage de programmation orienté objet à typage fort, créé par la société Microsoft, et notamment un de ses employés, Anders Hejlsberg⁹.

Il a été créé afin que la plate-forme .NET soit dotée d'un langage permettant d'utiliser toutes ses capacités. Il est très proche du Java dont il reprend la syntaxe générale ainsi que les concepts (la syntaxe reste cependant relativement semblable à celles de langages tels que le C++ et le C). Un ajout notable à Java est la possibilité de surcharge des opérateurs, inspirée du C++.

Le C# est, d'une certaine manière, le langage de programmation qui reflète le mieux l'architecture .NET qui fait fonctionner toutes les applications .NET, et en est par conséquent extrêmement dépendant. Le CLR (Common Language Runtime) est obligatoire pour exécuter des applications écrites en C#, comme l'est la JVM (Java Virtual Machine, ou Machine virtuelle Java) pour des applications Java.

Points forts

- support complet des classes de programmation orientées objet, avec interface et héritage d'implémentation, les fonctions virtuelles et la surcharge d'opérateur ;
- un jeu de types de base cohérent et bien défini ;
- le support interne pour la génération d'une documentation XML ;
- le nettoyage automatique de la mémoire allouée dynamiquement ;
- possibilité de marquer les classes ou les méthodes avec des attributs définis par l'utilisateur ;
- accès complet à la bibliothèque .NET des classes de base, accès aisé à l'API Windows;
- support des propriétés et des événements dans le style de Visual Basic.

Points faibles

- difficulté de faire face à des temps d'exécution critiques et assurer des performances très élevées du code;
- impossibilité de spécifier des fonctions en ligne et des destructeurs garantissant leur exécution à certains points précis du code.

3.3.1.2- Solution retenue

Nous avons décidé d'utiliser le langage JAVA. En effet, C# offre un support complet de classes de programmation orientées objet, avec interface et héritage d'implémentation, des fonctions virtuelles et la surcharge d'opérateur ainsi que de jeu de types de base cohérent et bien défini avec un support interne pour la génération d'une documentation XML. Mais ce n'est pas leur unique particularité. Il fournit de nombreuses fonctions pour développer des applications partagées et demande une logique de développement particulière. Nous ne pensons pas n'ont plus avoir besoin d'exploiter toutes les fonctions offertes par C#.

En effet, les fonctionnalités importantes qu'offre JAVA sont connues depuis plus longtemps, donc activées à l'école. C# est un peu « jeune » et nous préférons partir sur

⁹ Anders Hejlsberg est un programmeur Danois. En 1980, il a commencé à écrire des programmes pour le micro-ordinateur Nascom ; il a en particulier écrit un compilateur Pascal qui a été vendu sous le nom de Blue Label Pascal compiler pour le Nascom-2

des bases solides et éprouvées. Pour le développement avec java , il existe plusieurs plateforme .Quel plateforme allons donc adopter ?

3.3.1.2.1- Plateforme de développement

3.3.1.2.1.1- Le besoin

Pour le développement de notre application, nous avons besoin d'une plateforme de développement Java pour la partie métier de notre application. De la même façon nous avons besoin d'un module pour développer l'interface graphique, pour la partie web de l'application.

3.3.1.2.1.2- Les solutions proposées

❖ Comparatif JBuilder/Eclipse/NetBeans

Fonctionnalités	JBuilder	Eclipse	NetBeans
Vérification syntaxique	Oui	Oui	Oui
Complétion du code	Oui	Oui	Oui
Compilation du code	Oui	Oui	Oui
Exécution du code	Oui	Oui	Oui
Débogage du code	Oui	Oui	Oui
Intégration de Ant	Oui	Oui	Oui
Intégration de CVS	Oui	Oui	Oui
Intégration de JUnit	Oui	Oui	Oui
Intégration de builders d'interfaces graphiques	Oui	Oui mais composant séparé	Oui
Intégration d'un support de développement Web	Oui	Non	Oui
Refactoring ¹⁰ automatique	Oui	Oui	Non
Extensible par plug-ins	Oui	Oui	Oui

Tableau 3.2 Comparatif JBuilder/Eclipse/NetBeans

❖ Borland JBuilder 2006 Developer

Borland JBuilder 2006 est l'un des meilleurs environnements professionnels pour le développement de solutions Java. Il intègre, dans une interface agréable, tous les concepts d'ingénierie moderne, WebServices, XML, Travail collaboratif, plugin Mobilset pour écrire des applications pour terminaux mobiles, tests unitaires, refactoring, déboguer HotSwap, conception d'EJB 1.1 et 2.0, JSP/Servlet et Struts, Ant, outils de productivité collective (TeamSource, ClearCase, CVS,...), optimisation avec OptimizeIt et outil pour UML. Couplé avec Together Control Center et Borland Entreprise Server, JBuilder améliorera de façon quantitative et qualitative la productivité.

Points forts

- accélère les développements Web grâce à ses modules visuels et bidirectionnels de conception JSF et Struts et à ses solutions de productivité JDK™ 5.0 ;
- améliore la qualité du code grâce à son système avancé de profilage mémoire et CPU et facilite les développements distribués grâce à ses nouvelles fonctionnalités de collaboration sur le code (édition partagée et débogage commun) ;

¹⁰ La refactorisation (anglicisme venant de refactoring) est une opération de maintenance du code informatique.

- facilite le développement de projet graphique en offrant un « designer » pour cela ;
- JBuilder est le RAD idéal pour les projets professionnels de grande envergure, impliquant l'interaction avec des serveurs d'applications comme BEA, BES, SunOne ou encore JBoss.
- **Systèmes d'exploitation supportés** : Windows, GNU/Linux, Solaris

Point faible

- Très gourmand en ressource système, nécessite une mémoire RAM de 256Mo minimum.

❖ Eclipse-SDK-3.1.2

Eclipse est une plateforme de développement écrite en Java, fruit du travail d'un consortium de grandes entreprises (IBM, Borland, Rational Rose, HP...). Il en résulte un IDE performant et OpenSource qui a su trouver sa place parmi les pointures du marché que sont JBuilder et NetBeans. Elle intègre pour cela la prise en charge des outils comme Ant, CVS, Junit...

Au niveau ergonomie, Eclipse n'as rien à envier à ses concurrents. Toutes les fonctionnalités indispensables sont là : création de projet, de template, refactoring, debuggage ... et remarquablement facile à prendre en main. Eclipse tire sont originalités dans le choix des composants graphiques utilisés lors de son développement. En effet, Eclipse ne contient pas d'AWT, ni de SWING mais uniquement des composants SWT/JFace (composants développé par IBM). Ainsi, l'interface gagne en rapidité et en convivialité.

Points forts

- ouverture de son noyau qui permet l'ajout de très nombreux plugins ;
- moins gourmand en espace mémoire ;
- n'intègre pas de JDK spécifique car paramétrage avec n'importe qu'elle version de JDK ;
- espace utilisateur bien agréable ;
- très facile à installer car il suffit juste décompresser l'archive sur une partie du disque.
- **Systèmes d'exploitation supportés** : Windows, GNU/Linux, Solaris.

Point faible

- Difficulté de prise en main pour les débutants du fait qu'il n'intègre pas beaucoup de composants le rendant prêt à l'emploi.

❖ NetBeans 5.5.1

NetBeans, créé à l'initiative de Sun Microsystems (Noyau de Forte4J/SunOne), présente toutes les caractéristiques indispensables à un RAD Java de qualité. De licence open source, NetBeans permet de développer et déployer rapidement et gratuitement des applications graphiques Swing, des Applets, des JSP/Servlet, des architectures J2EE, dans un environnement fortement personnalisable.

NetBeans 5.5.1 propose entre autres :

- système de projets basé sur Ant : ce nouveau système de gestion de projets va permettre une meilleure indépendance et une portabilité des travaux ;

- refactoring : ajout de nouvelles fonctionnalités comme le renommage, le déplacement et la recherche des déclarations pour classes, méthodes et packages. ;
- support de J2SE 5.0 (Tiger) : support des nouveaux mots clefs du langage, des annotations, des génériques, etc. ;
- développement des applications Web : améliorations pour l'édition des JSP, la gestion serveur, le support des dernières versions de Tomcat, etc. ;
- support pour J2ME : support des standards MIDP 2.0 et CLDC 1.1, possibilités d'intégrations d'émulateurs extérieurs, etc. ;
- améliorations du débogueur ;
- améliorations de l'interface graphique.

Points forts

- facilité de prise en main pour les débutants ;
- facilité de réalisation d'interface graphique avec Matisse, un éditeur WYSIWYG¹¹ ;
- facilité d'ajout de nouvelle fonctionnalité via un système de module téléchargeable et installable depuis l'interface du programme ;
- système de plugins performant ;
- possibilité de développement de modules pour le logiciel.
- **Systèmes d'exploitation supportés** : Windows, GNU/Linux, Solaris

Point faible

- très gourmand en espace mémoire.

3.3.1.2.1.2-Solution retenue : NetBeans5.5.1

Les trois IDE présentés possèdent bien plus de similarités que de différences. Ainsi, ils proposent tous les trois l'ensemble des fonctionnalités nécessaires au développement de notre application. Néanmoins, après quelques essais de ces IDE, **NetBeans** apparaît être plus simple d'utilisation, plus rapide et plus stable. C'est pourquoi, nous l'avons retenu comme environnement de programmation pour le développement de l'application.

¹¹ WYSIWYG: What You See Is What You Get, désigner les interfaces utilisateurs graphiques permettant de composer visuellement le résultat voulu.

3.4-Architecture réseau

3.4.1- Les symboles utilisés

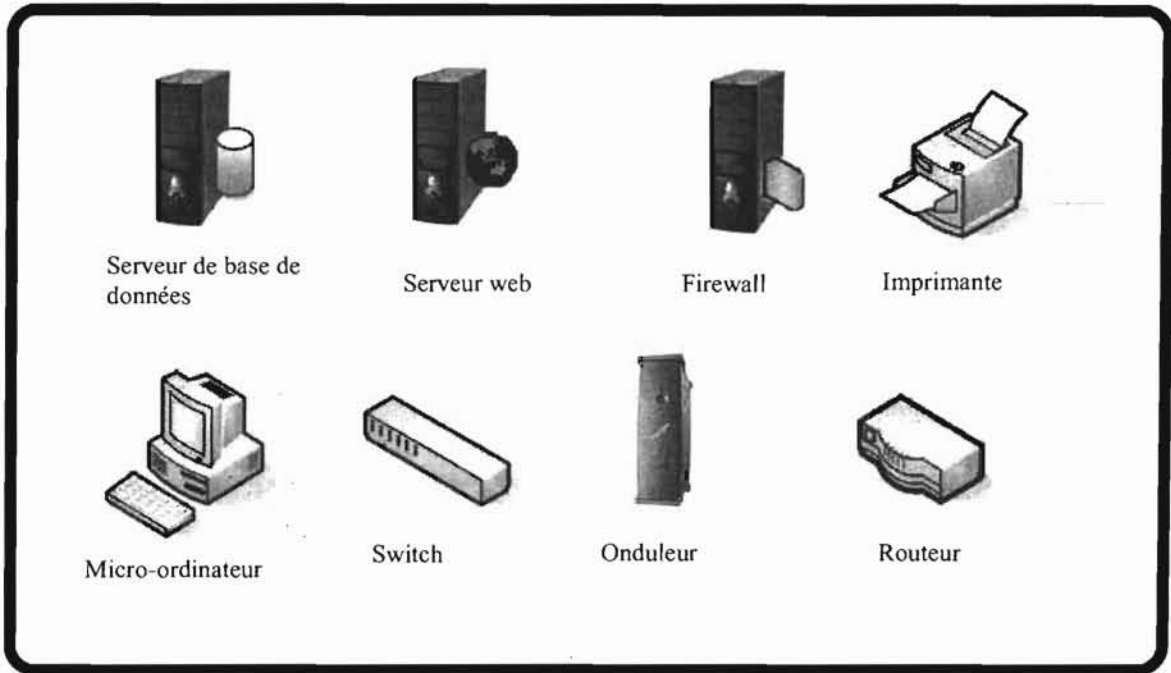


Figure 3.1 les symboles utilisés

3.4.2-Architecture

le CPFACE ne dispose pas actuellement d'un réseau informatique.

existante

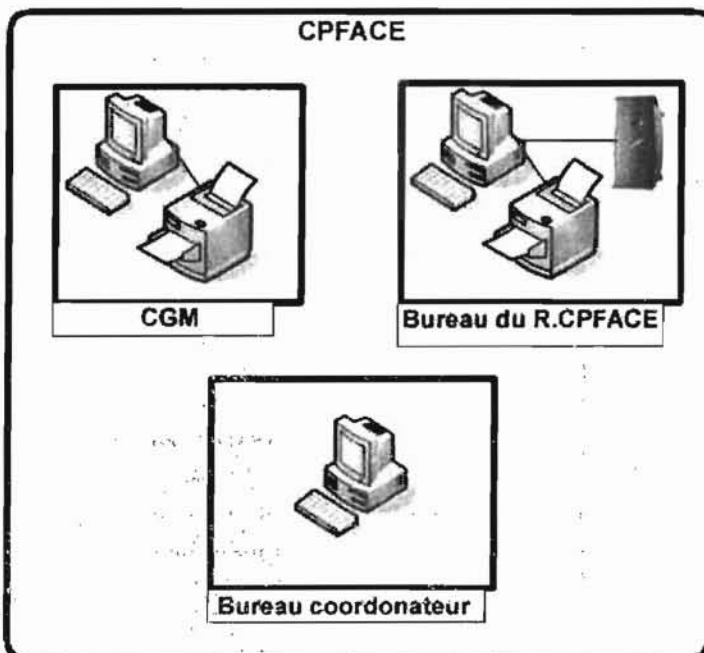


Figure3.2 Ressources existantes

3.5-Modèle de calcul des coûts de réalisation

L'estimation du coût total d'un projet logiciel comprend le coût de développement du logiciel et du matériel, le coût de la formation, le coût des outils... Nous nous limitons ici au coût de développement du logiciel qui sera calculé en fonction du temps passé à développer celui-ci par les ingénieurs dont on connaît le salaire moyen. Les coûts du développement logiciel sont donc étroitement liés à la notion de productivité des ingénieurs logiciels, c'est à dire le nombre de lignes de code (validées) produites par unité de temps. Le modèle de calcul dont les paramètres sont adaptables aux contextes de développement est le modèle COCOMO.

La méthode COCOMO, pour CONstructive COSt Model a été développée par Dr. Barry Boehm pour estimer l'effort et le temps de développement d'un produit logiciel. A l'origine elle a été construite à partir d'une analyse des données par régression pratiquée sur 63 projets logiciels (gestion et informatique industrielle) comprenant de 2000 à 100.000 lignes de code dans l'entreprise TRW (USA). COCOMO a l'avantage d'être un modèle ouvert. Les données de calibrage, les formules et tous les détails des définitions sont disponibles.

Le modèle COCOMO s'appuie sur une estimation de l'effort en homme*mois (HM) calculée par la formule :

- Effort = C*M*taille*s
- C facteur de complexité
- M facteur multiplicateur
- Taille en milliers de lignes de code source livrées (KDSI)
- s proche de 1

Hypothèses :

KDSI livré :

- Exclut en général les environnements de tests, les supports de développement
- Exclut les commentaires
- Inclut le Shell

HOMMES MOIS (HM)

- MM = 152 Heures (Normes américaines), tient compte des vacances, arrêts maladie...

Depuis 1981, le modèle COCOMO est constitué de trois modèles :

- le modèle de base
- le modèle intermédiaire
- le modèle détaillé ou expert

Nous ne présenterons que les grandes lignes du modèle qui cadre bien le contexte de notre projet.

Le modèle de base

Le modèle de base estime l'effort (le nombre d'homme mois) en fonction du nombre de milliers d'instructions source livrées (KDSI), de la productivité (le nombre de lignes de code par personne par mois) et d'un facteur d'échelle qui dépend du type de projet.

Les 3 types de projet identifiés sont :

• Projet de mode organique :

Il est défini par une innovation minimale, une organisation simple en petites équipes expérimentées. (Ex: petite gestion, système de notes dans une école, petits systèmes d'exploitation, traducteurs)

- **Projet de mode médian (semi-détaché) :**

Il est défini par un degré d'innovation raisonnable, (exemples: Compilateurs, Contrôle de processus simple, système bancaire interactif)

- **Projet de mode embarqué**

Dans ces projets le niveau d'innovation est important, l'organisation complexe, couplage fort avec beaucoup d'interactions, (exemples: Gros système d'exploitation, Systèmes transactionnels complexes, système de contrôle aérospatial...)

Type de projet	Effort Homme Mois (HM)	Temps de développement (TDev)
Organique	2.4(KDSI) ^{1.05}	2.5(HM) ^{0.38}
médian	3.0(KDSI) ^{1.12}	2.5(HM) ^{0.35}
embarqué	3.6(KDSI) ^{1.20}	2.5(HM) ^{0.32}

Tableau 3. 3 les coûts du modèle de base

3.6-Présentations des scenarii

3.6.1 Premier scénario

Ce scénario consiste en la mise en place de deux applications client/serveur distinctes mais complémentaires. Une application 3-tiers, accessible à partir d'un navigateur adressée aux internautes et aux différentes partenaires du centre. La seconde est une application 2-tiers qui sera installée sur chacun des postes des agents du CPFACE (le responsable, le coordonnateur, le chargé de la gestion de la médiathèque) qui permettra de gérer les différentes activités du centre. Nous prévoyons la mise en place d'un réseau au sein du CPFACE.

- **Architecture du premier scénario**

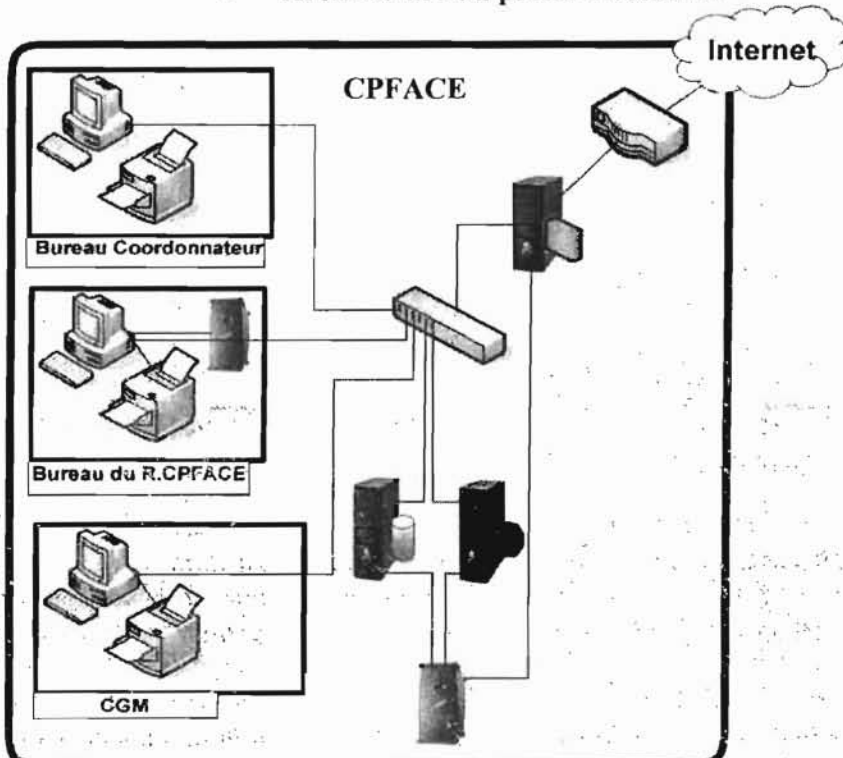


Figure 3.3 Architecture réseau du premier scénario

- **Besoins matériels et Logiciels**

Désignations	Caractéristiques	Quantité	Prix unitaire (FCFA)	Montant (FCFA)
Serveur	Marque et modèle : ALPHA DS25 Système d'exploitation : HP TRU64 UNIX V5.B Mémoire RAM : 4 GB Processeur : 2 Alpha 64 bits de 2 GHz et 4 MB cache chacun Disques durs : 02 disques ultra SCSI de 72,3 GB chacun	02	existant	existant
micro-ordinateur	Processeur : 3.2 GHz Disque dur : 70 GO Mémoire RAM : 448 Mo Lecteur CD, Carte réseau intégré Écran : 15"	03	existant	existant
imprimante	Marque et modèle : HP	03	existant	existant
Onduleur	Marque et modèle : LIEBERT 80 KVA			
Routeur	Marque et modèle : CISCO 1721	01	750 000	750 000
Switch	Marque et modèle : 8 port 10/100Base T ECO	01	35 000	35 000
Antivirus	Kaspersky Anti-virus (7.0.0.125)	01	Gratuit	Gratuit
SGBD	Oracle 9.i	01	existant	existant
IDE	NetBeans 5.5.1	01	Gratuit	gratuit
Total			785 000	

Tableau 3. 4 besoins matériels et logiciels du premier scénario

- **Evaluation des coûts**

- **Coût de développement**

Pour ce scénario, les formules du mode médian ou semi-détaché s'adaptent le mieux. En effet dans notre application, nous énumérons une centaine de classes, chaque classe ayant en moyenne 100 instructions sources. Nous aurons alors par application des valeurs approximatives :

$$-HM : 3*(60*100/1000)^{1.12}=22.31 \text{ Homme /Mois}$$

$$-TDev : 2.5 (22.31)^{0.35}=7.41 \text{ mois}$$

$$-\text{cout total} : 22.31*200\ 000 = 4\ 462\ 000.$$

- **Coût de la formation**

Prix de l'horaire (FCFA)	Nombre d'heure par utilisateur	Nombre d'utilisateurs	Montant (FCFA)
3000	10	4	120000

Tableau 3. 5 Coût de la formation du premier scénario

- **Coût total de la mise œuvre**

Désignation	Montant (FCFA)
Coût matériels et logiciels à acquérir	785 000
Coût de développement	4 462 000
Coût de la formation	120 000
Coût de mise en œuvre du réseau	200 000
Coût total du scénario	5 567 000

Tableau 3. 6 Coût total de la mise en œuvre

- **Critique du scénario premier scénario**

Avantage :

• Système d'information facile à sécuriser ;

- Facilité d'utilisation car respectant l'organisation actuelle du CPFACE ;
- Disponibilité permanente de la base de données au différent poste de travail ;
- Centralisation des données au niveau du serveur offrant ainsi une possibilité de sauvegardes ;
- Mise à jour de la base de données en temps réel.
- Vitesse de transfert de données élevée.

Inconvénients :

- Développement assez complexe.
- Mise en œuvre assez coûteux.

3.6.2- Deuxième scénario

Nous prévoyons dans ce scénario la mise en place d'une seule application 3-tiers accessibles aussi bien aux internautes qu'aux utilisateurs internes au CPFACE via un navigateur web. Ce scénario soulève des problèmes de sécurité du fait de son ouverture totale sur Internet, par conséquent l'architecture réseau préconisée devra être à la hauteur. Nous prévoyons l'interconnexion du CPFACE au DIS par une liaison Boucle Locale Radio (BLR).

• **Architecture du deuxième scénario**

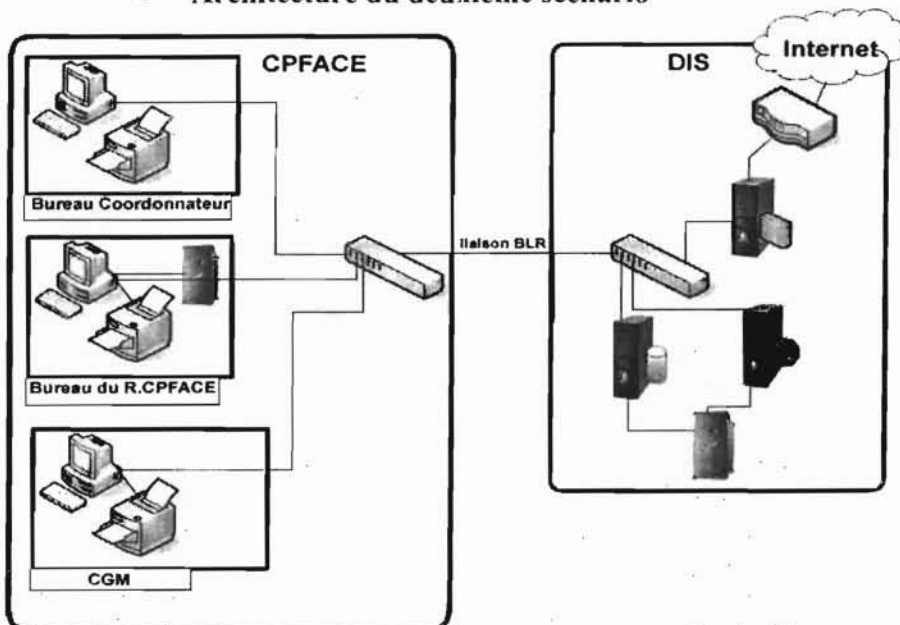


Figure 3.4 Architecture réseau du deuxième scénario.

- **Besoins matériels et Logiciels**

Désignations	Caractéristiques	Quantité	Prix unitaire (FCFA)	Montant (FCFA)
Serveur	Marque et modèle : ALPHA DS25 Système d'exploitation : HP TRU64 UNIX V5.B Mémoire RAM : 4 GB Processeur : 2 Alpha 64 bits de 2 GHz et 4 MB cache chacun Disques durs : 02 disques ultra SCSI de 72,3 GB chacun	02	existant	existant
micro-ordinateur	Processeur : 3.2 GHz Disque dur : 70 GO Mémoire RAM : 448 Mo Lecteur CD, Carte réseau intégré Ecran : 15"	03	existant	existant
imprimante	Marque et modèle : HP	03	existant	existant
Onduleur	Marque et modèle : LIEBERT 80 KVA			
Routeur	Marque et modèle : CISCO 1721	01	750 000	750 000
Switch	Marque et modèle : 8 port 10/100Base T ECO	01	35 000	35 000
Antivirus	Kaspersky Anti-virus (7.0.0.125)	01	Gratuit	Gratuit
SGBD	Oracle 9.i	01	existant	existant
IDE	NetBeans 5.5.1	01	Gratuit	gratuit
Total				785 000

Tableau 3.7 besoins matériels et logiciels du premier scénario

- **Evaluation des coûts**
- **Coût de développement**

Pour ce scénario, les formules du mode médian ou semi-détaché s'adaptent le mieux. En effet dans notre application, nous énumérons une centaine de classes, chaque classe ayant en moyenne 100 instructions sources. Nous aurons alors par application des valeurs approximatives :

$$\text{-HM} : 3 * (70 * 100 / 1000)^{1.12} = 26.52 \text{ Homme / Mois}$$

$$\text{-TDev} : 2.5 (26.52)^{0.35} = 7.84 \text{ mois}$$

$$\text{-cout total} : 26.52 * 200\ 000 = 5\ 304\ 000.$$

- **Coût de la formation**

Prix de l'heure (FCFA)	Nombre d'heure par utilisateur	Nombre d'utilisateurs	Montant (FCFA)
3000	20	4	240 000

Tableau 3.8 coût de la formation du premier scénario

- **Coût total de la mise œuvre**

Désignation	Montant (FCFA)
Coûts matériels et logiciels à acquérir	785 000
Coût de développement	5 304 000
Coût de la formation	240 000
Coût de mise en œuvre du réseau	200 000
Coût total du scénario	6 529 000

Tableau 3.9 coût total de la mise en œuvre

- **Critique du scénario**

Avantage :

- Facilité d'utilisation car respectant l'organisation actuelle du CPFACE ;
- Disponibilité permanente de la base de données au différent poste de travail ;
- Centralisation des données au niveau du serveur offrant ainsi une possibilité de sauvegardes régulières ;
- Mise à jour de la base de données en temps réel de l'intérieur comme de l'extérieur.

Inconvénients :

- Système d'information difficile à sécuriser ;
- Développement assez complexe.
- Mise en œuvre coûteux
- Risque d'intrusion et d'attaque très élevés étant donné qu'internet est un réseau public.
- Vitesse de transfert de données faible.

3.7-Scénario retenu

- **Etude comparative des scénarii proposés**

Les deux scénarii ci-dessus décrits répondent au besoins des utilisateurs tout en tenant compte des contraintes imposées du point de vue techniques et du point de vue des objectifs visés par le système à mettre en place. Le résumé comparatif des caractéristiques des scénarii est décrit dans le tableau suivant :

Critères Scénarii	Efficacité de traitement	Sécurité des données	Coût de mise en œuvre	Exploitation de l'application
Scénario 1	****	***	**	Facile
Scénario 2	***	**	***	Facile

Tableau 3.10 résumé comparatif des caractéristiques des scénarii

Légende : Faible *
 Moyen **
 Elevé ***
 Très élevé ****

En considérant les caractéristiques de chacun de ces scénarii, le groupe d'utilisateur a opté pour la mise en œuvre du premier scénario du fait qu'il présente des avantages notables.

Chapitre 4 Etude du futur système d'information

4.1-Phase d'élaboration

L'élaboration reprend les éléments de la phase d'analyse des besoins et les précise pour arriver à une spécification détaillée de la solution à mettre en œuvre.

L'élaboration permet de préciser la plupart des cas d'utilisation, de concevoir l'architecture du système et surtout de déterminer l'architecture de référence.

4.1.1 Expression des besoins

4.1.1.1 Le diagramme de cas d'utilisation

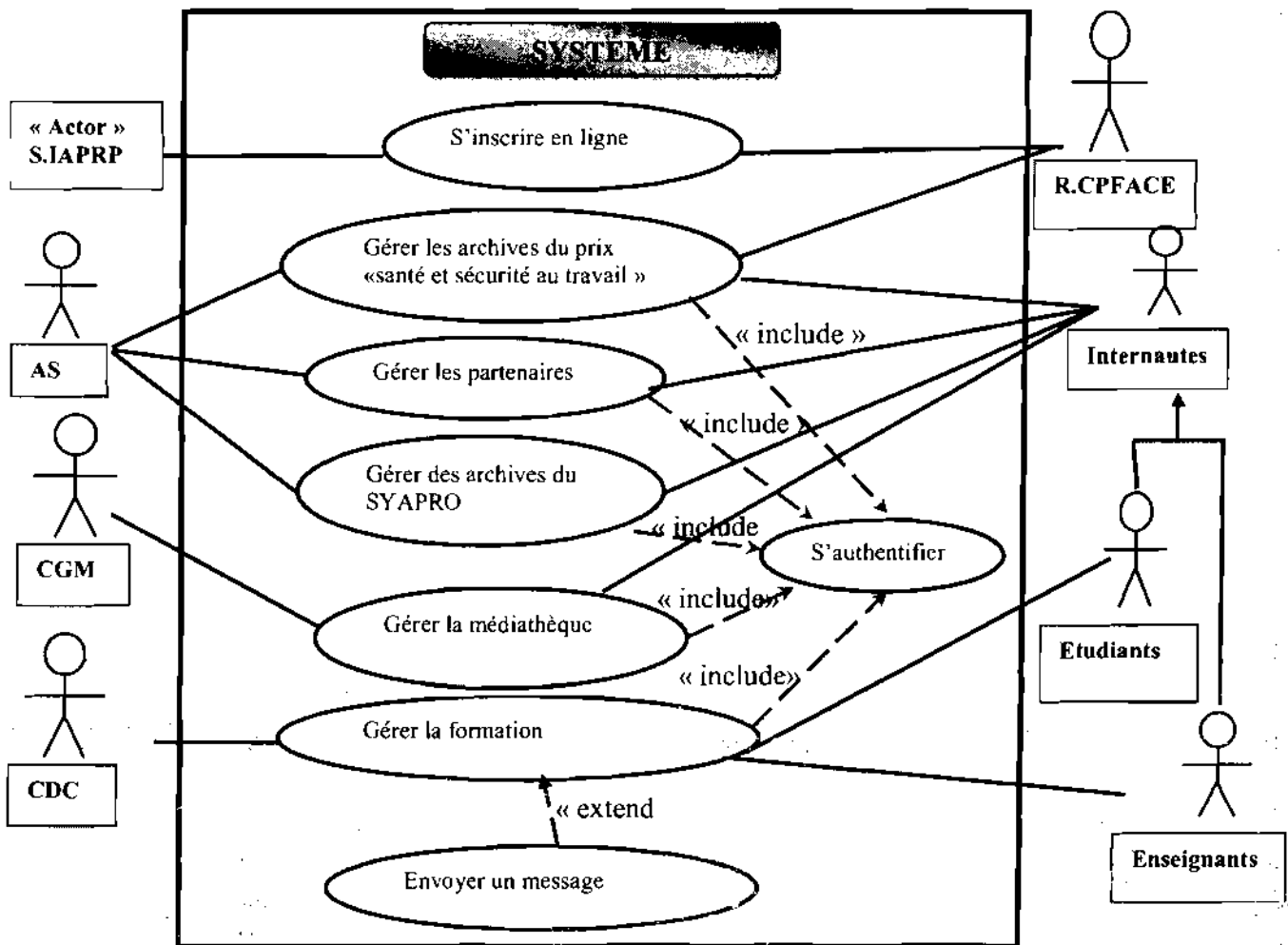


Diagramme4.1 Diagramme de cas d'utilisation du futur système

4.1.1 .2-Fiches descriptives de chaque cas d'utilisation

- **Formalisme adoptée pour la description textuelle des cas d'utilisation(CU)**

Pour une meilleure lisibilité nous avons adopté le formalisme suivant :

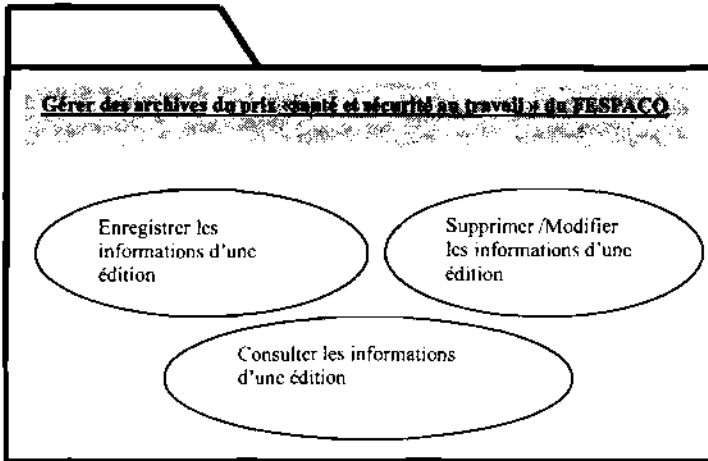
CU_i : « Nom du cas d'utilisation»		N°du tableau concernant le CU_i	
Résumé du CU_i		Pré condition	
Type de scénario/règle d'organisation et de gestion	N° de la version	Date de réalisation	
Scénario nominal: 01 : Opération1 02 : Opération1 03 : Opération1 04 : Opération1 . . N : OpérationN			
Extensions : 01. A : Condition d'application de l'extension A sur l'opération1 01. A1 : Opération1 de l'extension A sur l'opération 1 01. B : Condition d'application de l'extension B sur l'opération1 01. B1 : Opération1 de l'extension B sur l'opération 1			
Règle de gestion et d'organisation :			

- **Description des cas d'utilisation**

- ❖ **Le cas d'utilisation « S'authentifier »**

CU_1: S'authentifier		Folio 1/1	
Résumé : Ce CU permet aux utilisateurs de se connecter au système		Pré condition : le système fonctionne correctement	
Scénario nominal	Version : 1.0	Date de réalisation : 28/09/2007	
L'administrateur, le R.CPFACE, le coordonnateur, le chargé de gestion de la médiathèque, les enseignants, les étudiants			
Description du scénario nominal « Début » 01 : l'utilisateur demande à se connecter ; 02 : le système l'invite à entrer son nom d'utilisateur et son mot de passe ; 03 : l'utilisateur entre son nom d'utilisateur et son mot de passe ; 04 : le système vérifie le nom utilisateur et le mot de passe saisis ;(A) 05 : le système ouvre l'espace de travail correspondant au profile de l'utilisateur. « Fin »			
Extensions : 04. A : le nom d'utilisateur et le mot de passe sont -ils corrects ? 04. A1 : Si oui on continue au point 05 du scénario nominal. 04. A2 : Si non le système informe l'utilisateur de l'incohérence des paramètres de connexion et le scénario reprend au point 02 du scénario nominal ;			
Règle de gestion et d'organisation : Seul l'administrateur du système peut attribuer ou retirer les droits à un utilisateur ; On ne peut faire une mise à jour sans s'authentifier			

❖ Le cas d'utilisation « Gérer des archives du prix «santé et sécurité au travail » »



- Cas d'utilisation « Enregistrer les informations d'une édition du prix « Santé et sécurité au travail »

CU_2: Enregistrer information d'une édition		Folio 1/3
Résumé : Ce CU décrit la procédure d'enregistrement des informations d'une édition du prix « Santé et sécurité au travail		Pré condition : le système fonctionne correctement et on suppose que les rapports de l'édition en question sont disponibles
Scénario nominal	Version : 1.0	Date de réalisation : 28/09/2007
L'administrateur du système		
Description du scénario nominal		
« Début »		
01 : inclusion du cas d'utilisation « S'authentifier » ;		
02 : le système affiche le menu ;		
03 l'administrateur choisi l'option « Enregistrer les informations d'une édition » ;		
04 : le système affiche le formulaire d'enregistrement		
05 : l'administrateur saisit les informations sur la nouvelle édition ;		
06 : le système vérifie les informations saisies ; (A)		
07 : le système enregistre les informations dans la base de données;(B)		
08 : le système confirme l'enregistrement de l'édition;		
« Fin »		
Extensions :		
06. A : le formulaire est-il bien rempli?		
06. A1 : Si oui on continue au point 07 du scénario nominal.		
06. A2 : Si non le système informe l'administrateur de l'incohérence des valeurs saisies et le scénario reprend au point 05 du scénario nominal ;		
07. B : l'enregistrement respect-il les contraintes d'intégrité ?		
07. B1 : Si oui on continue au point 08 du scénario nominal ;		
07. B2: Si non le système informe l'administrateur de l'incohérence de l'enregistrement et le scénario reprend au point 05 du scénario nominal ;		
Règles de gestion et d'organisation :		

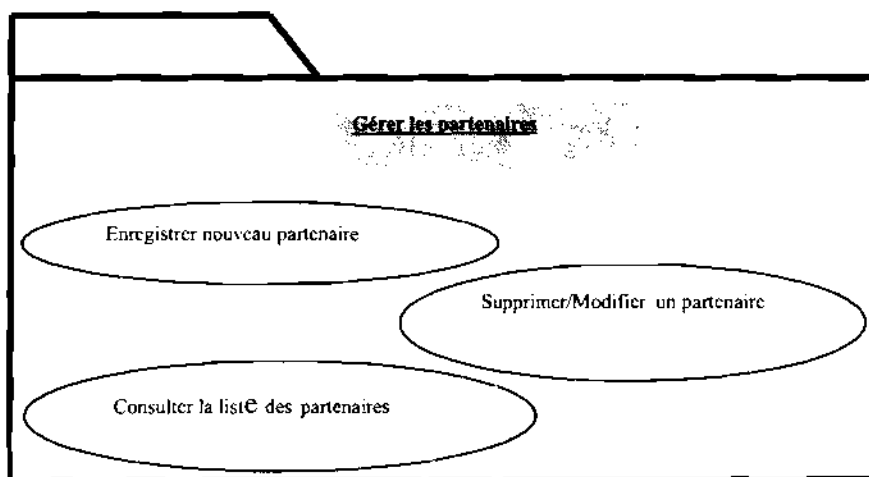
- Le cas d'utilisation « Supprimer/Modifier les informations d'une édition du prix « Santé et sécurité au travail » »

CU_3: modifier les informations d'une édition		Folio 2/3
Résumé: Ce CU permet de modifier les informations d'une édition du prix « Santé et sécurité au travail »		Pré condition: le système fonctionne correctement et qu'il y ait au moins un enregistrement d'édition.
Scénario nominal	Version: 1.0	Date de réalisation: 28/09/2007
L'administrateur du système		
Description du scénario nominal « Début » 01: inclusion du cas d'utilisation « S'authentifier » ; 02: le système affiche le menu ; 03: l'administrateur choisi l'option « Supprimer/Modifier les informations d'une édition du prix « Santé et sécurité au travail » » ; 04: le système affiche le formulaire de suppression/Modification des informations d'une édition du prix « Santé et sécurité au travail » ; 05: le système affiche la liste des éditions du prix « Santé et sécurité au travail » ; 06: l'administrateur du système choisi l'édition à supprimer/modifier ; (A) 07: le système demande une confirmation de l'opération (Supprimer/Modifier) ; (B) 08: le système effectue l'opération choisie ; 09: le système confirme la mise à jour ; « Fin »		
Extensions : 06. A : l'opération choisie est-elle valide? 06. A1 : Si oui on continue au point 07 du scénario nominal ; 06. A2 : Si non le système informe l'administrateur du système de l'invalidité de l'opération choisie et le scénario reprend au point 05 du scénario nominal ; 07. B : la mise à jour respect-elle les contraintes d'intégrité ? 07. B1 : Si oui on continue au point 08 du scénario nominal ; 07. B2: Si non le système informe l'administrateur du système de l'incohérence de l'enregistrement ; et le scénario reprend au point 05 du scénario nominal ;		
Règles de gestion et d'organisation :		

- Le cas d'utilisation « consulter les informations d'une édition du prix « Santé et sécurité au travail» »

CU 5: consulter les informations d'une édition		Folio 3/3
Résumé : Ce CU permet de consulter une édition du prix « Santé et sécurité au travail		Pré condition : le système fonctionne correctement et qu'il y ait au moins un enregistrement d'édition.
Scénario nominal	Version : 1.0	Date de réalisation : 28/09/2007
Tous les acteurs du système		
Description du scénario nominal		
« Début »		
01 : le système affiche le menu ;		
02 : l'utilisateur choisi l'option « consulter les informations d'une édition » ;		
03 : le système affiche le formulaire de Consultation des informations d'une édition ;		
04 : le système affiche la liste de toutes les éditions ;		
05 : l'utilisateur choisi ou lance une recherche de l'édition à consulter ; (A)		
06 : le système affiche les informations de l'édition choisie;		
« Fin »		
Extensions :		
05. A : la recherche est-elle fructueuse ?		
05. A1 : Si oui on continue au point 06 du scénario nominal.		
05. A2 : Si non le système informe l'administrateur de l'absence d'une édition ayant les paramètres de recherche saisie et le scénario reprend au point 05 du scénario nominal ;		
Règles de gestion et d'organisation :		

❖ Le cas d'utilisation « Gérer les partenaires »



- Le cas d'utilisation « Enregistrer un nouveau partenaire »

CU 6: Enregistrer un nouveau partenaire		Folio 1/3
Résumé : Ce CU permet 'ajouter un nouveau partenaire		Pré condition : le système fonctionne correctement et on suppose que les rapports de l'édition en question sont disponibles
Scénario nominal	Version : 1.0	Date de réalisation : 28/09/2007
L'administrateur du système		
Description du scénario nominal « Début » 01 : inclusion du cas d'utilisation « S'authentifier » ; 02 : le système affiche le menu ; 03 : l'administrateur choisi l'option « enregistrer un nouveau partenaire » ; 04 : le système affiche le formulaire d'enregistrement de nouveau partenaire ; 05 : l'administrateur saisit les informations ; 06 : le système vérifie les informations saisies ; (A) 07 : le système enregistre les informations dans la base de données ;(B) 08 : le système confirme l'enregistrement du partenaire ; « Fin »		
Extensions : 06. A : le formulaire est-il bien rempli ? 06. A1 : Si oui on continue au point 07 du scénario nominal. 06. A2 : Si non le système informe l'administrateur de l'incohérence des valeurs saisies et le scénario reprend au point 05 du scénario nominal ; 07. B : l'enregistrement respect-il les contraintes d'intégrité ? 07. B1 : Si oui on continue au point 08 du scénario nominal ; 07. B2 : Si non le système informe l'administrateur de l'incohérence de l'enregistrement et le scénario reprend au point 05 du scénario nominal ;		
Règles de gestion et d'organisation :		

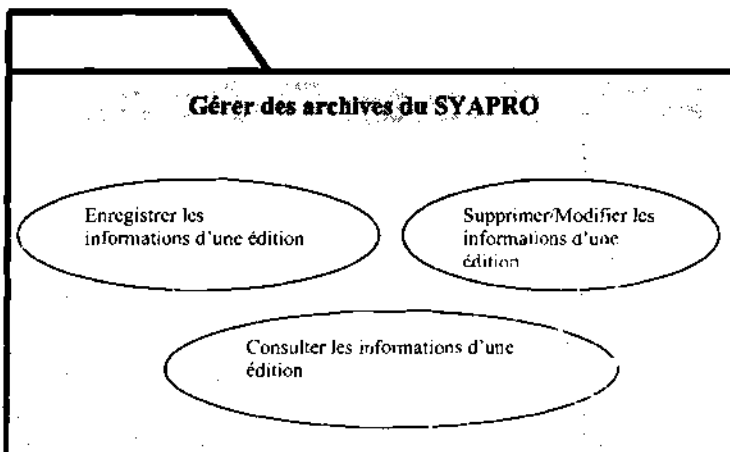
- Le cas d'utilisation « consulter la liste des partenaires »

CU_7: consulter la liste des partenaires		Folio 2/3
Résumé : Ce CU décrit la procédure de consultation des informations des partenaires		Pré condition : le système fonctionne correctement et on suppose que les rapports de l'édition en question sont disponibles
Scénario nominal	Version : 1.0	Date de réalisation : 28/09/2007
Tous les acteurs du système		
Description du scénario nominal « Début » 01 : le système affiche le menu ; 02 : l'utilisateur choisi l'option « consulter la liste des partenaires » ; 03 : le système affiche le formulaire de Consultation des informations d'un partenaire ; 04 : le système affiche la liste de tous les partenaires ; 05 : l'utilisateur choisi ou lance une recherche du partenaire à consulter ; (A) 06 : le système affiche les informations de l'édition choisie ; « Fin »		
Extensions : 05. A : la recherche est-elle fructueuse ? 05. A1 : Si oui on continue au point 05 du scénario nominal. 05. A2 : Si non le système informe l'utilisateur de l'absence d'une édition ayant les paramètres de recherche saisie et le scénario reprend au point 04 du scénario nominal ;		
Règles de gestion et d'organisation :		

- Le cas d'utilisation « Supprimer/Modifier les informations d'un partenaire »

CU_8: Supprimer/Modifier les informations d'un partenaire		Folio 3/3
Résumé: Ce CU décrit la procédure de Suppression/Modification des informations d'un partenaire		Pré condition: le système fonctionne correctement
Scénario nominal	Version: 1.0	Date de réalisation: 28/09/2007
Administrateur du système		
Description du scénario nominal		
« Début »		
01: inclusion du cas d'utilisation « S'authentifier »;		
02: le système affiche le menu;		
03: l'administrateur choisi l'option « Supprimer/Modifier les informations d'un partenaire »;		
04: le système affiche le formulaire de suppression/Modification des informations d'un partenaire;		
05: le système affiche la liste des partenaires;		
06: L'administrateur du système choisi le partenaire à supprimer/modifier; (A)		
07: le système demande une confirmation de l'opération (Supprimer/Modifier); (B)		
08: le système effectue l'opération choisie;		
09: le système confirme la mise à jour;		
« Fin »		
Extensions:		
06. A: l'opération choisie est-elle valide?		
06. A1: Si oui on continue au point 07 du scénario nominal.		
06. A2: Si non le système informe l'administrateur du système de l'invalidité de l'opération choisie et le scénario reprend au point 05 du scénario nominal;		
07. B: la mise à jour respect-elle les contraintes d'intégrité ?		
07. B1: Si oui on continue au point 08 du scénario nominal;		
07. B2: Si non le système informe l'administrateur du système de l'incohérence de l'enregistrement; et le scénario reprend au point 05 du scénario nominal;		
Règles de gestion et d'organisation:		

❖ Cas d'utilisation « Gérer des archives du SYAPRO »



- Le cas d'utilisation « Enregistrer information d'une édition du SYAPRO »

CU 9: Enregistrer information d'une édition du SYAPRO		Folio 1/3
Résumé : Ce CU permet d'enregistrer les informations d'une édition du SYAPRO		Pré condition : le système fonctionne correctement et on suppose que les rapports de l'édition en question sont disponibles
Scénario nominal	Version : 1.0	Date de réalisation : 28/09/2007
L'administrateur du système		
Description du scénario nominal		
« Début »		
01 : inclusion du cas d'utilisation « S'authentifier » ;		
02 : le système affiche le menu ;		
03 : l'administrateur choisi l'option « Enregistrer les informations d'une édition du SYAPRO » ;		
04 : le système affiche le formulaire d'enregistrement		
05 : l'administrateur saisit les informations sur la nouvelle édition ;		
06 : le système vérifie les informations saisies ; (A)		
07 : le système enregistre les informations dans la base de données;(B)		
08 : le système confirme l'enregistrement de l'édition;		
« Fin »		
Extensions :		
06. A : le formulaire est-il bien rempli?		
06. A1 : Si oui on continue au point 07 du scénario nominal.		
06. A2 : Si non le système informe l'administrateur de l'incohérence des valeurs saisies et le scénario reprend au point 05 du scénario nominal ;		
07. B : l'enregistrement respect-il les contraintes d'intégrité ?		
07. B1 : Si oui on continue au point 08 du scénario nominal ;		
07. B2: Si non le système informe l'administrateur de l'incohérence de l'enregistrement et le scénario reprend au point 05 du scénario nominal ;		
Règles de gestion et d'organisation :		
Seul l'administrateur du système peut enregistrer les informations d'une édition ;		

- Le cas d'utilisation « Supprimer/Modifier les informations d'une édition du SYAPRO »

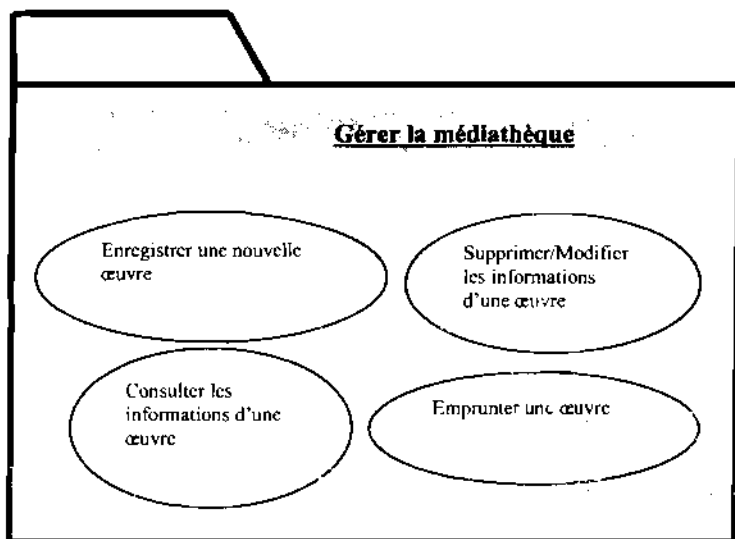
CU 10: modifier les informations d'une édition du SYAPRO	Folio 2/3
Résumé : Ce CU permet de modifier les informations d'une édition du SYAPRO	Pré condition : le système fonctionne correctement et que des éditions ont déjà été enregistrées.
Scénario nominal	Version : 1.0
Date de réalisation : 28/09/2007	
L'administrateur du système	
Description du scénario nominal	
« Début »	
01 : inclusion du cas d'utilisation « S'authentifier » ;	
02 : le système affiche le menu ;	
03 : l'administrateur choisi l'option « Supprimer/Modifier les informations d'une édition » ;	
04 : le système affiche le formulaire de suppression/Modification des informations d'une édition ;	
05 : le système affiche la liste des éditions ;	
06 : Le chargé de la gestion choisi l'édition à supprimer/modifier ; (A)	
07 : le système demande une confirmation de l'opération (Supprimer/Modifier) ; (B)	
08 : le système effectue l'opération choisie ;	
09 : le système confirme la mise à jour ;	
« Fin »	
Extensions :	
06. A : l'opération choisie est-elle valide ?	
06. A1 : Si oui on continue au point 07 du scénario nominal.	
06. A2 : Si non le système informe le chargé de la gestion de l'invalidité de l'opération choisie et le scénario reprend au point 05 du scénario nominal ;	
07. B : la mise à jour respect-elle les contraintes d'intégrité ?	
07. B1 : Si oui on continue au point 08 du scénario nominal ;	
07. B2: Si non le système informe Le chargé de la gestion de l'incohérence de l'enregistrement ; et le scénario reprend au point 05 du scénario nominal ;	
Règles de gestion et d'organisation :	

- Le cas d'utilisation « consulter les informations d'une édition du SYAPRO »

CU_12: consulter les informations d'une édition du SYAPRO		Folio 3/3
Résumé : Ce CU décrit la procédure de consultation des informations d'une édition du SYAPRO		Pré condition : le système fonctionne correctement et que des éditions ont déjà été enregistrées.
Scénario nominal	Version : 1.0	Date de réalisation : 28/09/2007
Tous les acteurs du système		
Description du scénario nominal « Début » 01 : le système affiche le menu principale; 02 : l'utilisateur choisi l'option « consulter les informations d'une édition » ; 03 : le système affiche le formulaire de Consultation des informations d'une édition ; 04 : le système affiche la liste de toutes les éditions ; 05 : l'utilisateur choisi ou lance une recherche de l'édition à consulter ; (A) 06 : le système affiche les informations de l'édition choisie; « Fin »		
Extensions : 05. A : la recherche est-elle fructueuse ? 05. A1 : Si oui on continue au point 06 du scenario nominal. 05. A2 : Si non le système informe l'administrateur de l'absence d'une édition ayant les paramètres de recherche saisie et le scénario reprend au point 04 du scénario nominal ;		
Règles de gestion et d'organisation :		

❖ Cas d'utilisation « Gérer la médiathèque »

Le cas d'utilisation gérer la médiathèque est un paquetage regroupant les cas d'utilisation «Enregistrer une nouvelle œuvre », «Emprunter une œuvre », « Consulter les informations d'une œuvre », « Supprimer/Modifier les informations d'une œuvre ».



- Cas d'utilisation « Enregistrer une nouvelle œuvre »

CU 13: Enregistrer une nouvelle œuvre		Folio 1/4
Résumé : Ce CU permet d'enregistrer des œuvres dans la médiathèque		Pré condition : le système fonctionne correctement et les informations descriptives de l'œuvre sont disponibles.
Scénario nominal	Version : 1.0	Date de réalisation : 28/09/2007
Chargé de gestion de la médiathèque		
Description du scénario nominal « Début » 01 : inclusion du cas d'utilisation « S'authentifier » ; 02 : le système affiche le menu ; 03 : le Chargé de gestion de la médiathèque choisi l'option « Enregistrer une nouvelle œuvre » ; 04 : le système affiche le formulaire d'enregistrement 05 : Chargé de gestion de la médiathèque saisit les informations sur la nouvelle œuvre ; 06 : le système vérifie les informations saisies ; (A) 07 : le système enregistre les informations dans la base de données;(B) 08 : le système confirme l'enregistrement des informations saisies; « Fin »		
Extensions : 06. A : le formulaire est-il bien rempli? 06. A1 : Si oui on continue au point 07 du scénario nominal. 06. A2 : Si non le système informe l'administrateur de l'incohérence des valeurs saisies et le scénario reprend au point 05 du scénario nominal ; 07. B : l'enregistrement respect-il les contraintes d'intégrité ? 07. B1 : Si oui on continue au point 08 du scénario nominal ; 07. B2 : Si non le système informe le Chargé de gestion de la médiathèque de l'incohérence de l'enregistrement et le scénario reprend au point 05 du scénario nominal ;		
Règles de gestion et d'organisation :		

- Cas d'utilisation «Supprimer/Modifier les informations d'une œuvre»

CU_15: Supprimer/Modifier une œuvre		Folio 2/4
Résumé : Ce CU décrit la procédure de suppression ou de modification des informations d'une œuvre donnée.		Pré condition : le système fonctionne correctement et au moins une œuvre a déjà été enregistrée
Scénario nominal	Version : 1.0	Date de réalisation : 28/09/2007
L'administrateur du système		
Description du scénario nominal « Début » 01 : inclusion du cas d'utilisation « S'authentifier » ; 02 : le système affiche le menu ; 03 : l'administrateur choisi l'option « Supprimer/Modifier une œuvre » ; 04 : le système affiche le formulaire de suppression/Modification d'une œuvre ; 05 : le système affiche la liste des œuvres ; 06 : Le chargé de la gestion choisi l'œuvre à supprimer/modifier ; (A) 07 : le système demande une confirmation de l'opération (Supprimer/Modifier) ; (B) 08 : le système effectue l'opération choisie ; 09 : le système confirme la mise à jour ; « Fin »		
Extensions : 06. A : l'opération choisie est-elle valide? 06. A1 : Si oui on continue au point 07 du scénario nominal. 06. A2 : Si non le système informe le chargé de la gestion de l'invalidité de l'opération choisie et le scénario reprend au point 05 du scénario nominal ; 07. B : la mise à jour respect-elle les contraintes d'intégrité ? 07. B1 : Si oui on continue au point 08 du scénario nominal ; 07. B2 : Si non le système informe Le chargé de la gestion de l'incohérence de l'enregistrement ; et le scénario reprend au point 05 du scénario nominal ;		
Règles de gestion et d'organisation : Les informations d'une œuvre déjà empruntée ne peuvent faire l'objet d'une mise à jour		

- Le cas d'utilisation « consulter les informations sur une œuvre »

CU_16: consulter les informations sur une œuvre		Folio 3/4
Résumé : Ce CU permet de faire la consultation des informations sur une œuvre		Pré condition : le système fonctionne correctement et au moins une œuvre a déjà été enregistrée
Scénario nominal	Version : 1.0	Date de réalisation : 28/09/2007
Tous les acteurs du système		
Description du scénario nominal « Début » 01 : le système affiche le menu ; 02 : l'utilisateur choisi l'option « consulter la liste des œuvres » ; 03 : le système affiche le formulaire de Consultation des œuvres ; 04 : le système affiche la liste de tous les œuvres ; 05 : l'utilisateur choisi ou lance une recherche de l'œuvre à consulter ; (A) 06 : le système affiche les informations de l'œuvre choisie ; « Fin »		
Extensions : 05. A : la recherche est-elle fructueuse ? 05. A1 : Si oui on continue au point 06 du scénario nominal. 05. A2 : Si non le système informe l'utilisateur de l'absence d'une œuvre ayant les paramètres de recherche saisie et le scénario reprend au point 04 du scénario nominal ;		
Règles de gestion et d'organisation :		

- Le cas d'utilisation « Emprunter une œuvre »

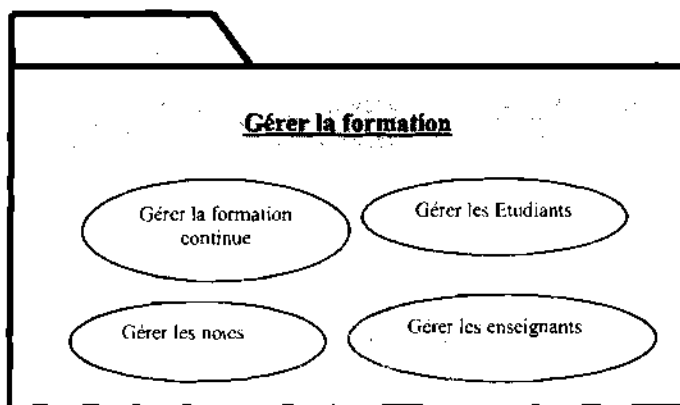
CU 17: Emprunter une œuvre		Folio 4/4
Résumé : Ce CU permet de gérer les emprunts des œuvres de la médiathèque		Pré condition : le système fonctionne correctement et l'œuvre à emprunter doit être disponible
Scénario nominal	Version : 1.0	Date de réalisation : 28/09/2007
Le chargé de gestion de la médiathèque		
Description du scénario nominal		
« Début »		
01 : le client fournit ses propres références et celles des œuvres à emprunter ;		
02 : inclusion du cas d'utilisation « S'authentifier » ;		
03 : le système affiche le menu ;		
04 : Le chargé de gestion de la médiathèque choisit l'option « gestion des emprunts d'œuvre » ;		
05 : le système affiche le formulaire d'emprunt des œuvres ;		
06 : Le chargé de gestion de la médiathèque entre les références du client ;		
07 : le système vérifie les références du client ;(A)		
08 : Le système affiche les informations sur le client ;		
09 : Le système vérifie les emprunts en cours du client;(B)		
10 : Pour chaque œuvre le chargé de gestion de la médiathèque introduit le code ;		
11 : Le système vérifie le code de l'œuvre ;(C)		
12 : Le système vérifie la disponibilité de l'œuvre ;(D)		
13 : Le chargé de gestion de la médiathèque valide les saisies ;		
14 : Le système enregistre l'emprunt de l'œuvre ;		
15 : Le système donne la main pour une nouvelle saisie ;		
16 : Le chargé de gestion de la médiathèque remet les œuvres au client.		
« Fin »		
Extensions :		
07. A : le client est-il déjà répertorié ?		
07. A1 : Si oui on continue au point 08 du scénario nominal.		
07. A2 : Si non le système informe le chargé de gestion que le client n'est pas répertorié et le scénario continue au point 10 du scénario nominal ;		
09. B : le client a-t-il des emprunts en cours ?		
09. B1 : si oui le système informe le chargé de gestion de la médiathèque que le client a des emprunts en cours ;		
09. B1 : sinon on continue au point 10 du scénario nominal ;		
11. C : Le code est-il valide ?		
11. C1 : si oui on continue au point 12 du scénario nominal ;		
11. C2 : sinon le système informe le chargé de gestion de la médiathèque de l'invalidité du code saisi et le scénario reprend au point 10 du scénario nominal ;		
12. D : L'œuvre est-elle disponible ?		
12. D1 : si oui on continue au point 13 du scénario nominal ;		
12. D2 : sinon le système informe le chargé de gestion de la médiathèque du non disponibilité de l'œuvre et le scénario reprend au point 10 du scénario nominal		
Règles de gestion et d'organisation :		

- Cas d'utilisation « Inscrire en ligne »

CU 18 : Inscrire en ligne		Folio 1/1
Résumé : Ce CU décrit la procédure d'inscription en ligne au CPFACE		Pré condition : le système fonctionne correctement et qu'un appel à candidature soit lancé
Scénario nominal	Version : 1.0	Date de réalisation : 28/09/2007
Les structures des pays membre de l'IAPRP (S.IAPRP)		
Description du scénario nominal « Début » 01 : le système affiche le menu ; 02 : l'utilisateur choisi l'option « Inscrire en ligne » ; 03 : le système affiche le formulaire d'inscription en ligne; 04 : l'intéressé remplit le formulaire d'inscription ; 05 : le système vérifie les informations saisies : (A) 06 : le système enregistre les informations dans la base de données; 07 : le système confirme l'envoi des dossiers d'inscription; « Fin »		
Extensions : 05. A : le formulaire est-il bien rempli? 05. A1 : Si oui on continue au point 06 du scénario nominal. 05. A2 : Si non le système informe l'administrateur de l'incohérence des valeurs saisies et le scénario reprend au point 04 du scénario nominal ;		
Règles de gestion et d'organisation :		

❖ Cas d'utilisation « Gérer la formation »

Ce cas d'utilisation représenté sous forme de paquetage, comprend la gestion des notes, la gestion de la formation continue, la gestion des étudiants et la gestion des enseignants.



- Cas d'utilisation « gérer la formation continue »

CU 19: Gérer la formation continue		Folio 1/4
Résumé : Ce CU décrit le processus de gestion d'une formation continue organisée par CPFACE		Pré condition : le système fonctionne correctement et
Scénario nominal	Version : 1.0	Date de réalisation : 28/09/2007
Le R.CPFACE		
Description du scénario nominal « Début » 01 : inclusion du cas d'utilisation « S'authentifier » ; 02 : le système affiche le menu ; 03 : l'utilisateur choisi l'option « Gérer formation continue » ; 04 : le système affiche le formulaire de gestion de la formations continue; 05 : le coordonnateur choisit une opération ; 06 : le système vérifie la validité de l'opération choisie ; (A) 07 : le système effectue une mise à jour des informations dans la base de données;(B) 08 : le système confirme la mise à jour des informations; « Fin »		
Extensions : 06. A : L'opération est-elle valide ? 06. A1 : Si oui on continue au point 07 du scénario nominal. 06. A2 : Si non le système informe le coordonnateur de l'invalidité de l'opération choisie et le scénario reprend au point 05 du scénario nominal ; 07. B : la mise à jour respect-elle les contraintes d'intégrité ? 07. B1 : Si oui on continue au point 08 du scénario nominal ; 07. B2: Si non le système informe Le coordonnateur de l'incohérence de la mise à jour ; et le scénario reprend au point 05 du scénario nominal.		
Règles de gestion et d'organisation :		

- Le cas d'utilisation « Emprunter une œuvre »

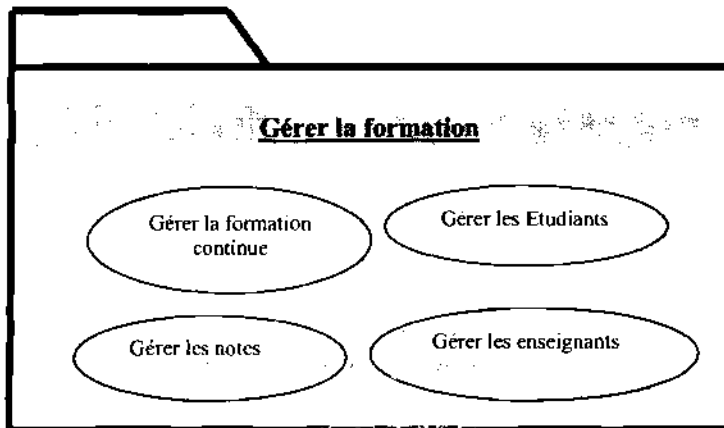
CU 17: Emprunter une œuvre		Folio 4/4
Résumé : Ce CU permet de gérer les emprunts des œuvres de la médiathèque		Pré condition : le système fonctionne correctement et l'œuvre à emprunter doit être disponible
Scénario nominal	Version : 1.0	Date de réalisation : 28/09/2007
Le chargé de gestion de la médiathèque		
Description du scénario nominal		
« Début »		
01 : le client fournit ses propres références et celles des œuvres à emprunter ;		
02 : inclusion du cas d'utilisation « S'authentifier » ;		
03 : le système affiche le menu ;		
04 : Le chargé de gestion de la médiathèque choisit l'option « gestion des emprunts d'œuvre » ;		
05 : le système affiche le formulaire d'emprunt des œuvres ;		
06 : Le chargé de gestion de la médiathèque entre les références du client ;		
07 : le système vérifie les références du client ;(A)		
08 : Le système affiche les informations sur le client ;		
09 : Le système vérifie les emprunts en cours du client;(B)		
10 : Pour chaque œuvre le chargé de gestion de la médiathèque introduit le code ;		
11 : Le système vérifie le code de l'œuvre ;(C)		
12 : Le système vérifie la disponibilité de l'œuvre ;(D)		
13 : Le chargé de gestion de la médiathèque valide les saisies ;		
14 : Le système enregistre l'emprunt de l'œuvre ;		
15 : Le système donne la main pour une nouvelle saisie ;		
16 : Le chargé de gestion de la médiathèque remet les œuvres au client.		
« Fin »		
Extensions :		
07. A : le client est-il déjà répertorié ?		
07. A1 : Si oui on continue au point 08 du scénario nominal.		
07. A2 : Si non le système informe au chargé de gestion que le client n'est pas répertorié et le scénario continue au point 10 du scénario nominal ;		
09. B : le client a-t-il des emprunts en cours ?		
09. B1 : si oui le système informe le chargé de gestion de la médiathèque que le client a des emprunts en cours ;		
09. B1 : sinon on continue au point 10 du scénario nominal ;		
11. C : Le code est-il valide ?		
11. C1 : si oui on continue au point 12 du scénario nominal ;		
11. C2 : sinon le système informe le chargé de gestion de la médiathèque de l'invalidité du code saisi et le scénario reprend au point 10 du scénario nominal ;		
12. D : L'œuvre est-elle disponible ?		
12. D1 : si oui on continue au point 13 du scénario nominal ;		
12. D2 : sinon le système informe le chargé de gestion de la médiathèque du non disponibilité de l'œuvre et le scénario reprend au point 10 du scénario nominal		
Règles de gestion et d'organisation :		

- Cas d'utilisation « Inscrire en ligne »

CU 18 : Inscrire en ligne		Folio 1/1
Résumé : Ce CU décrit la procédure d'inscription en ligne au CPFACE		Pré condition : le système fonctionne correctement et qu'un appel à candidature soit lancé
Scénario nominal	Version : 1.0	Date de réalisation : 28/09/2007
Les structures des pays membre de l'IAPRP (S.IAPRP)		
Description du scénario nominal		
« Début »		
01 : le système affiche le menu ;		
02 : l'utilisateur choisi l'option « Inscrire en ligne » ;		
03 : le système affiche le formulaire d'inscription en ligne;		
04 : l'intéressé remplit le formulaire d'inscription ;		
05 : le système vérifie les informations saisies ; (A)		
06 : le système enregistre les informations dans la base de données;		
07 : le système confirme l'envoi des dossiers d'inscription;		
« Fin »		
Extensions :		
05. A : le formulaire est-il bien rempli?		
05. A1 : Si oui on continue au point 06 du scénario nominal.		
05. A2 : Si non le système informe l'administrateur de l'incohérence des valeurs saisies et le scénario reprend au point 04 du scénario nominal ;		
Règles de gestion et d'organisation :		

❖ Cas d'utilisation « Gérer la formation »

Ce cas d'utilisation représenté sous forme de paquetage, comprend la gestion des notes, la gestion de la formation continue, la gestion des étudiants et la gestion des enseignants.



- Cas d'utilisation « gérer des étudiants »

CU_22: Gérer les étudiants		Folio 4/4
Résumé : Ce CU décrit le processus de gestion des étudiants du CPFACE		Pré condition : le système fonctionne correctement et les dossiers des étudiants sont disponibles
Scénario nominal	Version : 1.0	Date de réalisation : 28/09/2007
Le coordonnateur		
Description du scénario nominal « Début » 01 : inclusion du cas d'utilisation « S'authentifier » ; 02 : le système affiche le menu ; 03 : l'utilisateur choisi l'option « Gérer les étudiants » ; 04 : le système affiche le formulaire de gestion des étudiants; 05 : le coordonnateur choisit une opération ; 06 : le système vérifie la validité de l'opération choisie ; (A) 07 : le système effectue une mise à jour des informations dans la base de données;(B) 08 : le système confirme la mise à jour des informations; « Fin »		
Extensions : 06. A : L'opération est-elle valide ? 06. A1 : Si oui on continue au point 07 du scenario nominal. 06. A2 : Si non le système informe le coordonnateur de l'invalidité de l'opération choisie et le scénario reprend au point 05 du scénario nominal. 07. B : la mise à jour respect-elle les contraintes d'intégrité ? 07. B1 : Si oui on continue au point 08 du scénario nominal ; 07. B2: Si non le système informe Le coordonnateur de l'incohérence de la mise à jour et le scénario reprend au point 05 du scénario nominal.		
Règles de gestion et d'organisation :		

❖ Cas d'utilisation « Envoyer un message »

CU_23: envoyer message		Folio 1/1
Résumé : Ce CU décrit le processus d'envoi des messages		Pré condition : le système fonctionne correctement
Scénario nominal	Version : 1.0	Date de réalisation : 28/09/2007
Tous les acteurs du système		
Description du scénario nominal « Début » 01 : le système affiche le menu ; 02 : l'utilisateur choisi l'option d'« Envoyer message » ; 03 : le système affiche le formulaire d'envoi de message; 04 : l'utilisateur remplit le formulaire d'envoi de message; 05 : le système vérifie les informations saisies ; (A) 06 : le l'utilisateur valide l'envoi du message 07 : le système enregistre les informations liés au message dans la base de données; 08 : le système confirme l'envoi du message aux destinataires; « Fin »		
Extensions : 05. A : le formulaire d'envoi est-il bien rempli? 05. A1 : Si oui on continue au point 06 du scenario nominal. 05. A2 : Si non le système informe l'utilisateur de l'incohérence des valeurs saisies et le scénario reprend au point 04 du scénario nominal ;		
Règles de gestion et d'organisation :		

4.1.1 .2- Diagramme de séquence

❖ cas d'utilisation «S'authentifier »

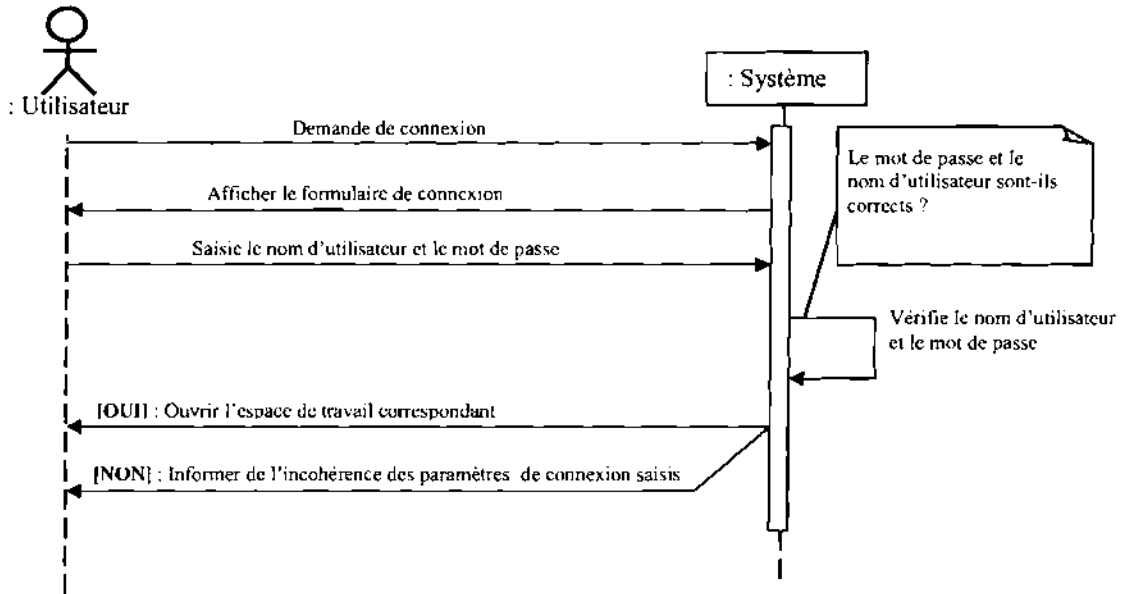


Diagramme4.2 Diagramme de séquence décrivant la procédure d'authentification

❖ Diagrammes de séquence du CU « Gérer les archives du prix «SST »

- CU Enregistrer les informations d'une édition

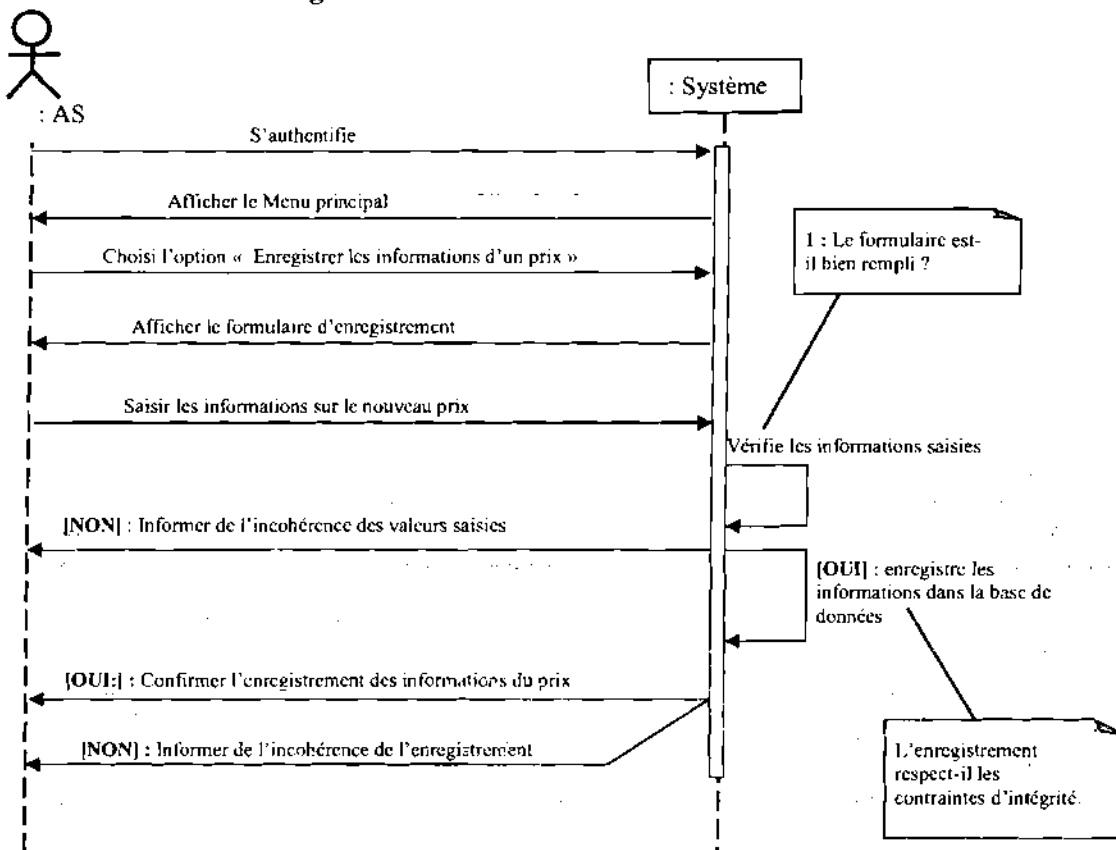


Diagramme4.3 Diagramme de séquence décrivant la procédure de Gestion des archives sur le prix « SST »

- CU Supprimer/Modifier les informations d'une édition

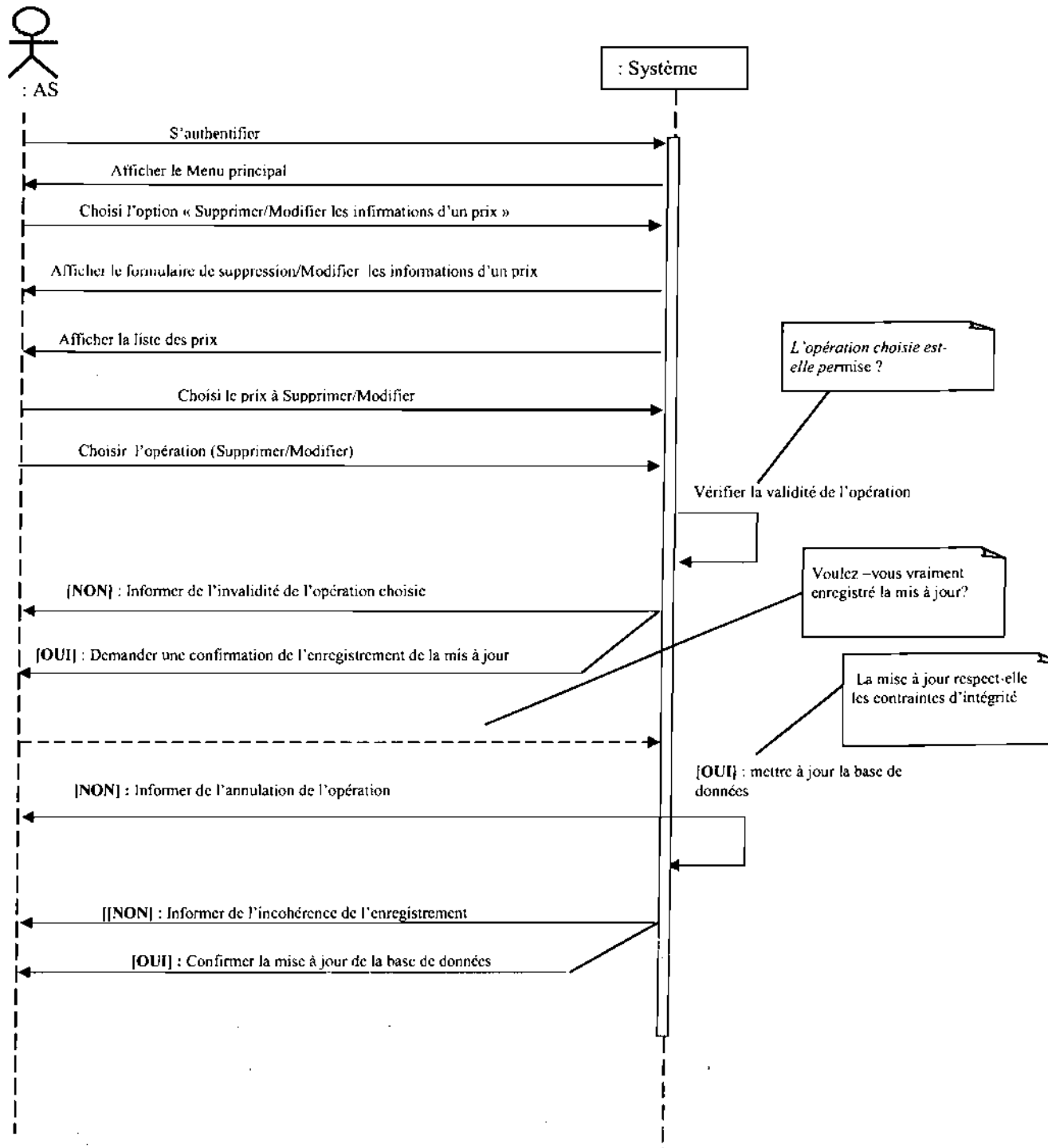


Diagramme4.4 Diagramme de séquence décrivant la procédure de suppression ou de modification d'information sur le prix « SST »

– CU Consulter des informations d'une édition

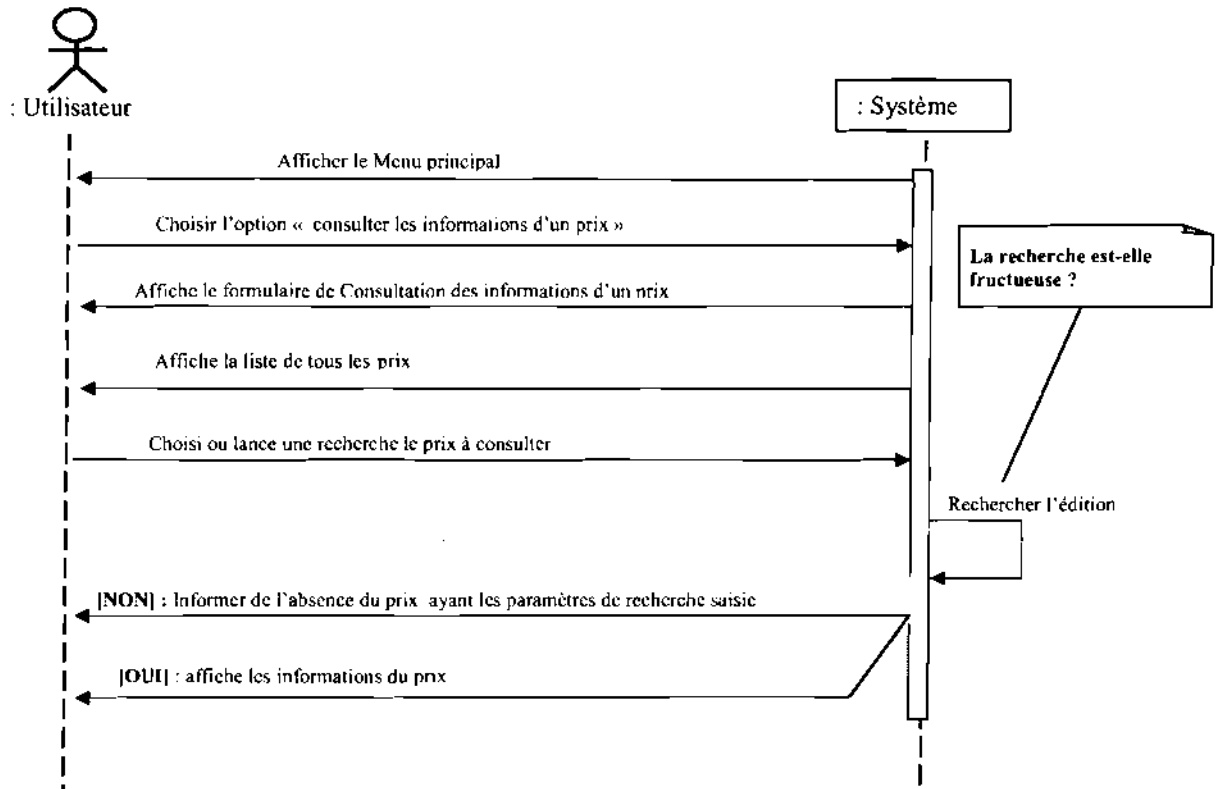


Diagramme4.5 Diagramme de séquence décrivant la procédure d'affichage d'informations sur le prix « SST »

❖ Le cas d'utilisation « Gérer les partenaires »
-CU Enregistrer un partenaire

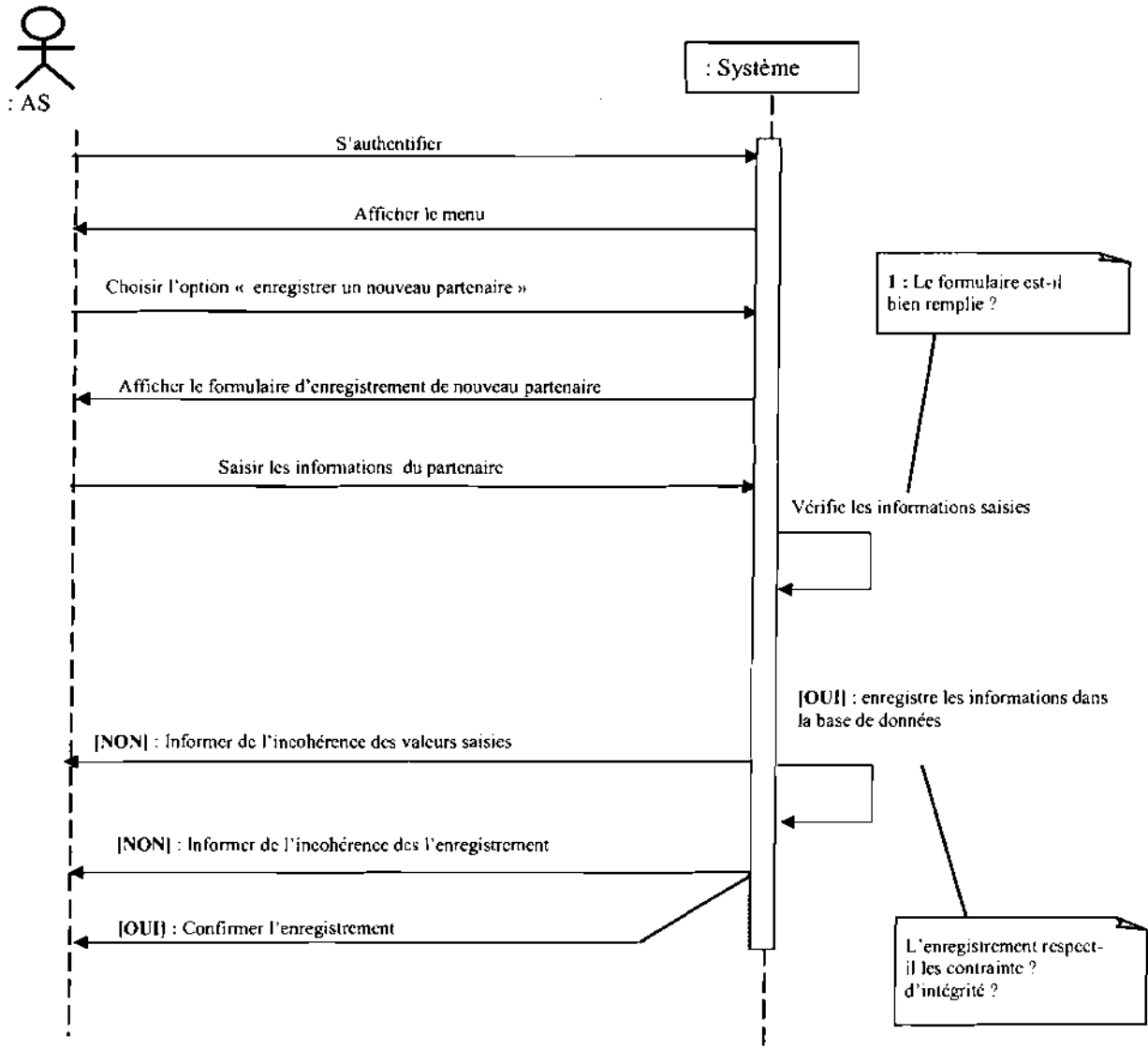


Diagramme4.6 Diagramme de séquence décrivant la procédure d'enregistrement d'un nouveau partenaire

-CU Supprimer/Modifier des informations d'un partenaire

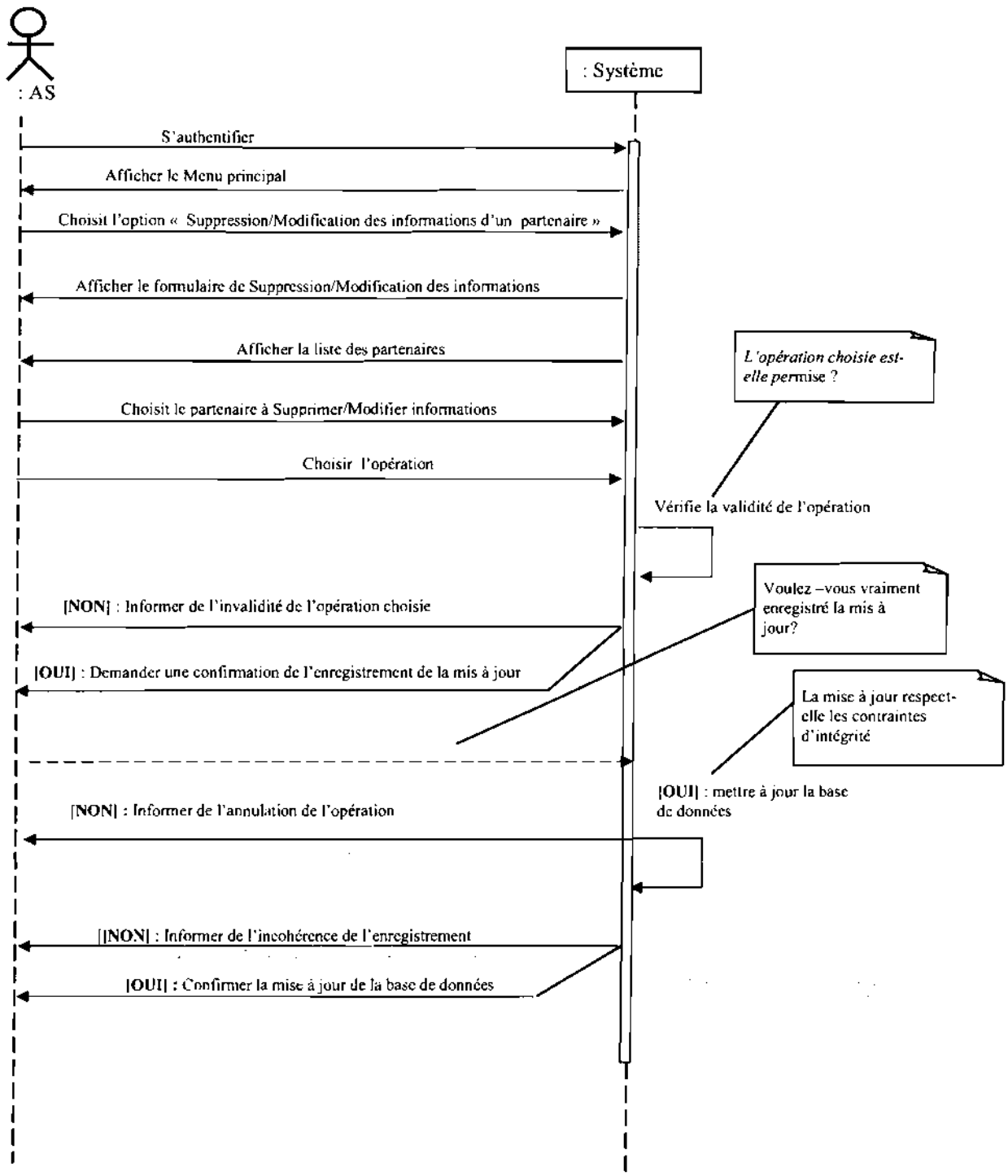


Diagramme4.7 Diagramme de séquence décrivant la procédure de suppression/modification d'information sur un partenaire

-CU Consulter des informations d'un partenaire

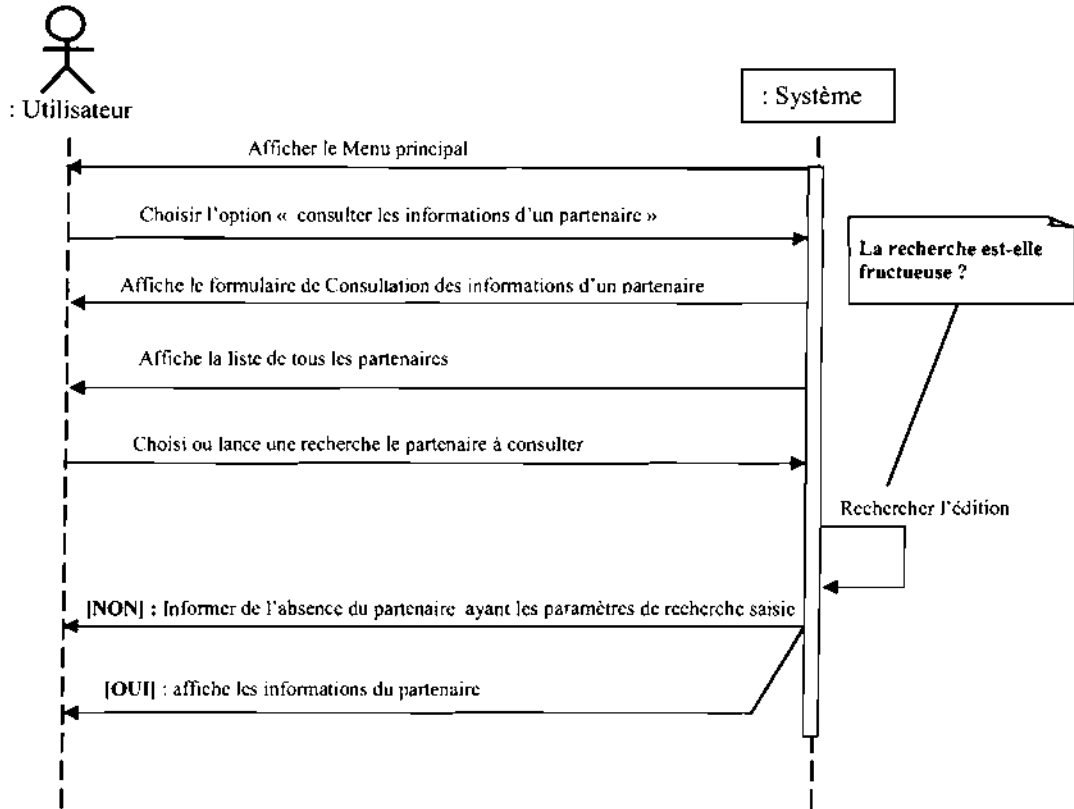


Diagramme4.8 Diagramme de séquence décrivant la procédure de consultation d'informations sur un partenaire

❖ **Le cas d'utilisation « Gérer les archives d'une édition du SYAPRO »**
 -CU Enregistrer les informations d'une édition

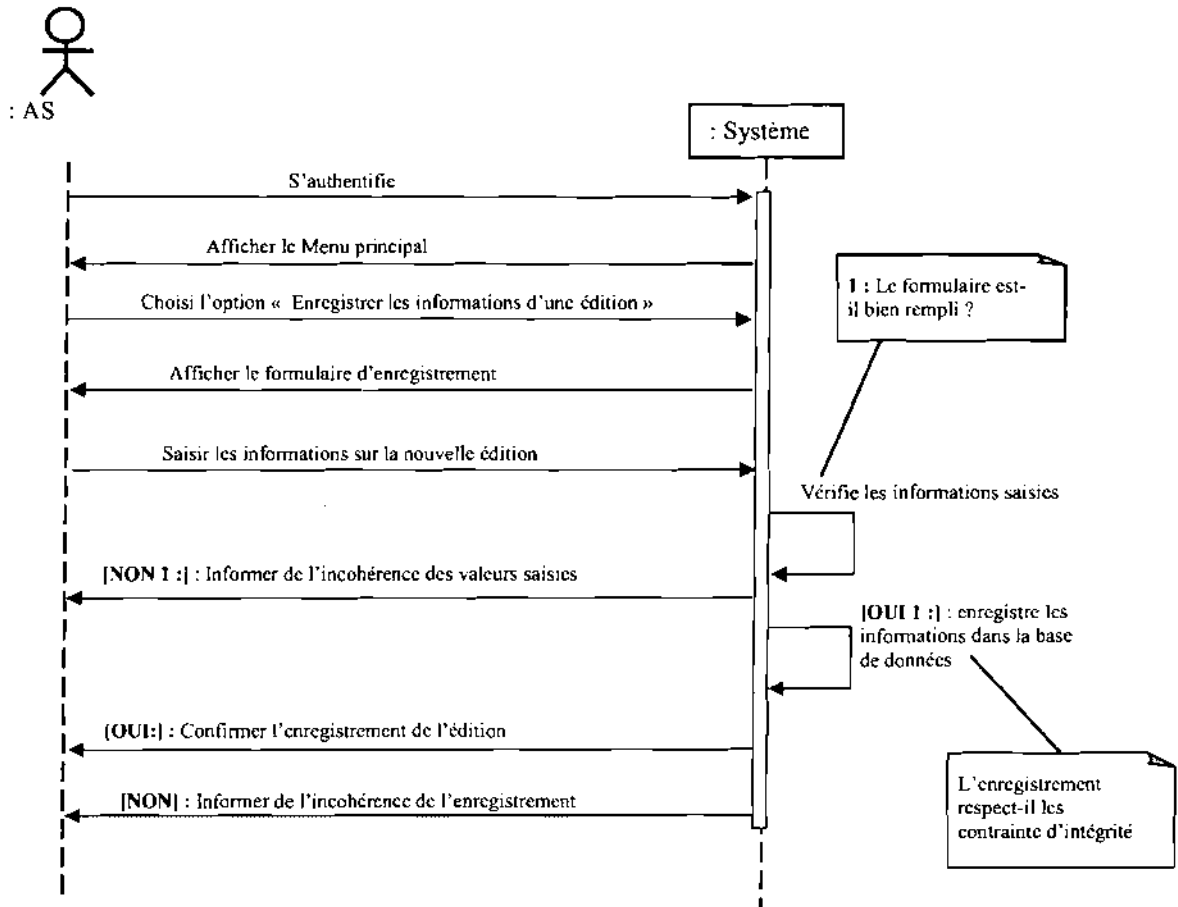


Diagramme4.9 Diagramme de séquence décrivant la procédure d'enregistrement d'information sur une édition du SYAPRO

- CU Modifier/ Supprimer les informations d'une édition

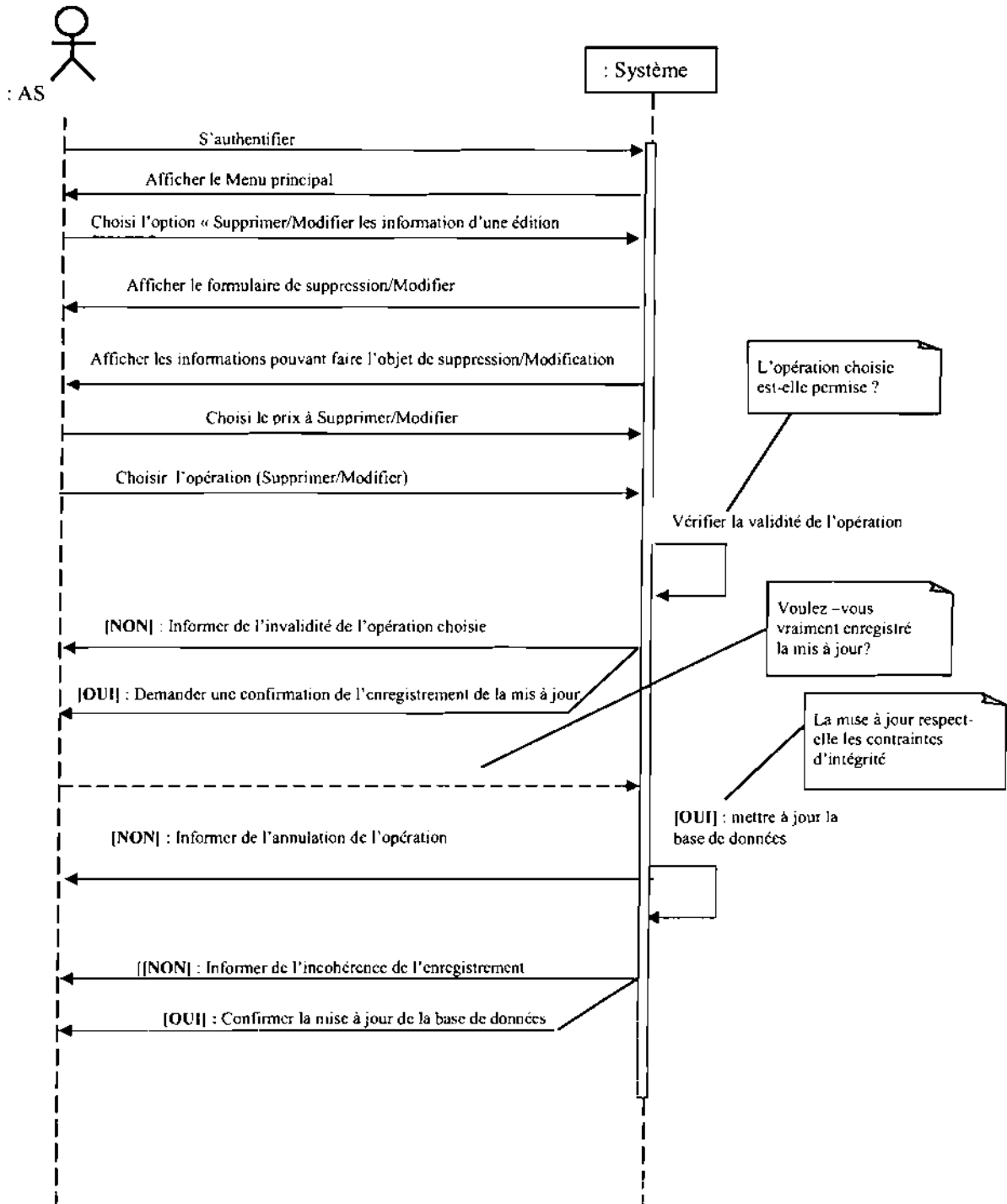


Diagramme4.10 Diagramme de séquence décrivant la procédure de Modification Suppression des Informations d'une édition

- CU Consulter les informations d'une édition du SYAPRO

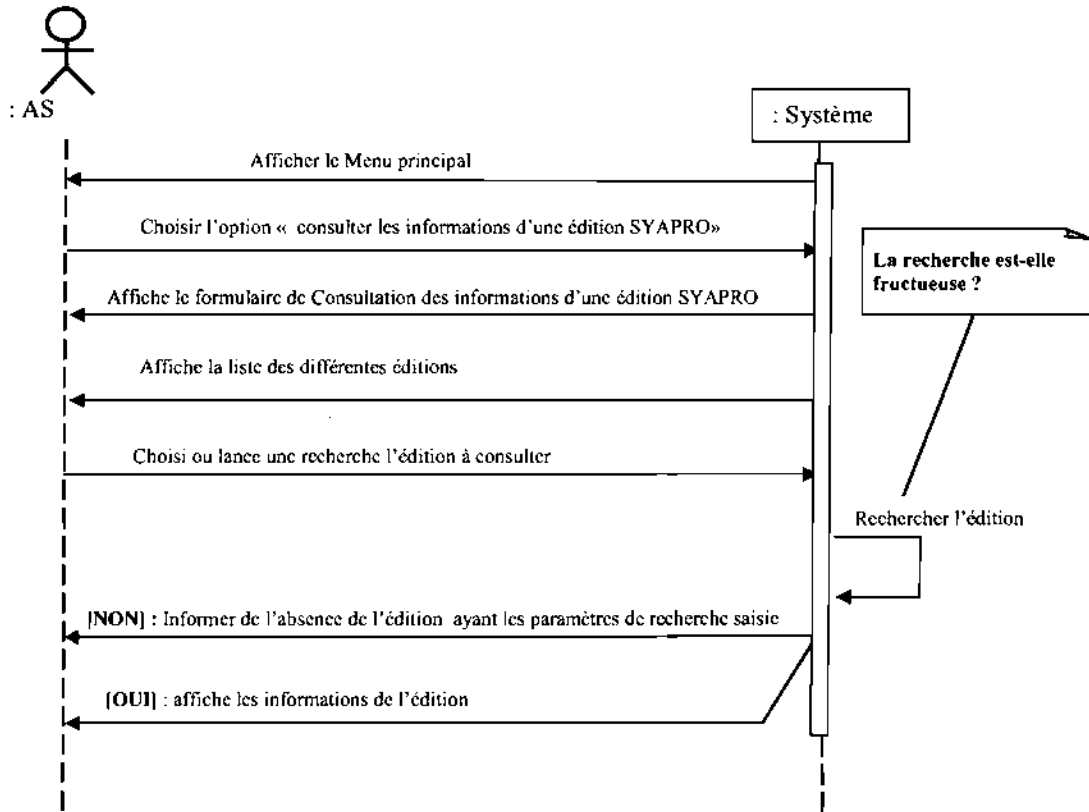


Diagramme4.11 Diagramme de séquence décrivant la procédure de consultation d'un formation sur une édition du SYAPRO

❖ Cas d'utilisation «S'inscrire en ligne »

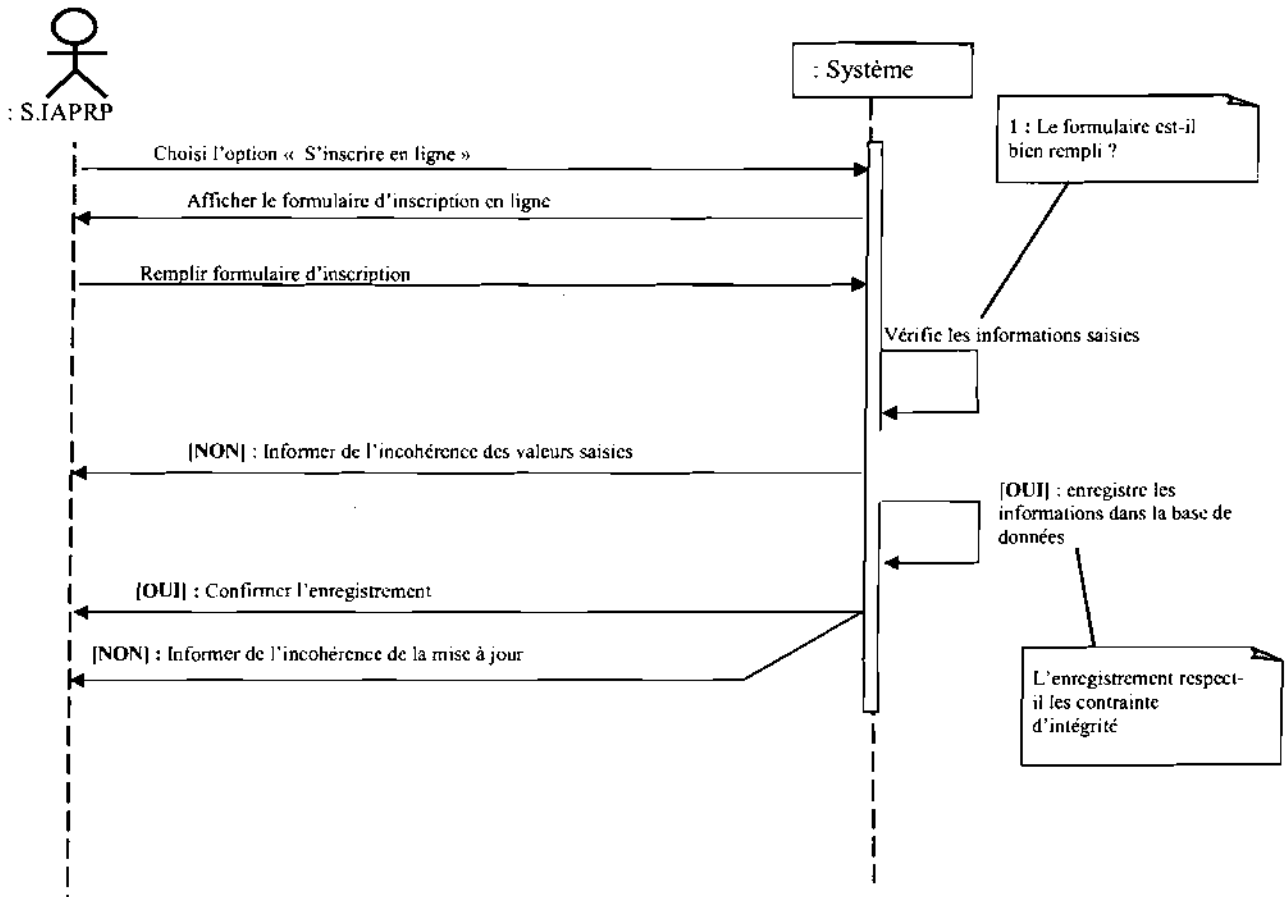


Diagramme4.12 Diagramme de séquence décrivant la procédure d'inscription en ligne

❖ Cas d'utilisation « Gérer la médiathèque »

– CU Enregistrer une œuvre

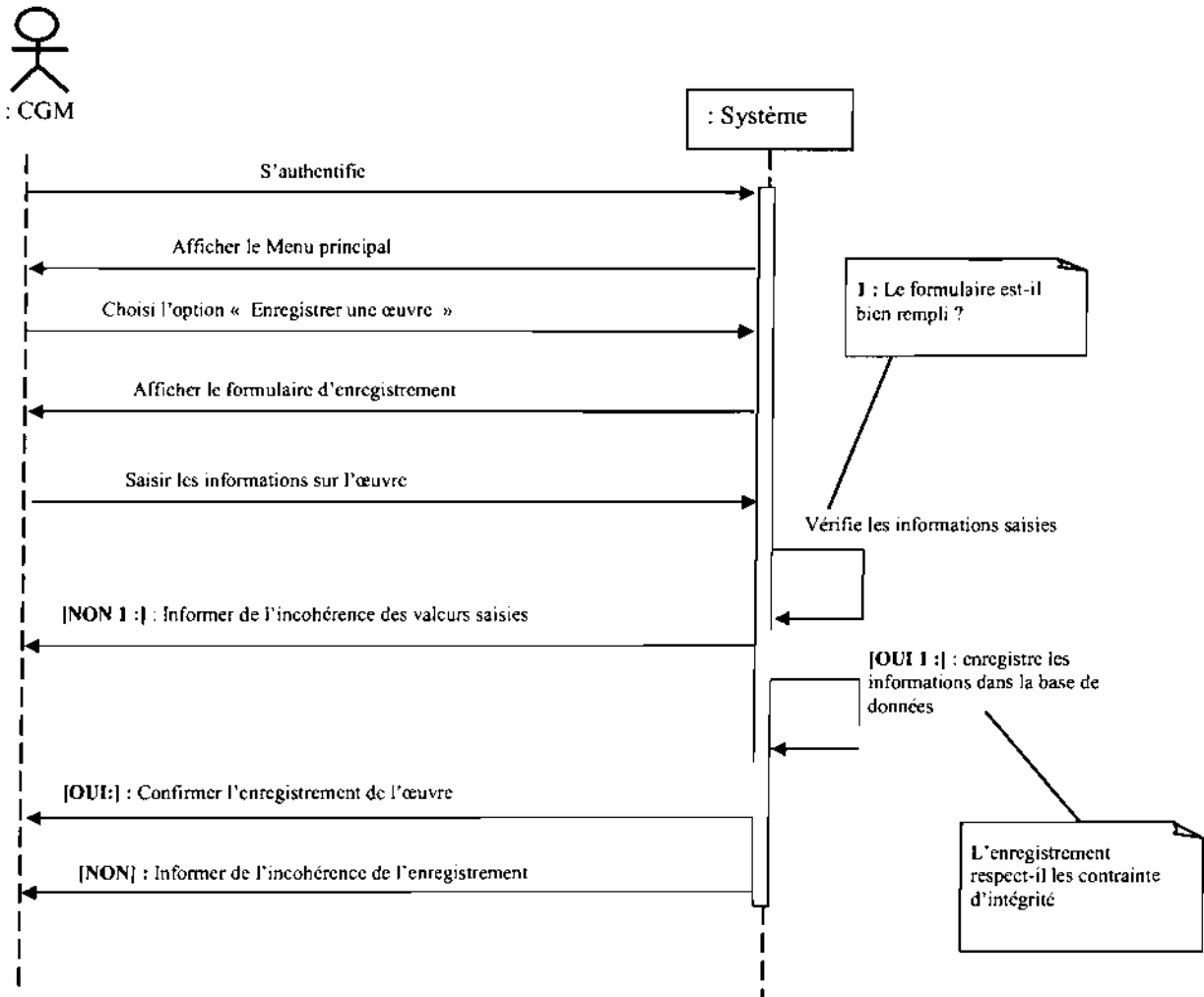


Diagramme4.13 Diagramme de séquence décrivant la procédure d'enregistrement d'une œuvre

– CU Supprimer/ Modifier les informations d'une œuvre

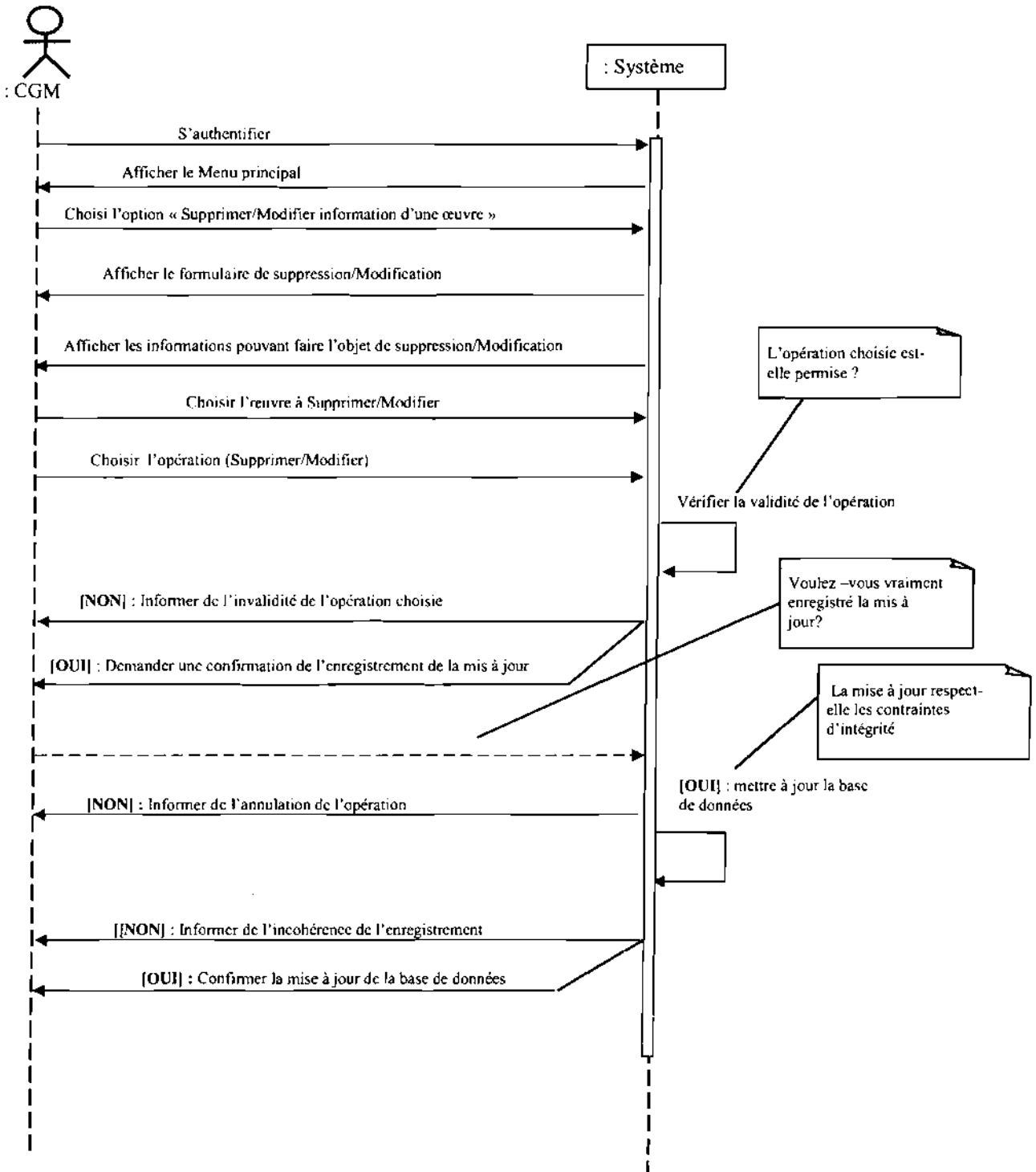


Diagramme4.14 Diagramme de séquence décrivant la procédure de suppression/modification des informations descriptives d'une œuvre

- CU Consulter une œuvre

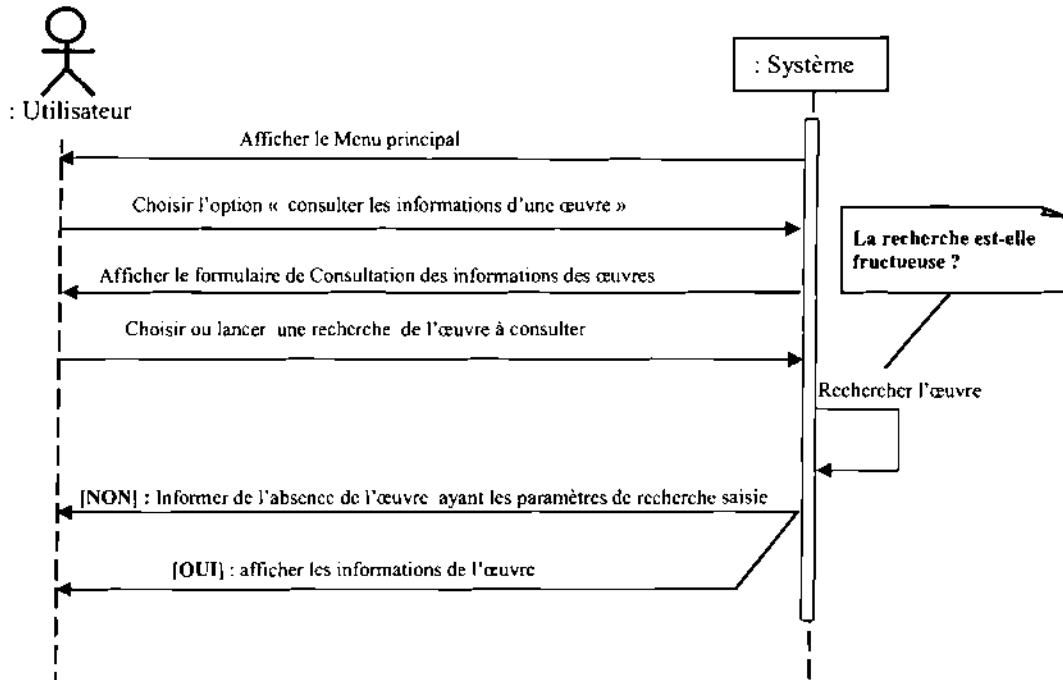


Diagramme4.15 Diagramme de séquence décrivant la procédure de consultation d'une œuvre

– CU Emprunter une œuvre

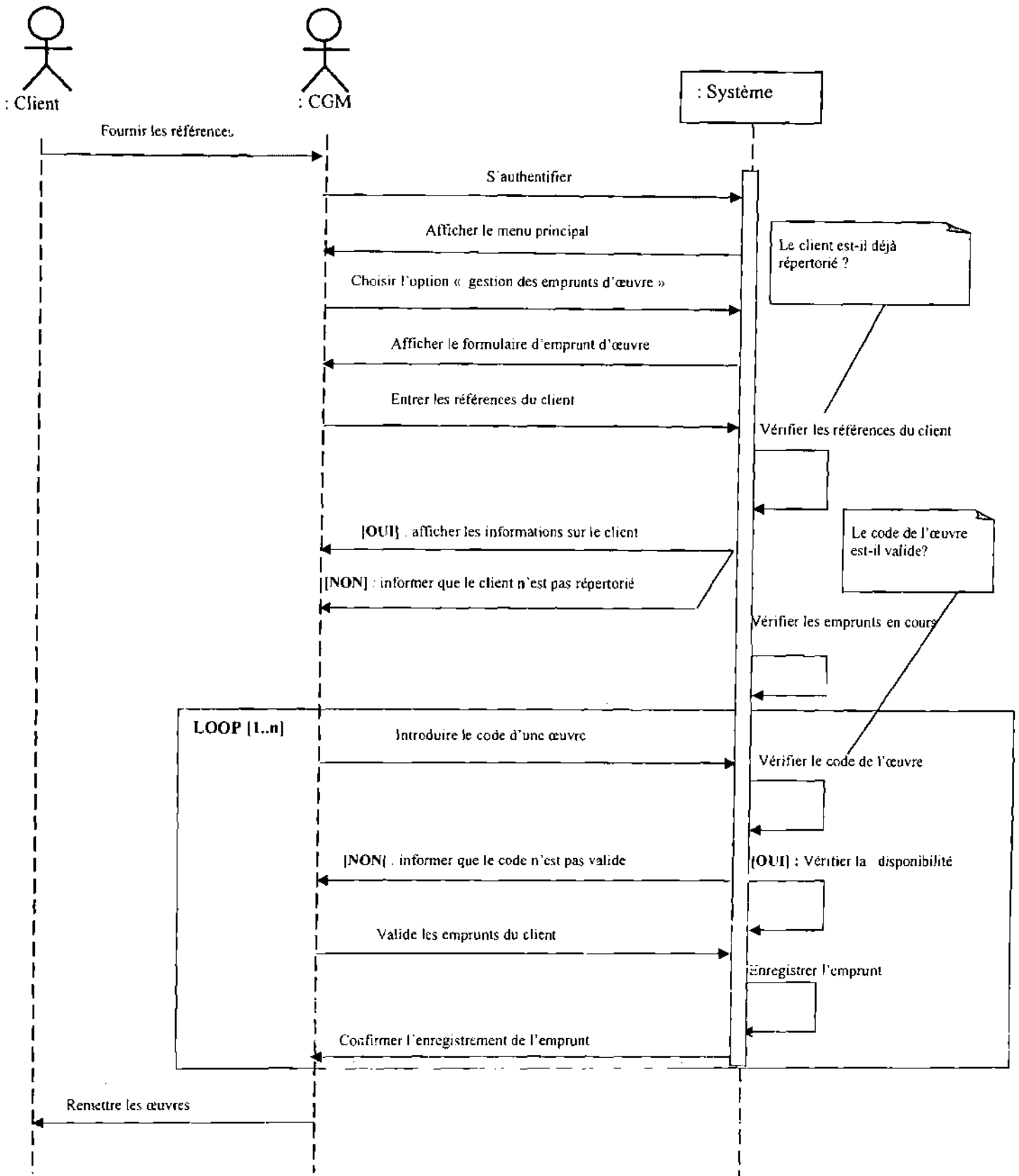


Diagramme4.16 Diagramme de séquence décrivant la procédure d'emprunt d'une œuvre

❖ CU Envoyer un message

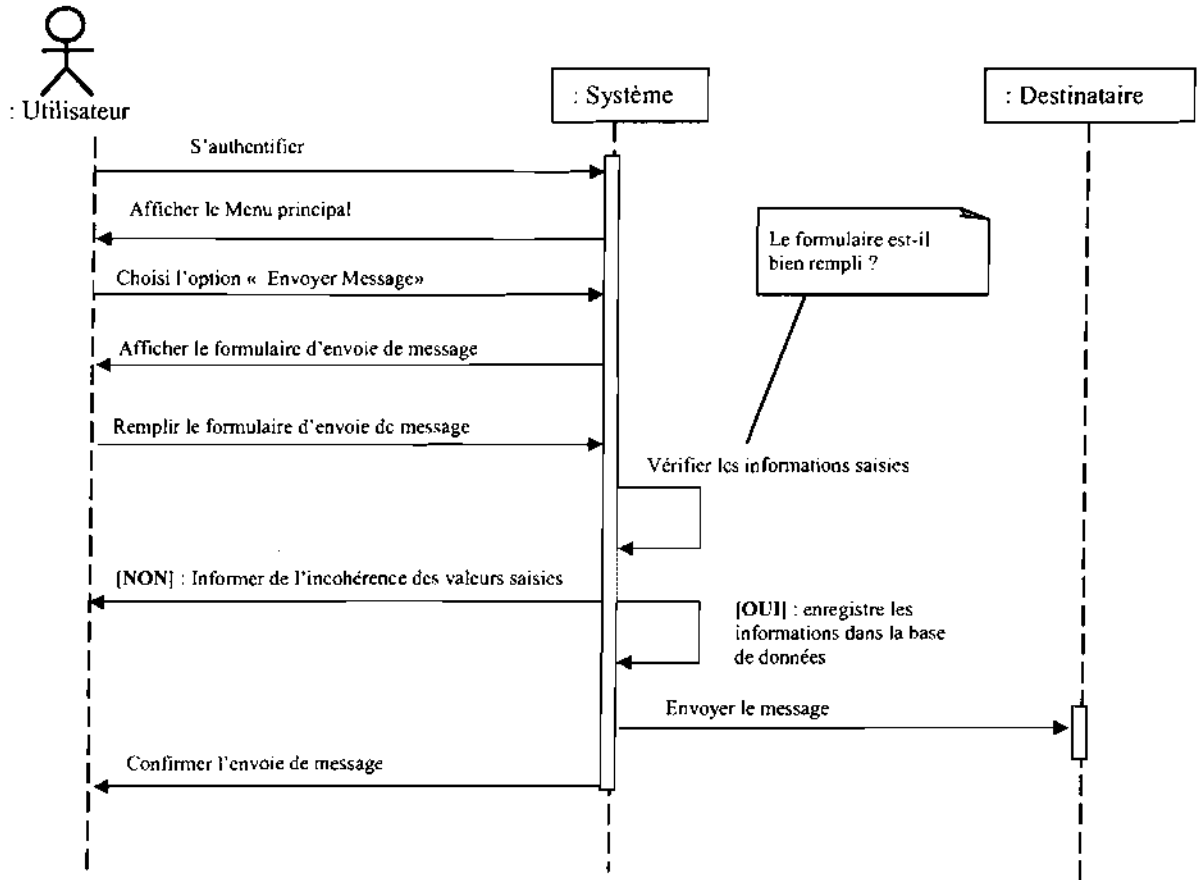


Diagramme4.17 Diagramme de séquence décrivant la procédure d'envoi d'un message

❖ Cas d'utilisation Gérer formation
- Gérer formation continue

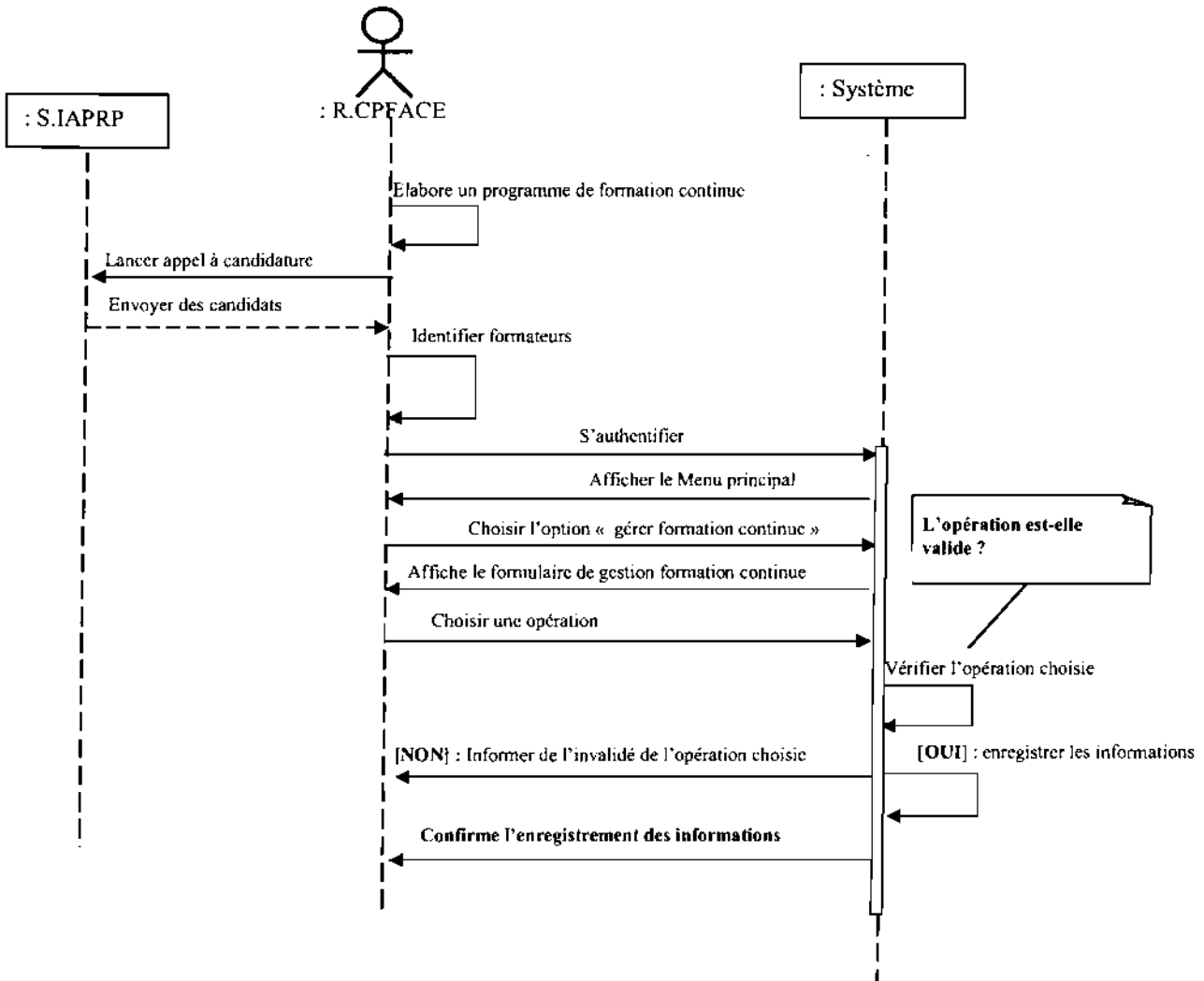


Diagramme4.18 Diagramme de séquence décrivant la procédure de gestion de la formation continue

- Gérer les notes

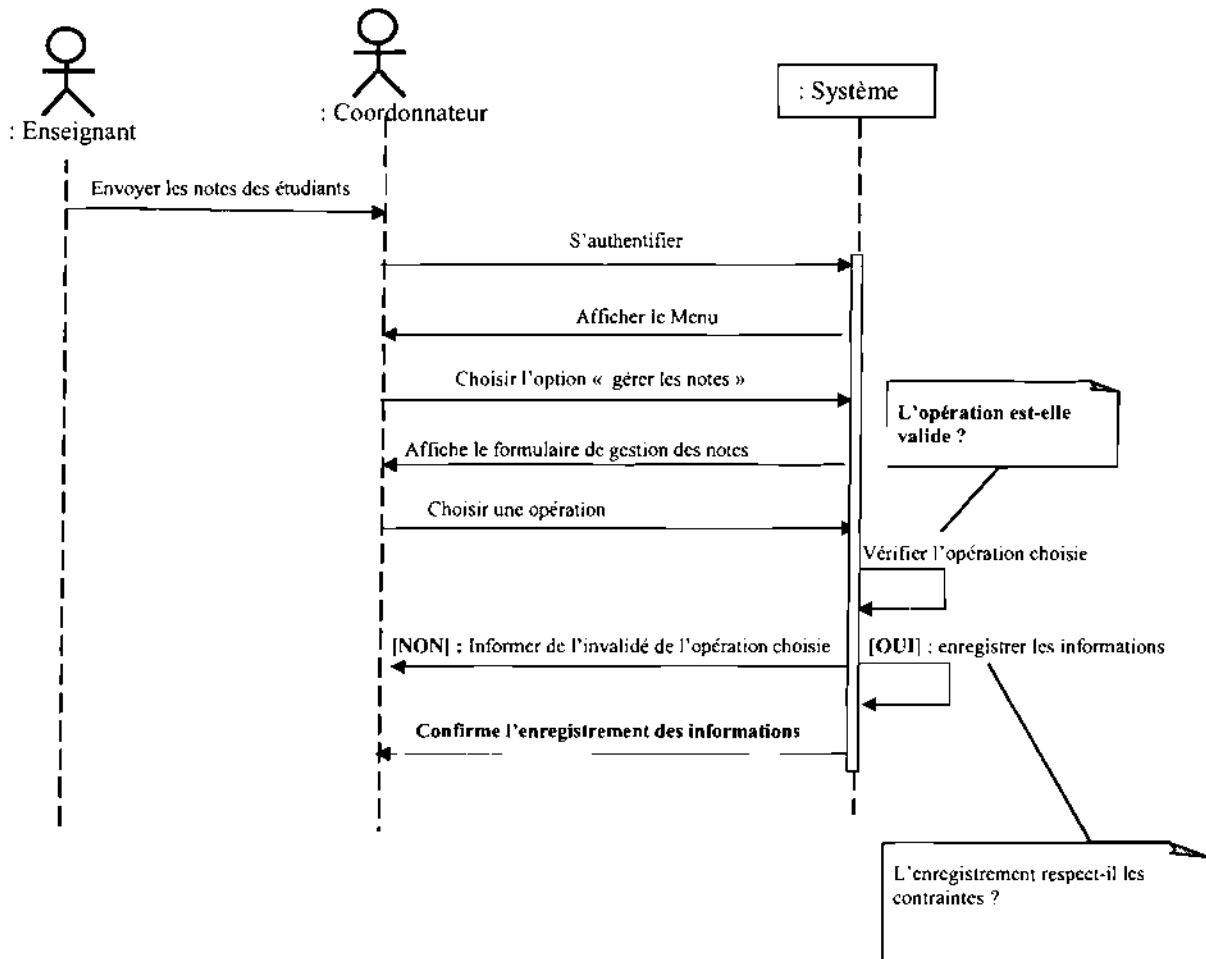


Diagramme4.19 Diagramme de séquence décrivant la procédure de gestion des notes

- Gérer les enseignants

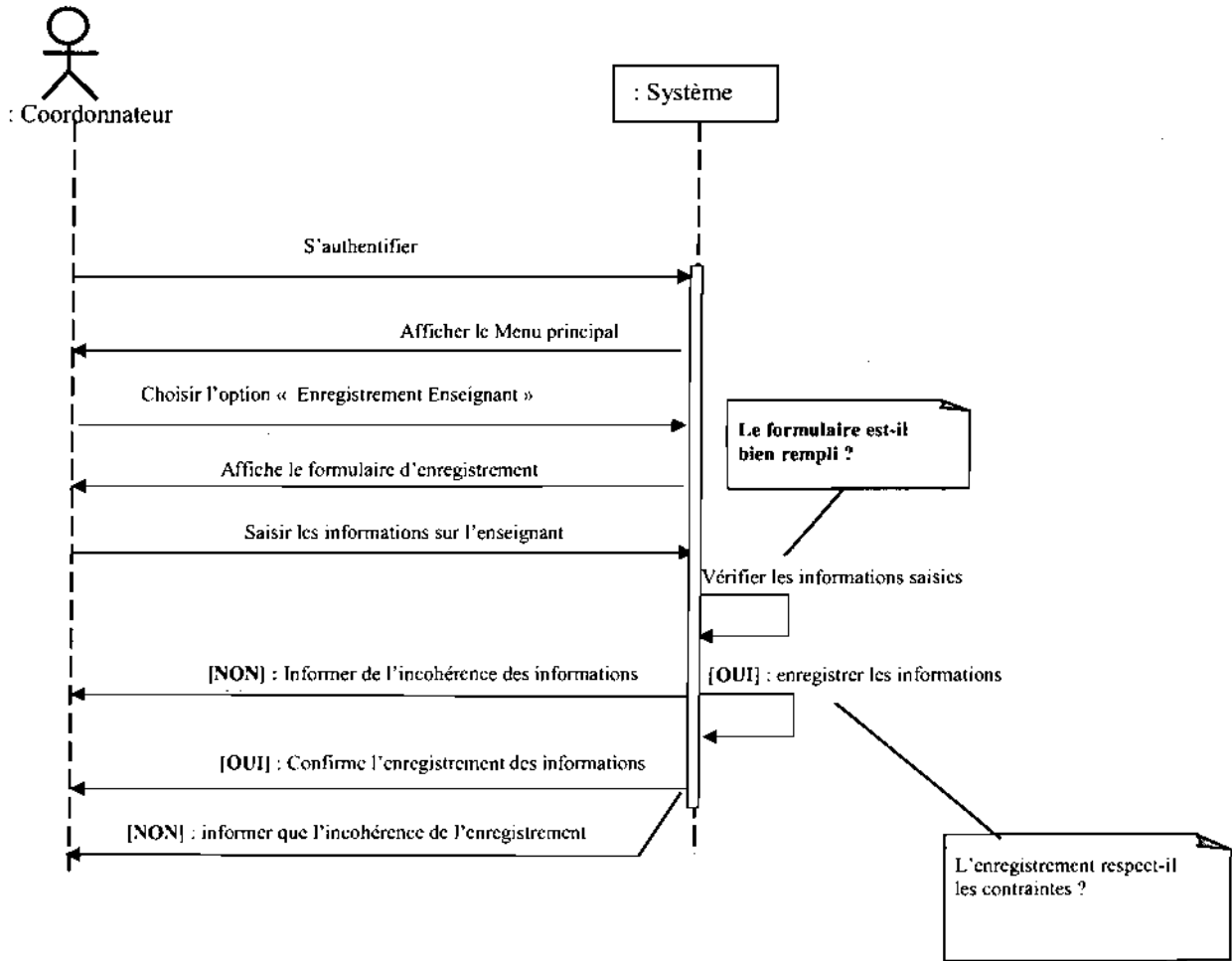


Diagramme4.20 Diagramme de séquence décrivant la procédure de gestion des enseignants

- Gérer les étudiants

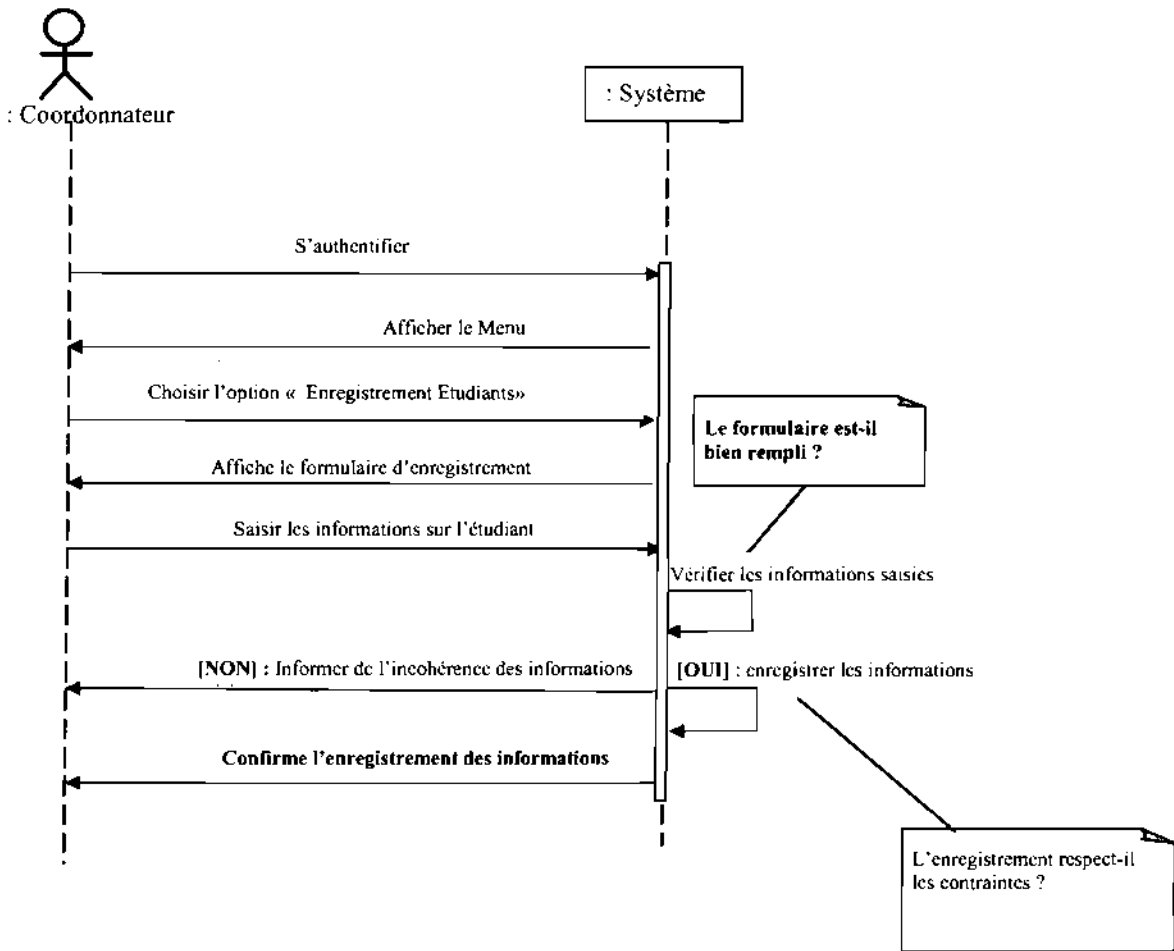


Diagramme4.21 Diagramme de séquence décrivant la procédure de gestion des étudiants

4.2-Conception générale

4.2.1- Besoins techniques : Choix technologiques

Cette partie présente différentes technologies nécessaires au développement de l'application. Elle décrit leurs utilités et leurs fonctionnalités afin de choisir les plus pertinentes, c'est-à-dire celles qui vont nous permettre de faciliter la phase de développement de l'application. Elle présente alors un comparatif entre plusieurs technologies et explique pourquoi telle ou telle technologie a été choisie.

Pour chacun des paragraphes, on fera apparaître les points suivants :

- Besoins de l'application,
- Descriptif des solutions,
- Conclusion et choix,

4.2.1.1.- Persistance des données

❖ Le besoin

Toutes les données nécessaires au bon fonctionnement de l'application sont stockées dans une base de données. Il existe alors deux moyens permettant d'exploiter ces données au sein de l'application :

- En faisant directement appel à la base de données par l'intermédiaire du langage de requête SQL ;
- En ayant une approche objets. Il s'agit dans ce cas d'utiliser un outil d'ORM (Object Relational Mapping) permettant de manipuler des objets reflétant les données contenues dans la base. On peut ainsi assurer la persistance de nos objets Java.

❖ Les solutions proposées

a) Comparatif JDBC / Hibernate / EJBs

Fonctionnalités	JDBC	Hibernate	EJBs
Mise en place	Très simple	Apprentissage nécessaire	Apprentissage nécessaire
Indépendance du SGBD	Oui	Oui	Oui
Persistance d'objet Java	Non	Oui	Oui
Mapping des objets Java	Non	Oui	Oui
Création de la base de données	Lui fournir le script SQL	Génération du code SQL automatiquement	-
Bufferisation des données afin d'éviter les connexions à la base	Non	Oui	Oui
Configuration	-	Fichier XML ou annotations Java	Fichier XML ou annotations Java
Déploiement	-	Intégration des librairies dans le projet	Nécessite un conteneur d'EJBs
Dernière version stable	JDBC 4.0	Hibernate Core 3.0.5	EJBs 3.0

Tableau 4.1 Comparatif JDBC / Hibernate / EJBs

b) JDBC

JDBC est l'acronyme de **Java Data Base Connectivity** et désigne une API définie par Sun pour permettre un accès aux bases de données relationnelles (SGBD) avec Java.

JDBC a pour objectif principal de permettre aux développeurs d'écrire un code indépendant de la base de données et du moyen de connectivité utilisé.

Voici les principaux avantages de JDBC :

- Une interface uniforme permettant un accès homogène au SGBD ;
- Simple à mettre en œuvre ;
- Indépendant du SGBD cible ;
- Supportant les fonctionnalités de base du langage SQL.

L'architecture JDBC standard comporte trois composants :

- le code de l'application, qui interagit avec la base de données,
- le Driver Manager fournit par SUN qui fait le lien entre l'application et les pilotes,
- le driver JDBC, fournit par le SGBD auquel il est propre, qui convertit les requêtes

JDBC (dans le code de l'application) en requêtes propres à la base.

L'API JDBC comporte 4 classes importantes : DriverManager, Connection, Statement (et PreparedStatement), et ResultSet, chacune correspondant à une étape de l'accès aux données.

Classe	Rôle
DriverManager	charge et configure le driver de la base de données.
Connection	réalise la connexion et l'authentification à la base de données.
Statement (et PreparedStatement)	contient la requête SQL et la transmet à la base de données.
ResultSet	permet de parcourir les informations retournées par la base de données dans le cas d'une sélection de données

Tableau 4.2 Classes importantes de l'API JDBC

Utilisation de JDBC

Chargement du pilote

Pour se connecter à une base en utilisant un driver spécifique, la documentation du driver fournit le nom de la classe à utiliser. Par exemple, si le nom de la classe est jdbc.DriverXXX, le chargement du driver se fera avec le code « *Class.forName* ("jdbc.DriverXXX") ».

Exemple de chargement du pilote pour une base PostgreSQL:

```
Class.forName ("postgresql.Driver");
```

Il n'est pas nécessaire de créer une instance de cette classe et de l'enregistrer avec le DriverManager car l'appel à *Class.forName* le fait automatiquement : ce traitement charge le pilote et crée une instance de cette classe.

N.B: La méthode static *forName* () de la classe *Class* peut lever l'exception

java.lang.ClassNotFoundException.

L'établissement de la connexion

Pour se connecter à une base de données, il faut instancier un objet de la classe *connection* en lui précisant sous forme d'URL la base à accéder.

Exemple de code pour connexion avec spécification DriverManager

Objets d'accès

Une fois la connexion établie, il est possible d'exécuter des ordres SQL. Les objets qui peuvent être utilisés pour obtenir des informations sur la base de données sont :

Exécution des requêtes

Les requêtes d'interrogation SQL sont exécutées avec les méthodes d'un objet Statement que l'on obtient à partir d'un objet Connection.

Exemple :

```
ResultSet résultats = null;
String requete = "SELECT * FROM Œuvre";
try {
Statement stmt = con.createStatement ();
résultats = stmt.executeQuery (requete);
} Catch (SQLException e) {
//traitement de l'exception
}
```

Un objet de la classe Statement permet d'envoyer des requêtes SQL à la base. La création d'un objet Statement s'effectue à partir d'une instance de la classe Connection

Exemple :

```
Statement stmt = con.createStatement ();
```

Pour une requête de type interrogation (SELECT), la méthode à utiliser est executeQuery (de la classe Statement). Pour des traitements de mise à jour, il faut utiliser la méthode

executeUpdate (). Lors de l'appel à la méthode d'exécution, il est nécessaire de lui fournir en paramètre la requête SQL sous forme de chaîne. Le résultat d'une requête d'interrogation est renvoyé dans un objet de la classe ResultSet par la méthode executeQuery ().

Exemple :

```
ResultSet rs = stmt.executeQuery ("SELECT * FROM Œuvre");
```

La méthode executeUpdate () retourne le nombre d'enregistrements qui ont été mis à jour

Exemple :

```
//insertion d'un enregistrement dans la table Œuvre
requete = "INSERT INTO Œuvre VALUES ('0112','A la sueur du
front',...)";
try {
Statement stmt = con.createStatement ();
int nbMaj = stmt.executeUpdate (requete);
Affiche ("nb mise a jour = "+nbMaj);
} Catch (SQLException e) {
e.printStackTrace ();
}
```

Lorsque la méthode executeUpdate () est utilisée pour exécuter un traitement de type DDL

(Data Definition Langage : définition de données) comme la création d'une table, elle retourne 0. Si la méthode retourne 0, cela peut signifier deux choses : le traitement de mise à jour n'a affecté aucun enregistrement ou le traitement concernait un traitement de type DDL. Si l'on utilise executeQuery() pour exécuter une requête SQL ne contenant pas d'ordre SELECT, alors une exception de type SQLException est levée.

Exemple :

```
requete = "INSERT INTO Œuvre VALUES ('012','le nouvel exode ou la  
ruée vers l'or)";  
try {  
Statement stmt = con.createStatement ();  
ResultSet résultats = stmt.executeQuery (requete);  
} catch (SQLException e) {  
e.printStackTrace ();  
}
```

Cette requête affiche le résultat suivant :

```
java.sql.SQLException: No ResultSet was produced  
java.lang.Throwable (java.lang.String)  
java.lang.Exception (java.lang.String)  
java.sql.SQLException (java.lang.String)  
java.sql.ResultSet  
sun.jdbc.odbc.JdbcOdbcStatement.executeQuery (java.lang.String)  
Void testjdbc.TestJDBC1.main (java.lang.String [])
```

Cette exécution peu s'avérer problématique car la requête est quand même effectuée. Dans l'exemple, un nouvel enregistrement est créé dans la table.

Remarque : Il n'est pas nécessaire de définir un objet Statement pour chaque ordre SQL. En effet, il est possible de définir un et de le réutiliser.

b) Hibernate

Hibernate est une surcouche à la base de données. Elle fournit un ensemble de méthodes et fonctions permettant de rendre persistant les objets Java d'une application. Il faut donc dans un premier temps permettre à Hibernate d'accéder au système de gestion de base de données utilisé en lui fournissant les paramètres adéquats. On peut ensuite définir les règles qui permettront à nos objets Java de devenir persistant. Cette étape se nomme le « mapping ».

Elle est très importante puisqu'elle va définir la façon dont seront organisées les tables en base de données. En effet les classes Java de notre application seront par ce processus représentées par des tables dans notre base de données. Le mapping permet de définir les différentes propriétés des tables de la base mais aussi les relations qu'il peut y avoir entre celles-ci.

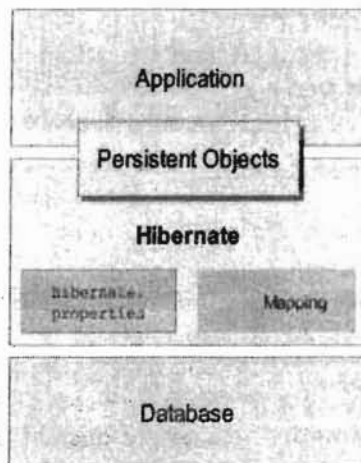


Figure 4.1 Mapping avec Hibernate

d) EJBs¹²

Les EJB sont des composants qui possèdent alors certaines caractéristiques comme la réutilisabilité, la possibilité de s'assembler pour construire une application, etc.

Les EJB et les beans n'ont en commun que d'être des composants. Les java beans sont des composants qui peuvent être utilisés dans toutes les circonstances. Les EJB doivent obligatoirement s'exécuter dans un environnement serveur dédié.

Les EJB sont parfaitement adaptés pour être intégrés dans une architecture trois tiers ou plus. Dans une telle architecture, chaque tiers assure une fonction particulière :

- le client « léger » assure la saisie et l'affichage des données,
- sur le serveur, les objets métiers contiennent les traitements. Les EJB sont spécialement conçus pour constituer de telles entités,
- une base de données assure la persistance des informations.

Les EJB s'exécutent dans un environnement particulier : le serveur d'EJB. Celui-ci fournit un ensemble de fonctionnalités utilisées par un ou plusieurs conteneurs d'EJB qui constituent le serveur d'EJB. En réalité, c'est dans un conteneur que s'exécute un EJB et il lui est impossible de s'exécuter en dehors.

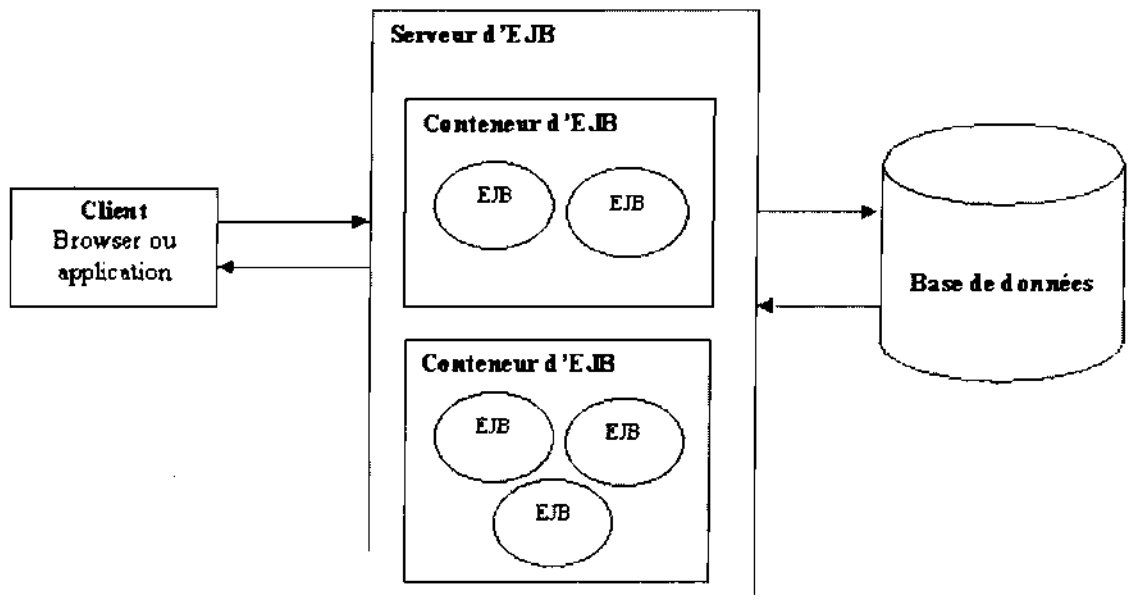


Figure 4.2 Environnement d'exécution d'EJB

Le conteneur d'EJB propose un certain nombre de services qui assurent la gestion :

- du cycle de vie du bean,
- de l'accès au bean,
- de la sécurité d'accès,
- des accès concurrents,
- des transactions.

Les entités externes au serveur qui appellent un EJB ne communiquent pas directement avec celui-ci. Les accès au EJB par un client se fait obligatoirement via le conteneur. Un objet héritant de EJBObject assure le dialogue entre ces entités et les EJB via le conteneur. L'avantage de passer par le conteneur est que celui-ci peut utiliser les services qu'il

¹² Enterprise Java Beans

propose et libérer ainsi le développeur de cette charge de travail. Ceci permet au développeur de se concentrer sur les traitements métiers proposés par le bean.

Il existe de nombreux serveurs d'EJB commerciaux : BEA Weblogic, IBM WebSphere, Sun IPlanet, Macromedia JRun, Borland AppServer, etc. Il existe aussi des serveurs d'EJB open source dont les plus avancés sont JBoss et Jonas.

❖ Solution retenu : JDBC

Nous retenons JDBC étant données qu'il a pour objectif principale de permettre aux développeurs d'écrire un code indépendant de la base de données et du moyen de connectivité utilisé. Comparativement à Hibernate et aux EJBs, la mise place de JDBC reste simple.

4.2.1.2- Framework web

❖ Le besoin

La partie web de notre application se repose sur une architecture Web Java .Voici une liste des principaux composants pouvant être assemblés pour constituer cette application Web :

- Servlets¹³,
- Java Server Pages¹⁴(JSP),
- Bibliothèques de balises personnalisées (taglibs),
- Classes utilitaires et composants métier,
- Documents statiques, par exemple pages HTML, images, scripts JavaScript, etc,
- Méta informations décrivant l'application Web.

La plate-forme J2EE NetBeans propose bon nombre de possibilités, via notamment les servlets Java et les pages JSP. Dans les deux cas, on peut générer, en dynamique, un contenu HTML dépendant des requêtes précédentes du client ainsi que d'éventuelles informations stockées dans la base de données. Mais ces deux techniques présentent toutes les deux des inconvénients : dans les deux cas, nous n'avons pas de séparation entre le code HTML et le code Java.

On peut cependant envisager la séparation des deux types de code: traitement de la requête et génération du flux HTML. Le gain principal réside dans une meilleure structuration du code et donc une meilleure maintenance.

Pour obtenir une telle séparation, on pourrait imaginer d'envoyer les requêtes HTTP vers des servlets Java. Ces servlets ne contiendront donc que du code "métier". Une fois le traitement terminé, chaque servlet pourrait renvoyer l'exécution vers une JSP qui aurait alors à charge de générer le code HTML de retour au navigateur.

Il existe un grand nombre de framework dédié aux applications Web et pouvant implémenter ce type d'architecture.

¹³ Les servlets Java sont des classes Java dérivant d'une classe bien précise.

¹⁴ Une JSP ressemble souvent à une page HTML et contenant du code java

Processus Struts

L'interaction entre les trois composants est orchestrée par le contrôleur principal:

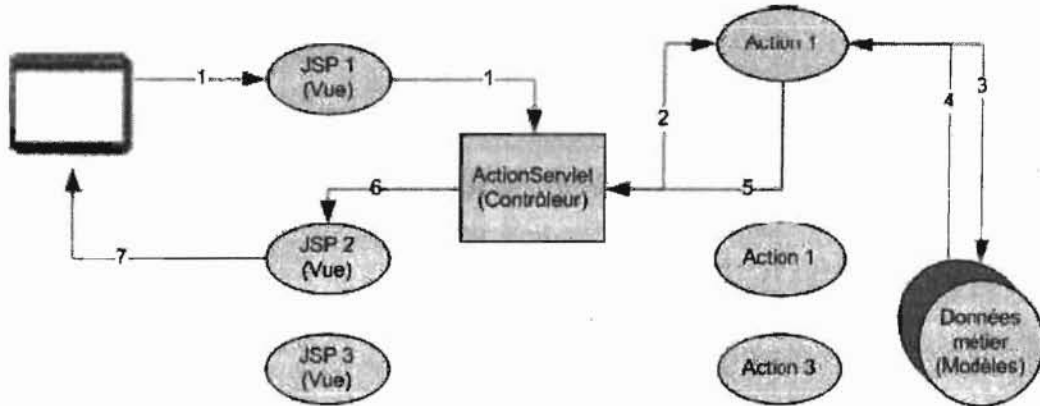


Figure 4. 3 Cycle de vie d'une requête http.

Afin de mieux comprendre le fonctionnement du framework, le cycle de vie d'une requête

HTTP est détaillé ci-dessous:

1. Le client envoie sa requête HTTP à l'application. Cette requête est prise en charge par le contrôleur principal, en l'occurrence ActionServlet,
2. La requête est redirigée vers le contrôleur adéquat,
3. Le contrôleur choisi prend en charge le traitement de la requête. Un dialogue avec les services métiers est entamé si nécessaire,
4. Le modèle fournit les données demandées,
5. Le contrôleur principal est notifié du résultat du traitement. En cas de succès, les données sont encapsulées dans des JavaBean (ActionForm) et transmis à la JSP sélectionnée par le contrôleur,
6. La JSP construit la réponse suivant les données transmises,
7. La réponse est envoyée au navigateur.

4.2.2 Répartition de l'application en paquetage

Il s'agit là de présenter les sous systèmes ou modules qui formeront l'application.

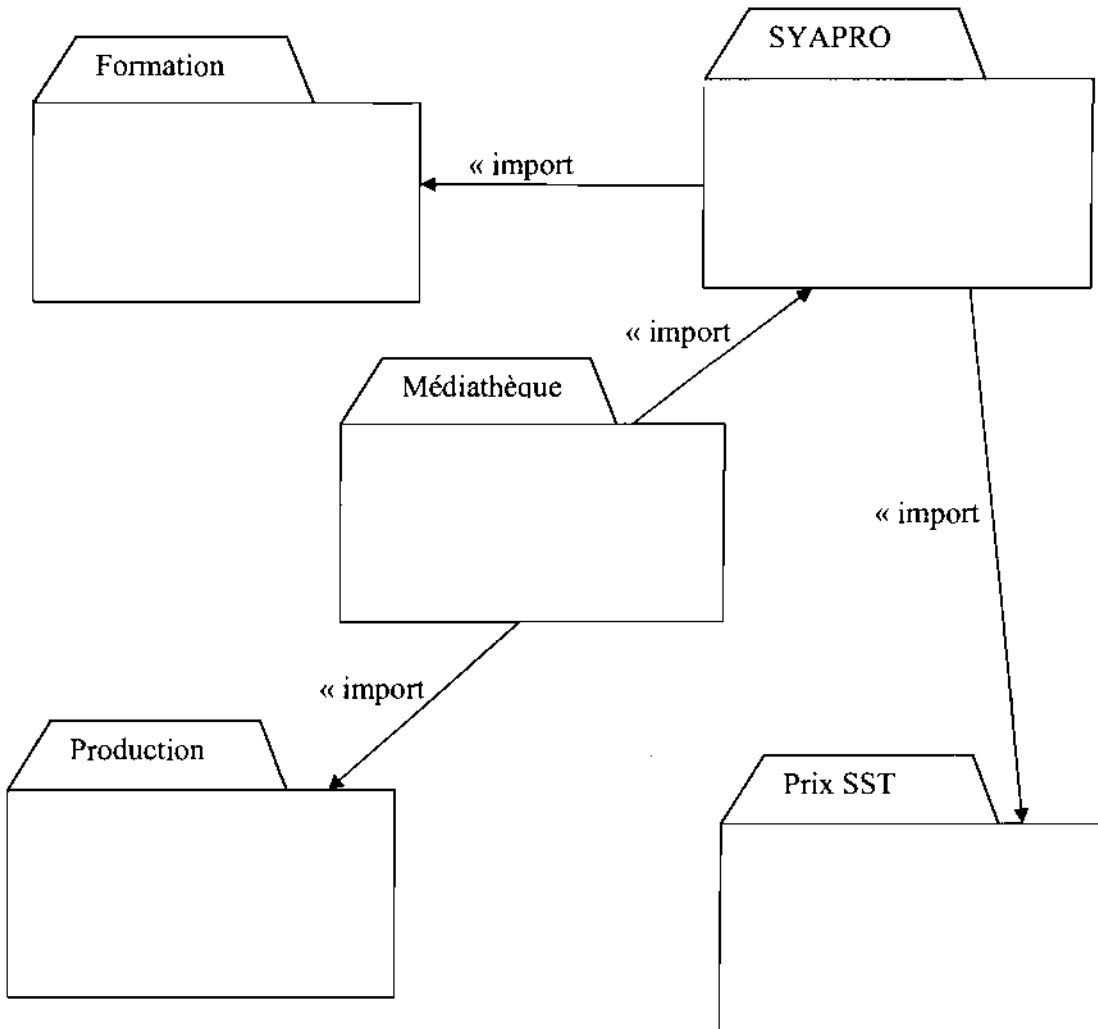


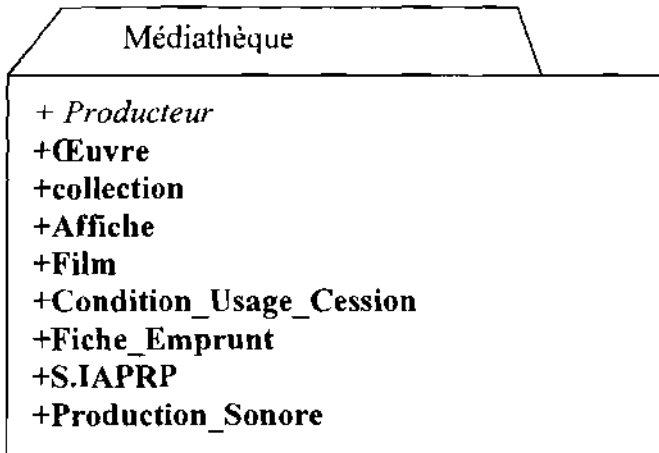
Diagramme4.22 Diagramme de paquetage du futur système

4.3-Conception détaillée

Dans cette activité Il s'agira de décomposer le travail d'implémentation en portions plus agréables.

4. 3.1 Représentation des diagrammes de classe¹⁶ des différents paquetages

Le Paquetage Médiathèque



Le diagramme de classe du paquetage Médiathèque

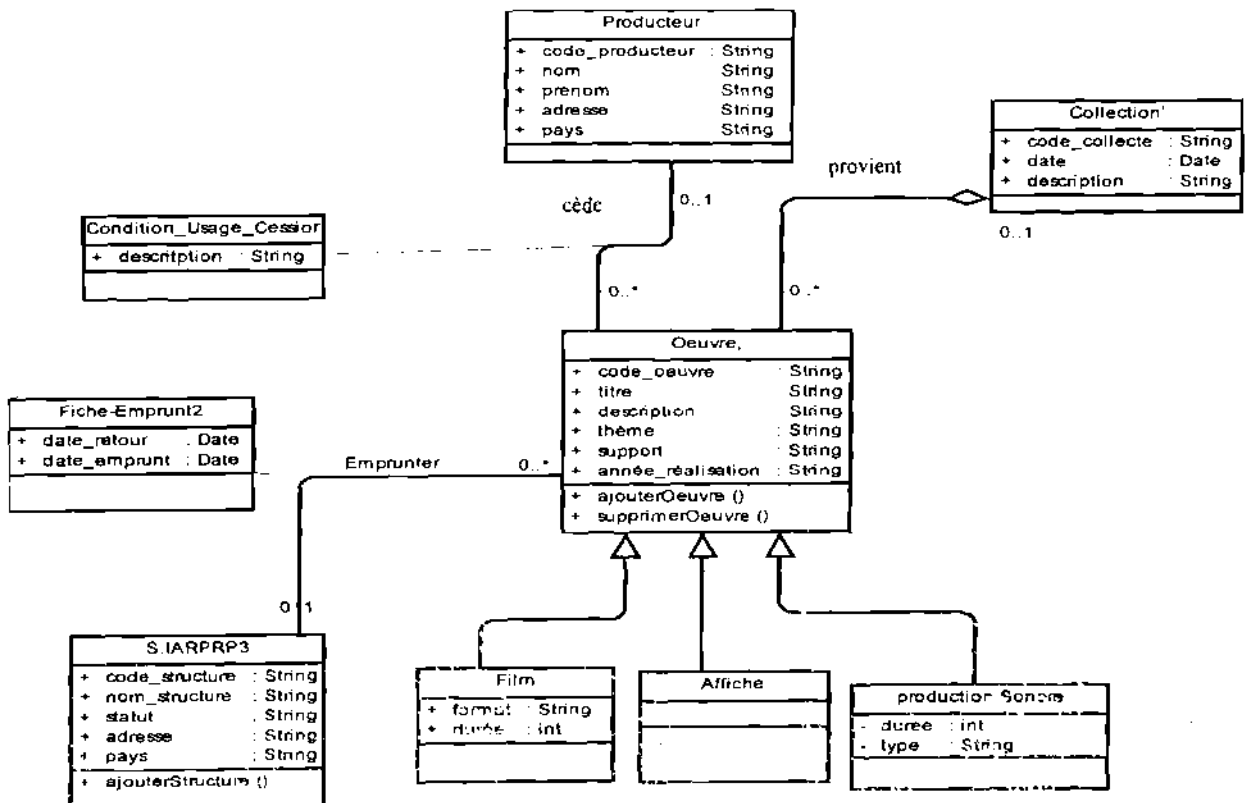
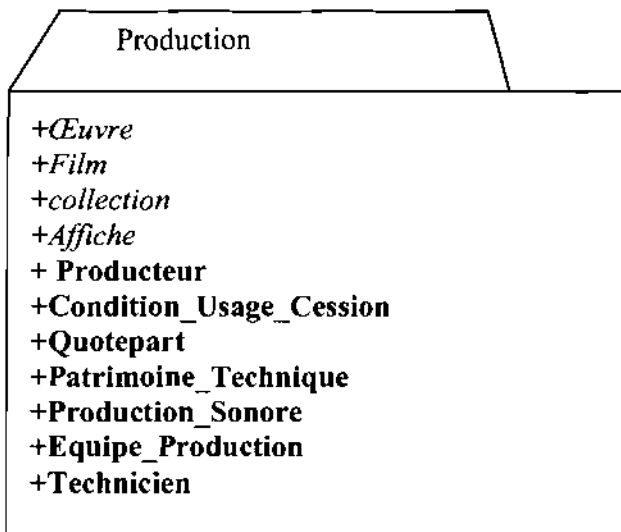


Diagramme4.23 Diagramme de classe du paquetage Médiathèque

¹⁶ Les concepts et les formalismes du diagramme de classe sont décrits en Annexes partie 4

Le Paquetage Production



Le diagramme de classe du paquetage Production

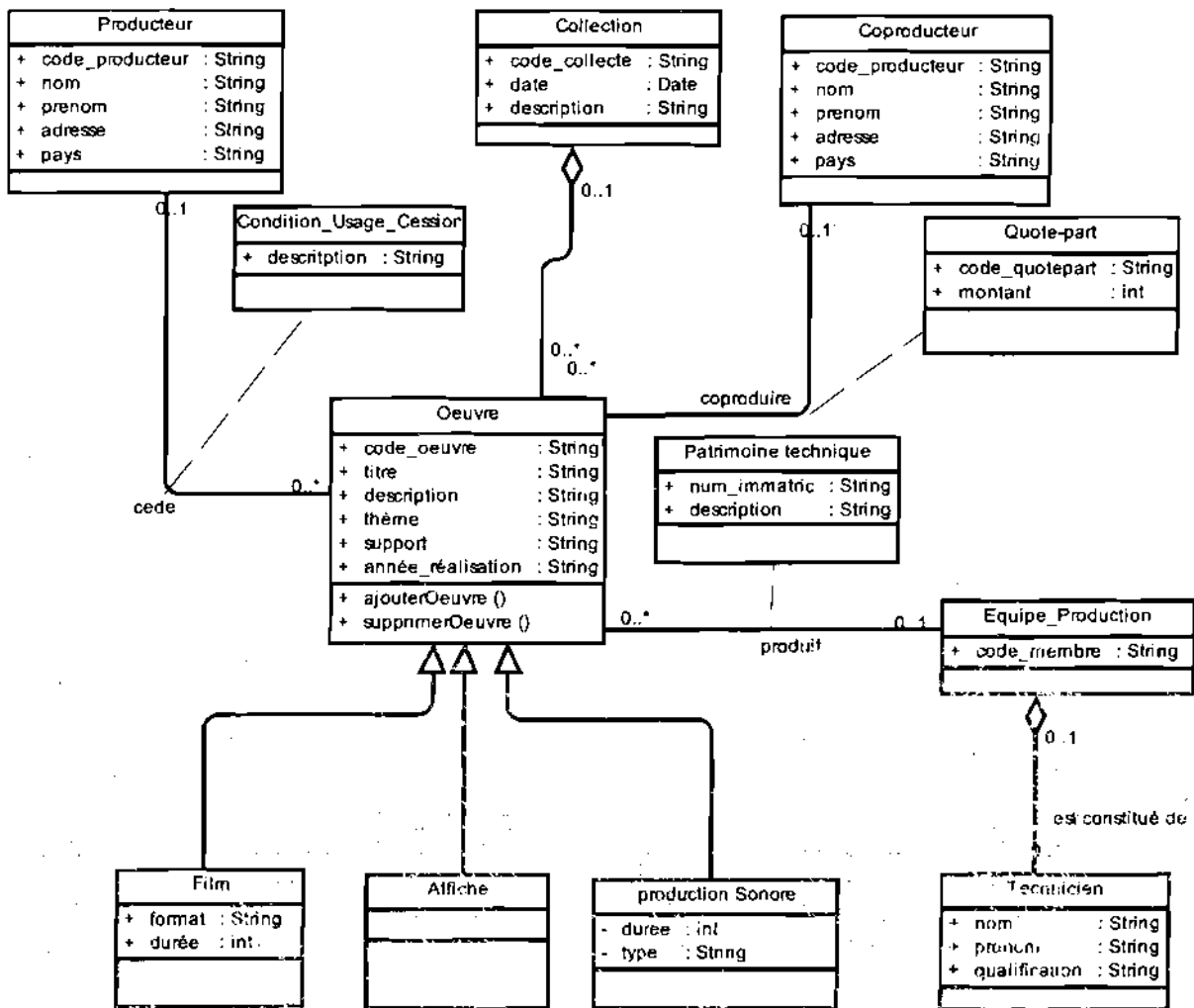
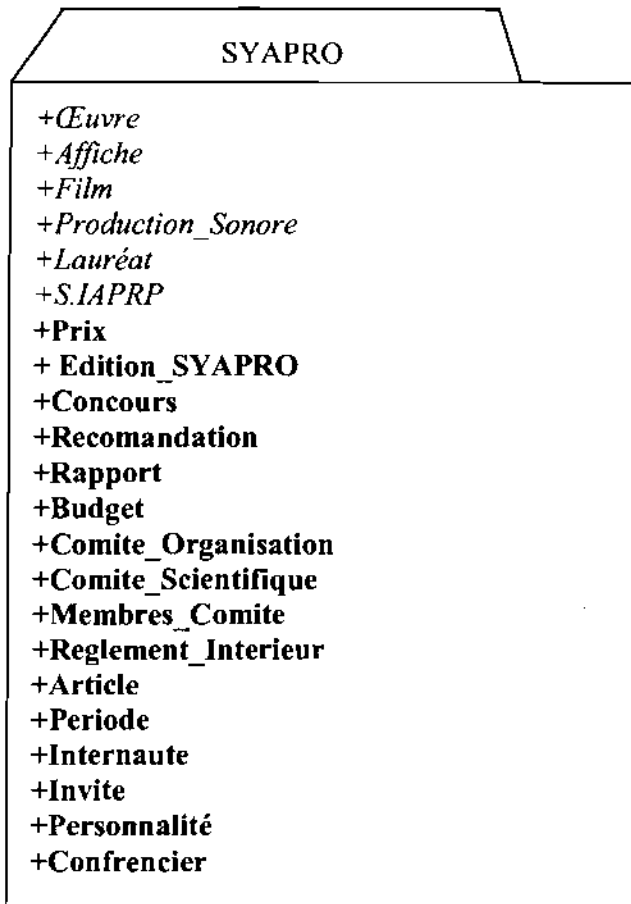


Diagramme4.24 Diagramme de classe du paquetage Production

Le Paquetage SYAPRO



Le diagramme de classe du paquetage SYAPRO

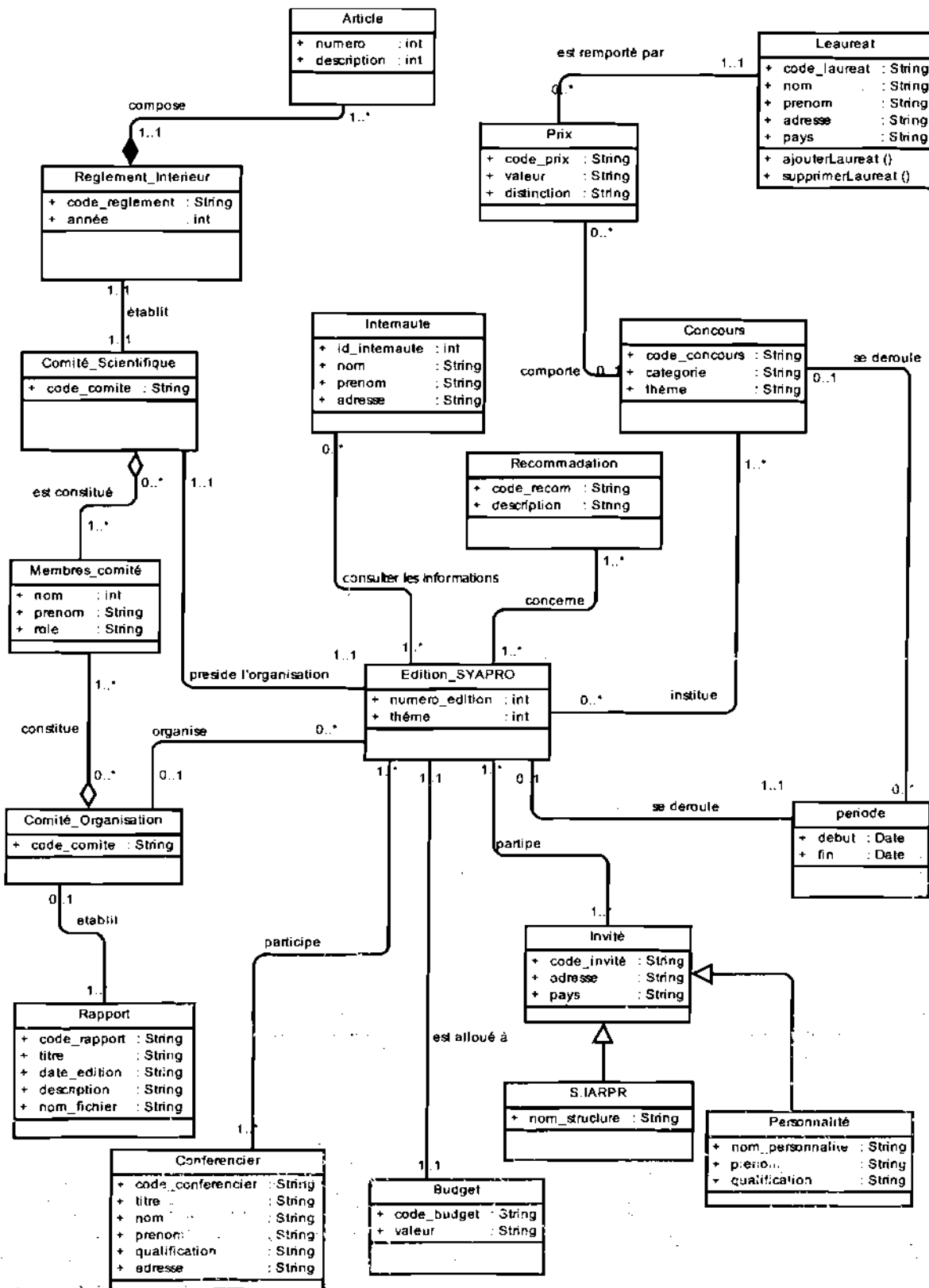
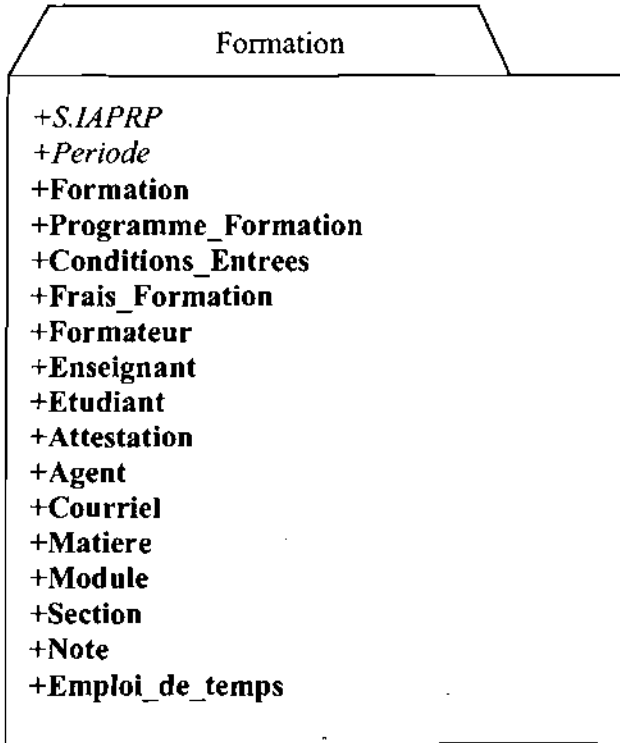


Diagramme4.25 Diagramme de classe du paquetage SYAPRO

Le Paquetage Formation



Le diagramme de classe du paquetage Formation

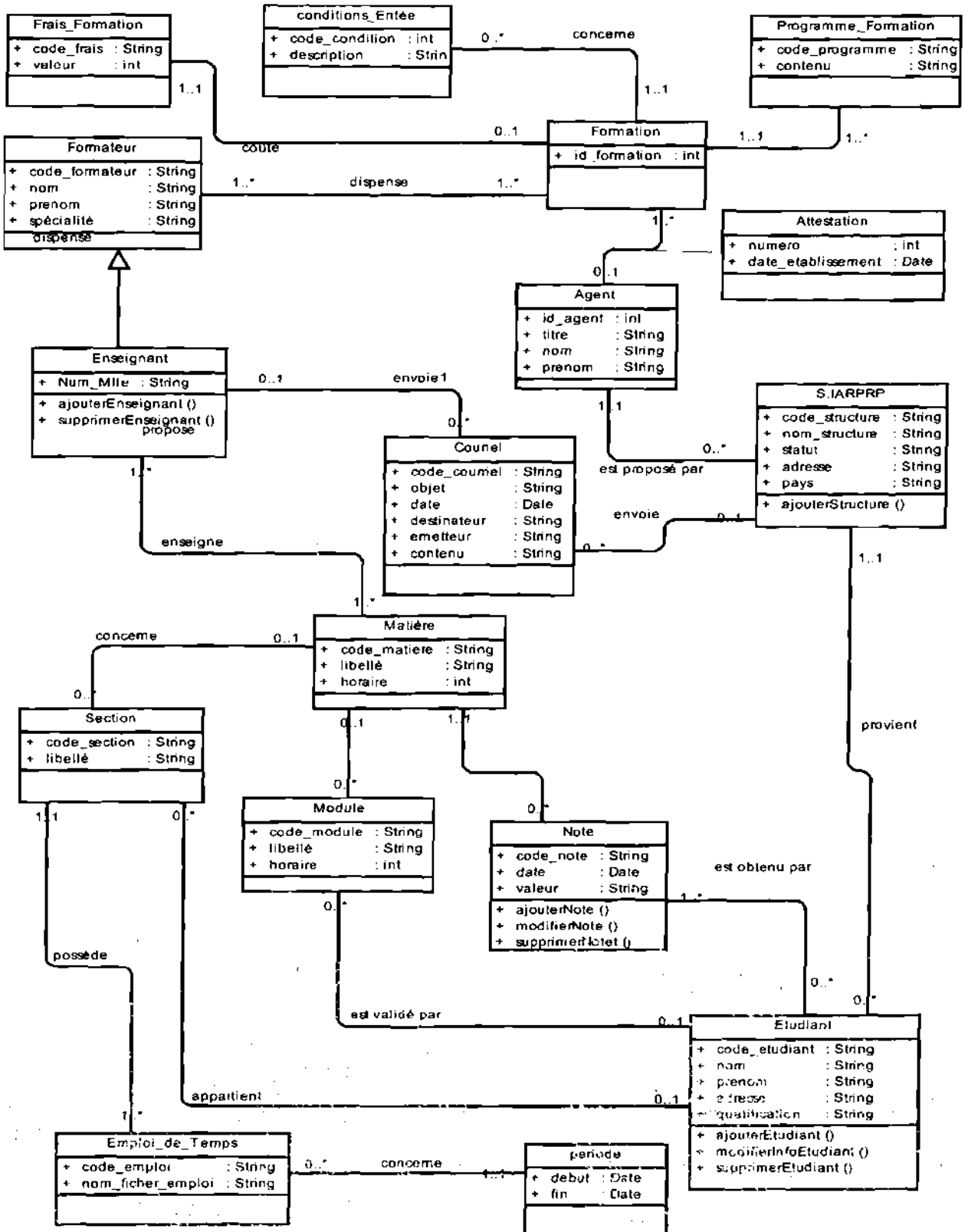
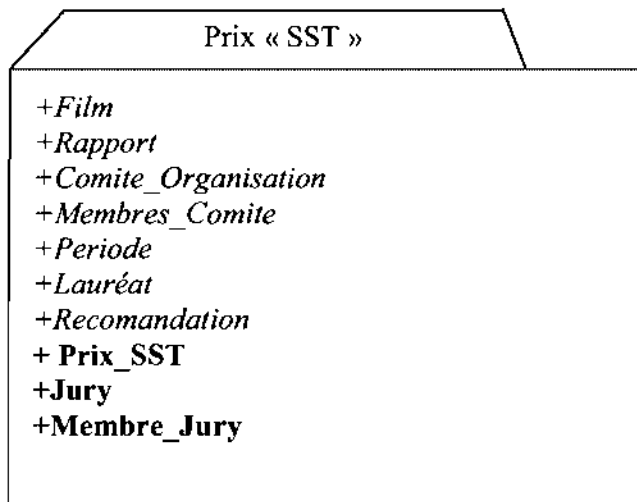


Diagramme4.26 Diagramme de classe du paquetage Formation

Le Paquetage prix « Santé et sécurité au Travail »



Le diagramme de classe du paquetage prix « Santé et sécurité u Travail »

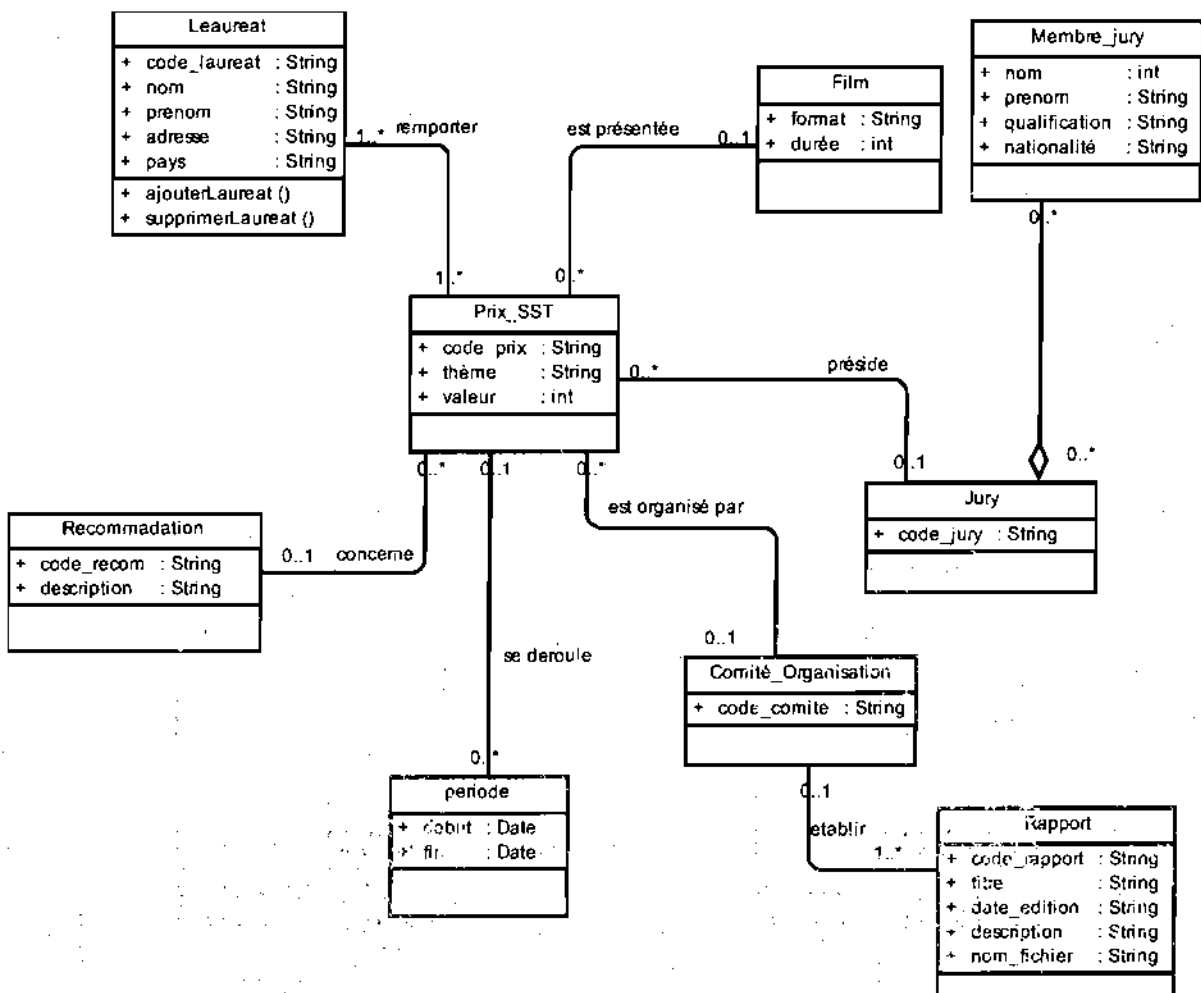


Diagramme4.26 Diagramme de classe du paquetage Formation prix « Santé et sécurité du Travail »

4. 3.2 Les règles de gestion

- RG1** : une structure membre d'IAPRP peut inscrire plusieurs Agents à une formation donnée ;
RG2 : une structure membre d'IAPRP peut emprunter plusieurs œuvres pour consultation ;
RG3 : une œuvre peut être produite par plusieurs producteurs ;
RG4 : une œuvre appartient à une collection donnée et est composée de film, d'affiche et de document ;
RG5 : un lauréat peut remporter plusieurs prix dans différents concours ;
RG6 : un concours comprend un à plusieurs prix ;
RG7 : Il est institué plusieurs concours lors d'une édition SYAPRO ;
RG8 : L'organisation d'une édition de SYAPRO est présidée par un comité scientifique ;
RG9 : le comité scientifique établit un règlement intérieur qui comporte plusieurs articles ;
RG10 : une formation est animée par un ou plusieurs formateurs ;
RG11 : un enseignant du CPFACE peut être un formateur ;
RG12 : un enseignant peut dispenser un ou plusieurs matières ;
RG14 : une matière appartient à un module de formation ;
RG15 : un étudiant est inscrit dans une seule section de formation ;
RG16 : la production d'une œuvre nécessite la mise en place d'une équipe de production ;
RG17 : une équipe de production est composée d'au moins d'un technicien ;
RG18 : A chaque édition du SYAPRO, il est invité des conférenciers, des personnalités et des institutions ;
RG19 : Pour chaque édition du SYAPRO un comité scientifique, un jury et un comité d'organisation sont mis en place ;
RG20 : Un rapport d'exécution et financier est produit à la fin de chaque édition SYAPRO ;
RG21 : Des recommandations sont émises à la fin de chaque édition SYAPRO ;
RG22 : Un budget est alloué à chaque édition du SYAPRO ;
RG23 : un budget est également alloué au Prix « Santé et Sécurité au Travail » du FESPACO ;
RG24 : Il est crée un jury dénommé Jury prix Spécial « Santé et Sécurité au Travail » au FESPACO ;
RG25 : un jury se compose de plusieurs membres ;

4. 3.3 Description des classes

Classe : Œuvre, classe généralisant de toute œuvre du système				
Attributs				
Visibilité	Nom		Type	Description
privé	Code_oeuvre		Alphanumérique	Le code de l'œuvre
public	Titre		Alphanumérique	Le titre de l'œuvre
public	Description		Alphanumérique	La description de l'œuvre
Public	Thème		Alphanumérique	Le thème traité par l'œuvre
Public	support		Alphanumérique	Le type de support
Public	Annee_realisation		Alphanumérique	L'année de réalisation
Méthodes				
Visibilité	Type	Nom	Paramètre	description
Public	Booléen	ajouterOeuvre	Void	Permet d'ajouter une œuvre
Public	Void	VisualiserOeuvre	Void	Permet de visualiser une œuvre
Public	Booléen	ModifierOeuvre	Void	Permet de modifier les informations d'une œuvre
Public	Booléen	SupprimerOeuvre	Void	Permet de supprimer les informations d'une œuvre

Classe : Film, classe décrivant les films				
Attributs				
Visibilité	Nom	Type	Description	
public	Duree_film	Numérique	La durée du film	
public	Format_film	Alphanumérique	Le format du film	
Méthodes				
Visibilité	Type	Nom	Paramètre	description
Public	Booléen	ajouterOeuvre	Void	Permet d'ajouter un film
Public	Void	VisualiserOeuvre	Void	Permet de visualiser un film
Public	Booléen	ModifierOeuvre	Void	Permet de modifier les informations d'un film
Public	Booléen	SupprimerOeuvre	Void	Permet de supprimer les informations d'un film

Classe : Production sonore, classe décrivant les productions sonores présentées lors des différents prix				
Attributs				
Visibilité	Nom	Type	Description	
public	Duree_Psonore	Numérique	La durée de la production	
public	Type_Psonore	Alphanumérique	Le type de la production	
Méthodes				
Visibilité	Type	Nom	Paramètre	description
Public	Booléen	ajouterPsonore	Void	Permet d'ajouter une production sonore
Public	Void	VisualisePsonore	Void	Permet de visualiser une production sonore
Public	Booléen	ModifierPsonore	Void	Permet de modifier les informations d'une production sonore
Public	Booléen	SupprimerPsonore	Void	Permet de supprimer les informations d'une production sonore

Classe : S.IAPRP, classe décrivant les structures des pays membre de l'IAPRP.				
Attributs				
Visibilité	Nom	Type	Description	
public	Code_Structure	Alphanumérique	Le code de la structure	
public	Nom_Structure	Alphanumérique	Le Nom de la structure	
Public	Statut_Structure	Alphanumérique	Le statut de la structure	
Public	telephone_Structure	Alphanumérique	Le numéro de téléphone de la structure	
Public	email_Structure	Alphanumérique	L'adresse électronique de la structure	
Public	adresse_Structure	Alphanumérique	L'adresse postale de la structure	
public	pays_Structure	Alphanumérique	Le pays de la structure	
Méthodes				
Visibilité	Type	Nom	Paramètre	description
Public	Booléen	ajouterStructure	Void	Permet d'ajouter une nouvelle structure
Public	Void	AfficherStructure	Void	Permet de visualiser la liste des structures enregistrées
Public	Void	envoyerMessage	Void	Permet d'envoyer un message à une structure enregistrée
Public	Booléen	modifierStructure	Void	Permet de modifier les informations d'une structure
Public	Booléen	supprimerStructure	Void	Permet de supprimer les informations d'une structure

Classe : Enseignant, classe décrivant les enseignants intervenant au CPFACE				
Attributs				
Visibilité	Nom	Type	Description	
public	Code_Enseignant	Alphanumérique	Le code de l'enseignant	
public	Nom_Enseignant	Alphanumérique	Le Nom de l'enseignant	
Public	Prenom_Enseignant	Alphanumérique	Le statut de l'enseignant	
Public	telephone_Enseignant	Alphanumérique	Le numéro de téléphone de l'enseignant	
Public	email_Enseignant	Alphanumérique	L'adresse électronique de l'enseignant	
Public	adresse_Enseignant	Alphanumérique	L'adresse postale de l'enseignant	
public	Nationalité_Enseignant	Alphanumérique	La nationalité de l'enseignant	
Méthodes				
Visibilité	Type	Nom	Paramètre	description
Public	Booléen	ajouterEnseignant	Void	Permet d'ajouter un nouvel enseignant
Public	Void	AfficherEnseignant	Void	Permet de visualiser la liste des enseignants enregistrés
Public	Void	envoyerMessage	Void	Permet d'envoyer un message à un enseignant
Public	Booléen	modifierEnseignant	Void	Permet de modifier les informations d'un enseignant
Public	Booléen	supprimerEnseignant	Void	Permet de supprimer les informations d'un enseignant

Classe : Courriel, classe décrivant les envois ou les réceptions de message				
Attributs				
Visibilité	Nom	Type	Description	
public	Titre	Alphanumérique	Le titre d'un courriel	
public	Contenu	Alphanumérique	Le contenu d'un courriel	
public	Date_Envoi	date	Date d'envoi du courriel	
public	Adresse_Dest	Alphanumérique	L'adresse du destinataire	
public	Adresse_Emet	Alphanumérique	L'adresse de l'émetteur	
Méthodes				
Visibilité	Type	Nom	Paramètre	description
Public	Booléen	Emettre	Void	Emettre un message
Public	Void	Envoyer	Void	Envoyer un message
Public	Booléen	Annuler	Void	Annuler l'envoi d'un message

Classe : Etudiant, classe décrivant les étudiants inscrits au CPFACE				
Attributs				
Visibilité	Nom	Type	Description	
public	Code_Etudiant	Alphanumérique	Le code de l'étudiant	
public	Nom_Etudiant	Alphanumérique	Le Nom de l'étudiant	
Public	Prenom_Etudiant	Alphanumérique	Le statut de l'étudiant	
Public	telephone_Etudiant	Alphanumérique	Le numéro de téléphone de l'étudiant	
Public	email_Etudiant	Alphanumérique	L'adresse électronique de l'étudiant	
Public	adresse_Etudiant	Alphanumérique	L'adresse postale de l'étudiant	
public	Nationalité_Etudiant	Alphanumérique	La nationalité de l'étudiant	
Méthodes				
Visibilité	Type	Nom	Paramètre	description
Public	Booléen	ajouterEtudiant	Void	Permet d'ajouter un nouvel étudiant
Public	Void	AfficherEtudiant	Void	Permet de visualiser la liste des étudiants enregistrés
Public	Void	envoyerMessage	Void	Permet d'envoyer un message à un étudiant
Public	Booléen	modifierInfosEtudiant	Void	Permet de modifier les informations d'un étudiant
Public	Booléen	supprimerEtudiant	Void	Permet de supprimer les informations d'un étudiant

Classe : Producteur, classe décrivant les producteurs des œuvres				
Attributs				
Visibilité	Nom	Type	Description	
public	Code_Producteur	Alphanumérique	Le code du producteur	
public	Nom_Producteur	Alphanumérique	Le Nom du producteur	
Public	Prenom_Producteur	Alphanumérique	Le statut du producteur	
Public	telephone_Producteur	Alphanumérique	Le numéro de téléphone du producteur	
Public	email_Producteur	Alphanumérique	L'adresse électronique du producteur	
Public	adresse_Producteur	Alphanumérique	L'adresse postale du producteur	
public	Nationalité_Producteur	Alphanumérique	La nationalité du producteur	
Méthodes				
Visibilité	Type	Nom	Paramètre	description
Public	Booléen	ajouter Producteur	Void	Permet d'ajouter un nouveau producteur
Public	Void	Afficher Producteur	Void	Permet de visualiser la liste des producteurs enregistrés
Public	Void	envoyerMessage	Void	Permet d'envoyer un message à un producteur
Public	Booléen	modifierInfosProducteur	Void	Permet de modifier les informations d'un producteur
Public	Booléen	supprimer Producteur	Void	Permet de supprimer les informations d'un producteur

Classe : Coproducteur, classe décrivant les coproducteurs des œuvres				
Attributs				
Visibilité	Nom	Type	Description	
public	Code_coproducteur	Alphanumérique	Le code du coproducteur	
public	Nom_coproducteur	Alphanumérique	Le Nom du coproducteur	
Public	Prenom_coproducteur	Alphanumérique	Le statut du coproducteur	
Public	telephone_coproducteur	Alphanumérique	Le numéro de téléphone du coproducteur	
Public	email_coproducteur	Alphanumérique	L'adresse électronique du coproducteur	
Public	adresse_coproducteur	Alphanumérique	L'adresse postale du coproducteur	
public	Nationalité_coproducteur	Alphanumérique	La nationalité du coproducteur	
Méthodes				
Visibilité	Type	Nom	Paramètre	description
Public	Booléen	ajouterCoproducteur	Void	Permet d'ajouter un nouveau coproducteur
Public	Void	AfficherCoproducteur	Void	Permet de visualiser la liste du coproducteur enregistré
Public	Void	envoyerMessage	Void	Permet d'envoyer un message à un coproducteur
Public	Booléen	modifierInfosCoproducteur	Void	Permet de modifier les informations d'un coproducteur
Public	Booléen	supprimer Producteur	Void	Permet de supprimer les informations d'un coproducteur

Classe : Formation, classe décrivant une formation continue				
Attributs				
Visibilité	Nom	Type	Description	
public	Code_Formation	Alphanumérique	Le code de la formation	
public	Thème	Alphanumérique	Thème de la formation	
Public	Cible	Alphanumérique	La population cible pour la formation	
Public	Date_Debut	Date	Date début de la formation	
Public	Date_Fin	Date	Date fin de la formation	
Méthodes				
Visibilité	Type	Nom	Paramètre	description
Public	Booléen	ajouterFormation	Void	Permet d'ajouter un nouveau module de formation
Public	Void	AfficherFormation	Void	Permet de visualiser la liste des formations enregistrées
Public	Booléen	modifierInfosFormation	Void	Permet de modifier les informations d'une formation
Public	Booléen	AnnulerFormation	Void	Permet d'annuler ne formation

Classe : Frais_Formation				
Attributs				
Visibilité	Nom	Type	Description	
public	Code_Frais	Numérique	Le code du frais de la formation	
public	Montant	Alphanumérique	Le montant pour la formation	

Classe : Programme_Formation				
Attributs				
Visibilité	Nom	Type	Description	
public	Code_Programme	Numérique	Le code du programme de formation	
public	Contenu	Alphanumérique	Le contenu d'une formation	

Classe : Condition_Usage_Cession			
Attributs			
Visibilité	Nom	Type	Description
public	description	Alphanumérique	Description des conditions d'usage et de cession faites par un producteur sur son œuvre

Classe : Quote-part			
Attributs			
Visibilité	Nom	Type	Description
public	montant	Numérique	Quote-part versé par chaque coproducteur

Classe : Collection			
Attributs			
Visibilité	Nom	Type	Description
public	Code_Collection	Alphanumérique	Code d'une collection d'œuvre
	date	date	Date de la collection
public	description	Alphanumérique	Description des conditions d'usage et de cession faites par un producteur sur son œuvre

Classe : Fiche_Emprunt			
Attributs			
Visibilité	Nom	Type	Description
public	Date_Emprunt	date	Date d'emprunt de l'œuvre
public	Date_Retour	date	Date de retour de l'œuvre empruntée

Classe : Condition_DEntree			
Attributs			
Visibilité	Nom	Type	Description
public	description	Alphanumérique	Description des conditions d'entrée au CPFACE

Classe : Utilisateur, classe décrivant les utilisateurs du système				
Attributs				
Visibilité	Nom		Type	Description
public	Nom_Utilisateur		Alphanumérique	Le nom de l'utilisateur
public	MotDePasse		Alphanumérique	Le mot de passe d'utilisateur
public	NiveauSecurite		Numérique	Niveau de sécurité pour la connexion
Méthodes				
Visibilité	Type	Nom	Paramètre	Description
Public	Booléen	ajouterUtilisateur	Void	Permet d'ajouter un nouvel utilisateur
Public	Void	listerUtilisateur	Void	Permet de voir la liste des utilisateurs
Public	Booléen	ModifierInfosUtilisateur	Void	Permet de modifier les informations d'un utilisateur
Public	Booléen	SupprimerUtilisateur	Void	Permet de supprimer les informations d'un utilisateur
public	Void	Connecter	Void	Permet à l'utilisateur de se connecter au système
public	Void	Deconnecter	Void	Permet à l'utilisateur de se déconnecter au système

Classe : Attestation			
Attributs			
Visibilité	Nom	Type	Description
public	Date_Etablis	date	Date d'établissement de l'attestation
public	description	Alphanumérique	Description de l'attestation

Classe : formateur, classe décrivant les formateurs intervenant dans la formation continue au CPFACE				
Attributs				
Visibilité	Nom	Type	Description	
public	Code_Formateur	Alphanumérique	Le code du formateur	
public	Nom_Formateur	Alphanumérique	Le Nom du formateur	
Public	Prenom_Formateur	Alphanumérique	Le statut du formateur	
Public	telephone_Formateur	Alphanumérique	Le numéro de téléphone du formateur	
Public	email_Formateur	Alphanumérique	L'adresse électronique du formateur	
Public	adresse_Formateur	Alphanumérique	L'adresse postale du formateur	
public	Nationalité_Formateur	Alphanumérique	La nationalité du formateur	
Méthodes				
Visibilité	Type	Nom	Paramètre	description
Public	Booléen	ajouterFormateur	Void	Permet d'ajouter un nouveau formateur
Public	Void	AfficherFormateur	Void	Permet de visualiser la liste des formateurs enregistrés
Public	Booléen	modifierInfosFormateur	Void	Permet de modifier les informations d'un formateur
Public	Booléen	supprimerFormateur	Void	Permet de supprimer les informations d'un formateur

Classe : PatrimoineTech, classe décrivant le patrimoine technique de la production d'une œuvre			
Attributs			
Visibilité	Nom	Type	Description
public	description	Alphanumérique	Description de la patrimoine technique

Classe : Agent, classe décrivant les agents suivant une formation continue				
Attributs				
Visibilité	Nom	Type	Description	
public	Code_Agent	Alphanumérique	Le code d'un agent	
public	Nom_Agent	Alphanumérique	Le nom d'un agent	
Public	Prenom_Agent	Alphanumérique	Le prénom d'un agent	
Méthodes				
Visibilité	Type	Nom	Paramètre	description
Public	Booléen	ajouterAgent	Void	Permet d'ajouter un nouvel agent
Public	Void	ListerAgent	Void	Permet de visualiser la liste des agents ayant pris part à la formation
Public	Booléen	supprimerAgent	Void	Permet de supprimer un agent

Classe : EquipeProduction, classe décrivant l'équipe de production d'une œuvre				
Attributs				
Visibilité	Nom	Type	Description	
public	Code_Equipe	Alphanumérique	Le code d'une équipe de production	
public	Nb_Equipe	Numérique	Le nombre de personne composant l'équipe	
Méthodes				
Visibilité	Type	Nom	Paramètre	description
Public	Booléen	MiseEnPlaceEquipe	Void	Mise en place d'une équipe de production
Public	Booléen	supprimerEquipe	Void	Permet de supprimer une équipe de production

Classe : Internaute, classe décrivant les internautes du système				
Attributs				
Visibilité	Nom	Type	Description	
public	Nom_Internaute	Alphanumérique	Le Nom de l'internaute	
Public	Prenom_Internaute	Alphanumérique	Le statut de l'internaute	
Public	telephone_Internaute	Alphanumérique	Le numéro de téléphone de l'internaute	
Public	email_Internaute	Alphanumérique	L'adresse électronique de l'internaute	
Public	adresse_Internaute	Alphanumérique	L'adresse postale de l'internaute	
Méthodes				
Visibilité	Type	Nom	Paramètre	description
Public	Booléen	Consulter	Void	Permet de consulter certaines fonctionnalités du système
Public	Void	envoyerMessage	Void	Permet d'envoyer un message à un acteur du système

Classe : Article			
Attributs			
Visibilité	Nom	Type	Description
public	Code_Arti	Alphanumérique	Code d'un article
public	Description_Arti	Alphanumérique	Description d'un article

Classe : ReglementInterieur			
Attributs			
Visibilité	Nom	Type	Description
public	Code_Regl	Alphanumérique	Code d'un règlement intérieur
public	Date_Regl	Date	Date de création du règlement intérieur

Classe : Comite_Organisation			
Attributs			
Visibilité	Nom	Type	Description
public	Code_Org	Alphanumérique	Code du comité d'organisation
public	Nb_Mbre	Numérique	Nombre de personne dans d'un comité

Classe : Recommandation				
Attributs				
Visibilité	Nom	Type	Description	
public	Code_Recom	Alphanumérique	Code des recommandations	
public	Contenu_Recom	Alphanumérique	Contenu d'une recommandation	
Méthodes				
Visibilité	Type	Nom	Paramètres	Description
Public	void	Editer	void	Editer les recommandations

Classe : Laureat, classe décrivant les lauréats d'un prix				
Attributs				
Visibilité	Nom	Type	Description	
public	Code_Laureat	Alphanumérique	Le code du lauréat	
public	Nom_Laureat	Alphanumérique	Le nom du lauréat	
Public	Prenom_Laureat	Alphanumérique	Le prénom du lauréat	
Public	telephone_Laureat	Alphanumérique	Le numéro de téléphone du lauréat	
Public	email_Laureat	Alphanumérique	L'adresse électronique du lauréat	
Public	adresse_Laureat	Alphanumérique	L'adresse postale du lauréat	
public	Nationalité_Laureat	Alphanumérique	La nationalité du lauréat	
Méthodes				
Visibilité	Type	Nom	Paramètre	description
Public	Booléen	ajouterLaureat	Void	Permet d'ajouter un nouveau lauréat
Public	Void	ListerDesLaureat	Void	Permet de visualiser la liste des lauréats d'un prix donné

Classe : Concours				
Attributs				
Visibilité	Nom	Type	Description	
public	Code_Concours	Alphanumérique	Identifiant d'un concours	
public	Theme_Concours	Alphanumérique	Thème du concours	
public	Categorie_Concours	Alphanumérique	Catégorie concernée par le concours	
Méthodes				
Visibilité	Type	Nom	Paramètres	Description
Public	Void	Editer	Void	Editer un concours

Classe : Prix				
Attributs				
Visibilité	Nom	Type	Description	
public	Code_Prix	Alphanumérique	Identifiant d'un prix	
public	Valeur_Prix	Alphanumérique	La valeur d'un prix	
public	Distinction_Prix	Alphanumérique	Distinction du prix	
Méthodes				
Visibilité	Type	Nom	Paramètres	Description
Public	Void	EditerPrix	Void	Editer un prix

Classe : Prix_SST représente la classe du prix santé et sécurité au travail du FESPACO				
Attributs				
Visibilité	Nom	Type	Description	
public	Code_PrixSST	Alphanumérique	Identifiant d'un prix_SST	
public	Valeur_Prix	Alphanumérique	La valeur du prix_SST	
public	Theme_PrixSST	Alphanumérique	Le thème du prix	
Méthodes				
Visibilité	Type	Nom	Paramètres	Description
Public	Void	EditerPrixSST	Void	Editer un prix_SST
Public	Void	ConsulterInfosPrixSST	Void	Consultation des informations relatives à un prix SST

Classe : Edition_Syapro				
Attributs				
Visibilité	Nom	Type	Description	
public	Code_Edition	Alphanumérique	Identifiant d'une édition SYAPRO	
public	Theme_Edition	Alphanumérique	Thème d'une édition	
Méthodes				
Visibilité	Type	Nom	Paramètres	Description
Public	Void	EditerEdition	Void	Editer une édition SYAPRO
Public	Void	ConsulterInfosSYAPRO	Void	Consultation des informations relatives à une édition du SYAPRO

Classe : Membre_Comite				
Attributs				
Visibilité	Nom	Type	Description	
public	Nom_Mbre	Alphanumérique	Nom d'un membre du comité scientifique	
public	Prenom_Mbre	Alphanumérique	Prénom d'un membre du comité scientifique	
public	Titre_Mbre	Alphanumérique	titre d'un membre du comité scientifique	

Classe : Conferencier, classe décrivant les conférenciers prenant part à une édition du SYAPRO				
Attributs				
Visibilité	Nom	Type	Description	
public	Titre_Conf	Alphanumérique	Le titre du conférencier	
public	Nom_Conf	Alphanumérique	Le nom du conférencier	
Public	Prenom_Conf	Alphanumérique	Le prénom du conférencier	
Public	telephone_Conf	Alphanumérique	Le numéro de téléphone du conférencier	
Public	email_Conf	Alphanumérique	L'adresse électronique du conférencier	
Public	adresse_Conf	Alphanumérique	L'adresse postale du conférencier	
public	Nationalité_Conf	Alphanumérique	La nationalité du conférencier	
Méthodes				
Visibilité	Type	Nom	Paramètre	description
Public	Booléen	ajouterConf	Void	Permet d'ajouter un nouveau conférencier
Public	Void	ListerDesConf	Void	Permet de voir la liste des conférenciers d'une édition

Classe : Budget				
Attributs				
Visibilité	Nom	Type	Description	
public	Code_Budget	Alphanumérique	Le code d'un budget alloué à une édition	
public	Valeur_Budget	Alphanumérique	La valeur du budget	

Classe : Invite				
Attributs				
Visibilité	Nom	Type	Description	
public	Code_invite	Alphanumérique	Identifiant d'un invité	
public	Adresse_invite	Alphanumérique	Adresse d'un invité	
public	Pays_invite	Alphanumérique	Le pays de l'invité	

Classe : Periode			
Attributs			
Visibilité	Nom	Type	Description
public	Debut	Date	Date début
public	Fin	Date	Date fin

Classe : Structure			
Attributs			
Visibilité	Nom	Type	Description
public	Nom_Struct	Alphanumérique	Nom de la structure invitée

Classe : Personnalite			
Attributs			
Visibilité	Nom	Type	Description
public	Nom_Pers	Alphanumérique	Nom de la personnalité
public	Prenom_Pers	Alphanumérique	Prénom de la personnalité
public	Qualification	Alphanumérique	Qualification de la personnalité invitée

Classe : Section			
Attributs			
Visibilité	Nom	Type	Description
public	Code_Section	Alphanumérique	Identifiant d'une section de formation
public	Libelle_Section	Alphanumérique	Le nom d'une section de formation

Classe : Rapport				
Attributs				
Visibilité	Nom	Type	Description	
public	Code_Rapport	Alphanumérique	L'identifiant du rapport	
public	Titre_Rapport	Alphanumérique	Le titre du rapport	
Public	DateEdition	Date	Date d'édition du rapport	
Public	Description	Alphanumérique	Description du contenu du rapport	
Public	Nom_Fichier	Alphanumérique	Le nom du fichier contenant le rapport	
Méthodes				
Visibilité	Type	Nom	Paramètre	description
Public	Booléen	ajouterRapport	Void	Permet d'ajouter un nouveau rapport
Public	Void	ConsulterRapport	Void	Consultation du contenu d'un rapport
Public	Void	SupprimerRapport	Void	Supprimer un rapport

Classe : Note				
Attributs				
Visibilité	Nom	Type	Description	
public	Code_Note	Alphanumérique	code de la note	
public	Valeur_Note	Numérique	La note eu par un étudiant	
Public	Date	Date	Date de composition	
Public	Appreciation	Alphanumérique	Description de la note	
Méthodes				
Visibilité	Type	Nom	Paramètre	description
Public	Booléen	ajouterNote	Void	Permet d'ajouter une nouvelle note
Public	Void	ConsulterNote	Void	Consultation les notes
Public	Void	SupprimerNote	Void	Supprimer une note
Public	Void	ModifierNote	Void	modifier une note d'un étudiant

Classe : Emploi_Temps				
Attributs				
Visibilité	Nom	Type	Description	
public	Nom_FichierEmploi	Alphanumérique	Nom du fichier de l'emploi de temps	
Méthodes				
Visibilité	Type	Nom	Paramètre	description
Public	Booléen	ajouterEmploiTemps	Void	Permet d'ajouter un nouvel emploi de temps
Public	Void	ConsulterEmploitemps	Void	Permet de consulter l'emploi de temps

Classe : Module				
Attributs				
Visibilité	Nom	Type	Description	
public	Code_Module	Alphanumérique	Code d'un module de cours	
public	Libelle_Module	Alphanumérique	Le nom du module de cours	

Classe : Matiere				
Attributs				
Visibilité	Nom	Type	Description	
public	Code_Matiere	Alphanumérique	Code d'une matière de formation	
public	Nom_Matiere	Alphanumérique	Nom de la matière	
public	VolumeHoraire_Mat	Alphanumérique	Le volume horaire de la matière	

Classe : Jury				
Attributs				
Visibilité	Nom	Type	Description	
public	Code_Jury	Alphanumérique	Code du jury	

Classe : Membre_Jury				
Attributs				
Visibilité	Nom	Type	Description	
public	Nom	Alphanumérique	Nom	
public	Prenom	Alphanumérique	Prenom	
public	Role	Alphanumérique	Rôle	

Classe : Technicien			
Attributs			
Visibilité	Nom	Type	Description
public	Nom_Tech	Alphanumérique	Nom du technicien
public	Prenom_Tech	Alphanumérique	Prenom du technicien
public	Role_Tech	Alphanumérique	Rôle du technicien

4.3.2 Représentation des diagrammes d'activités¹⁷

Diagramme d'activités1 : CU authentification

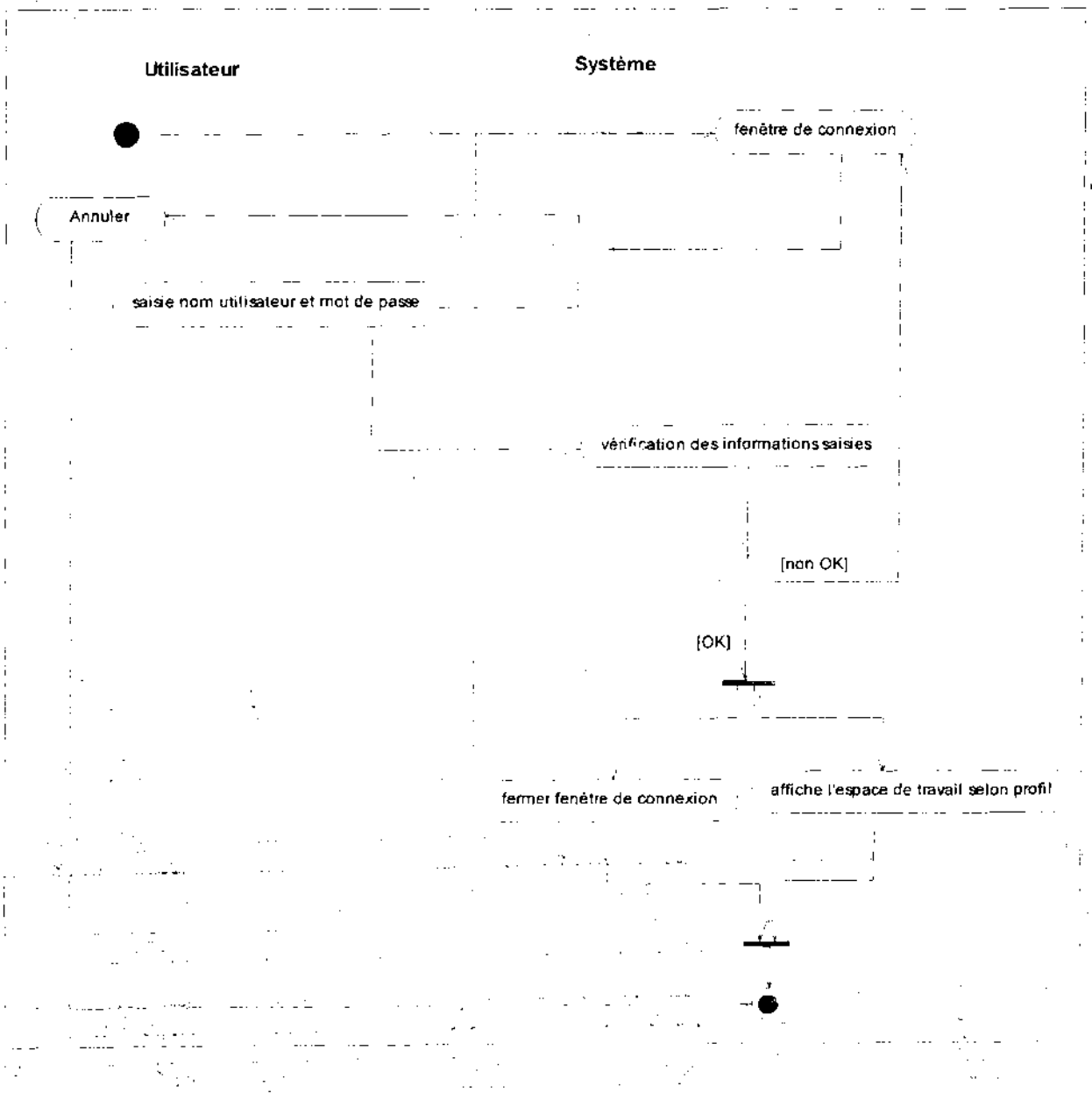


Diagramme 4.27 Diagramme d'activité décrivant la procédure d'authentification

¹⁷ Les concepts et les formalismes décrits en Annexes partie 5

❖ Diagramme d'activités du CU s'inscrire en ligne

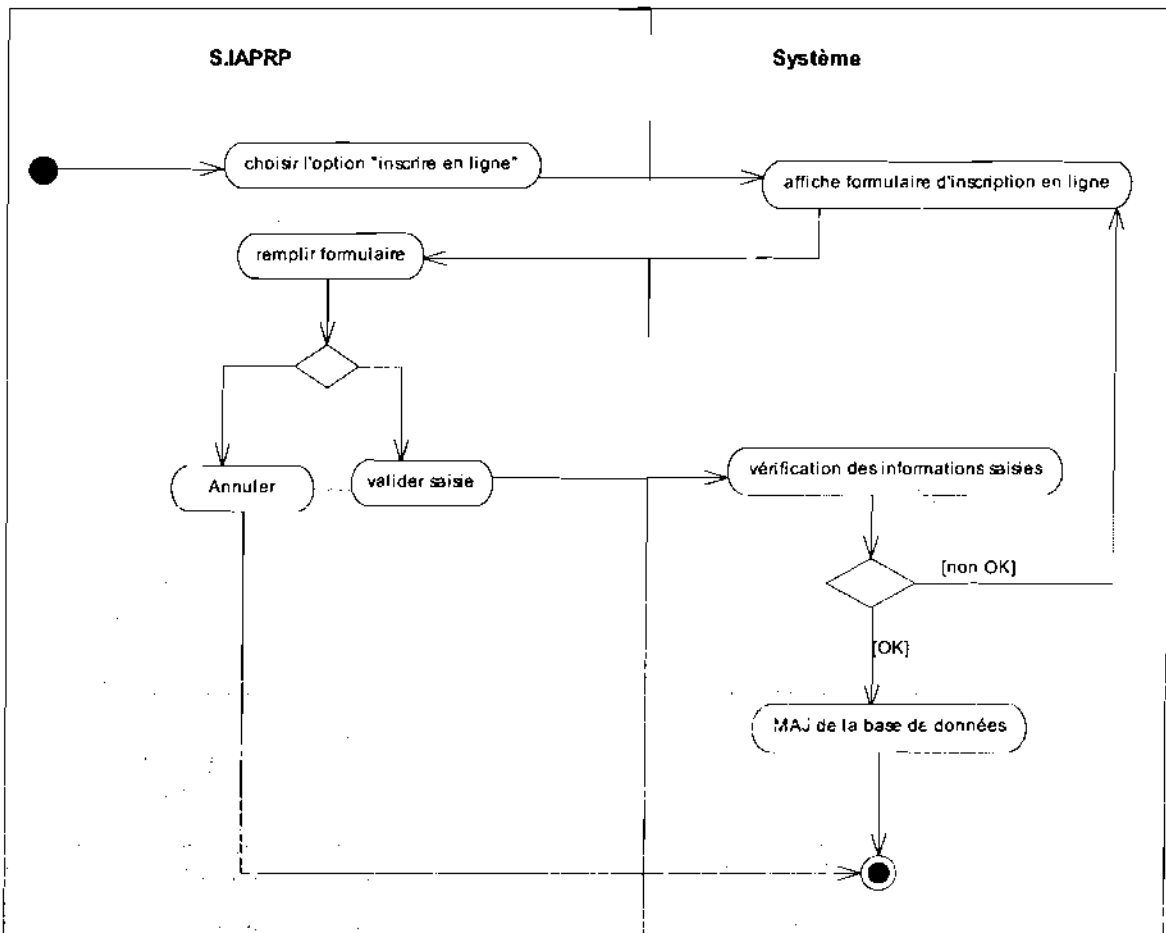


Diagramme4.28 Diagramme d'activité décrivant la procédure d'inscription en ligne

❖ Diagramme d'activité du CU envoyer message

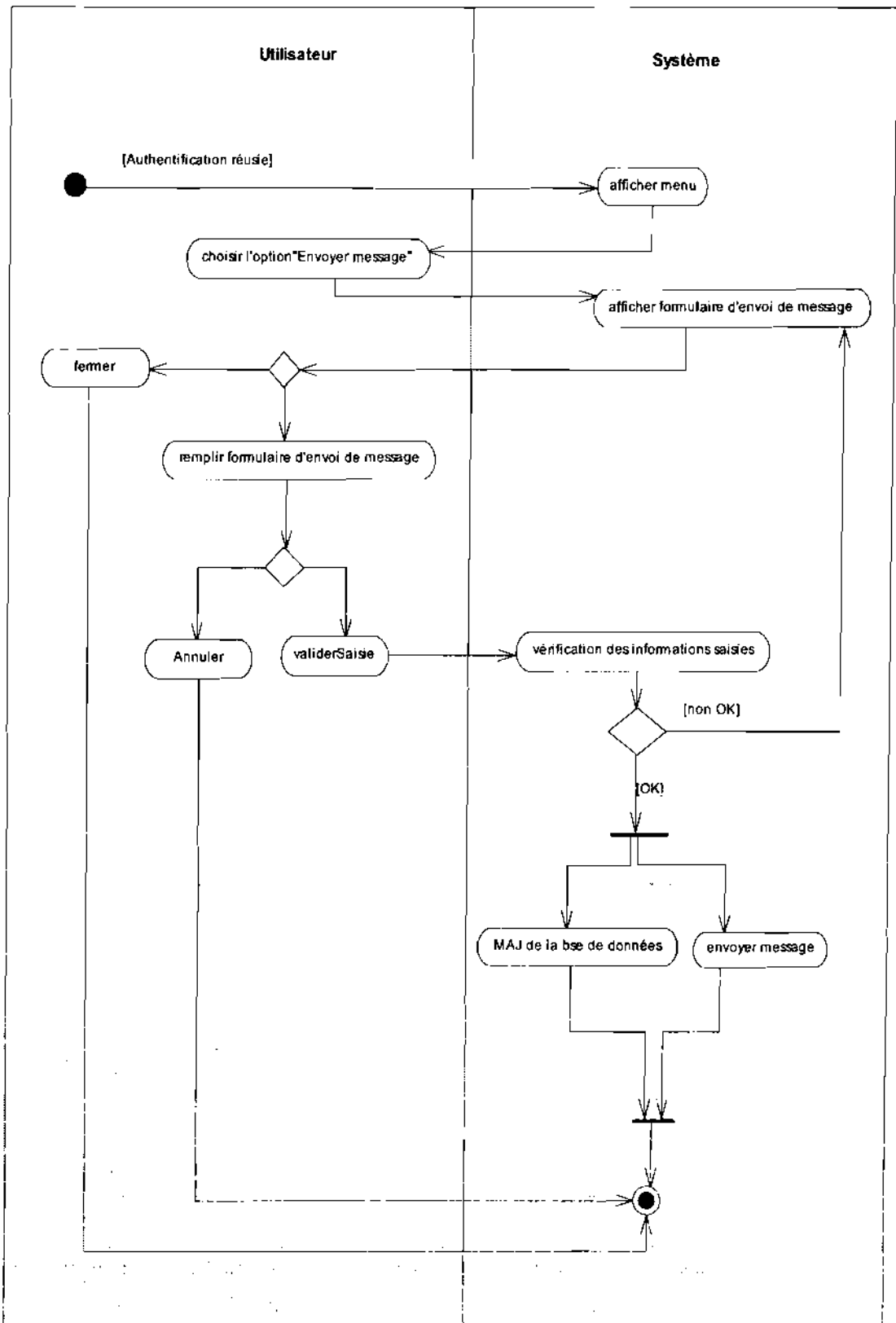


Diagramme4.29 Diagramme d'activité décrivant la procédure d'envoi de message

❖ Diagramme d'activité du CU gérer la médiathèque

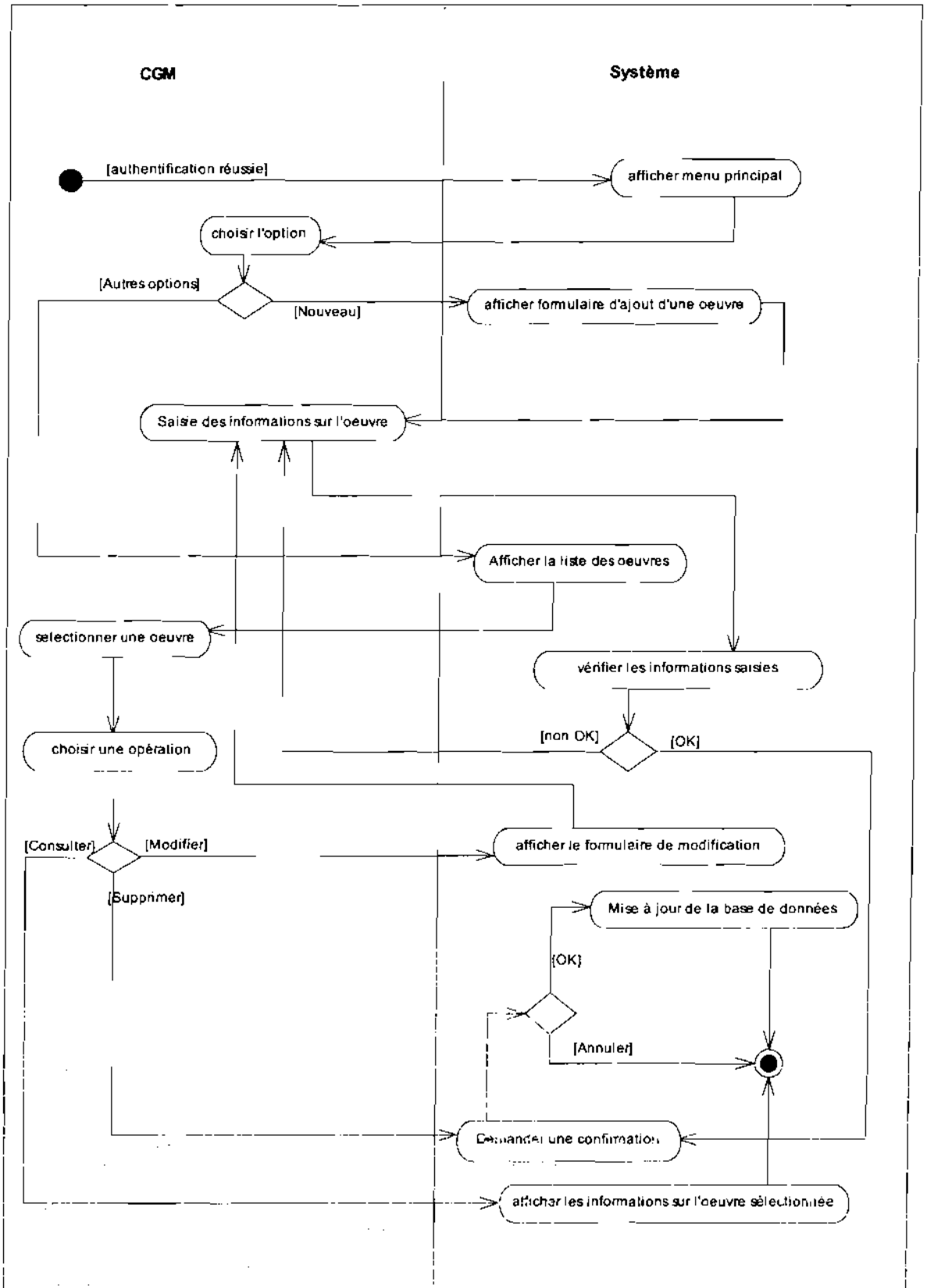


Diagramme4.30 Diagramme d'activité décrivant la procédure de gestion de la médiathèque

❖ Diagramme d'activité du CU Emprunter une œuvre

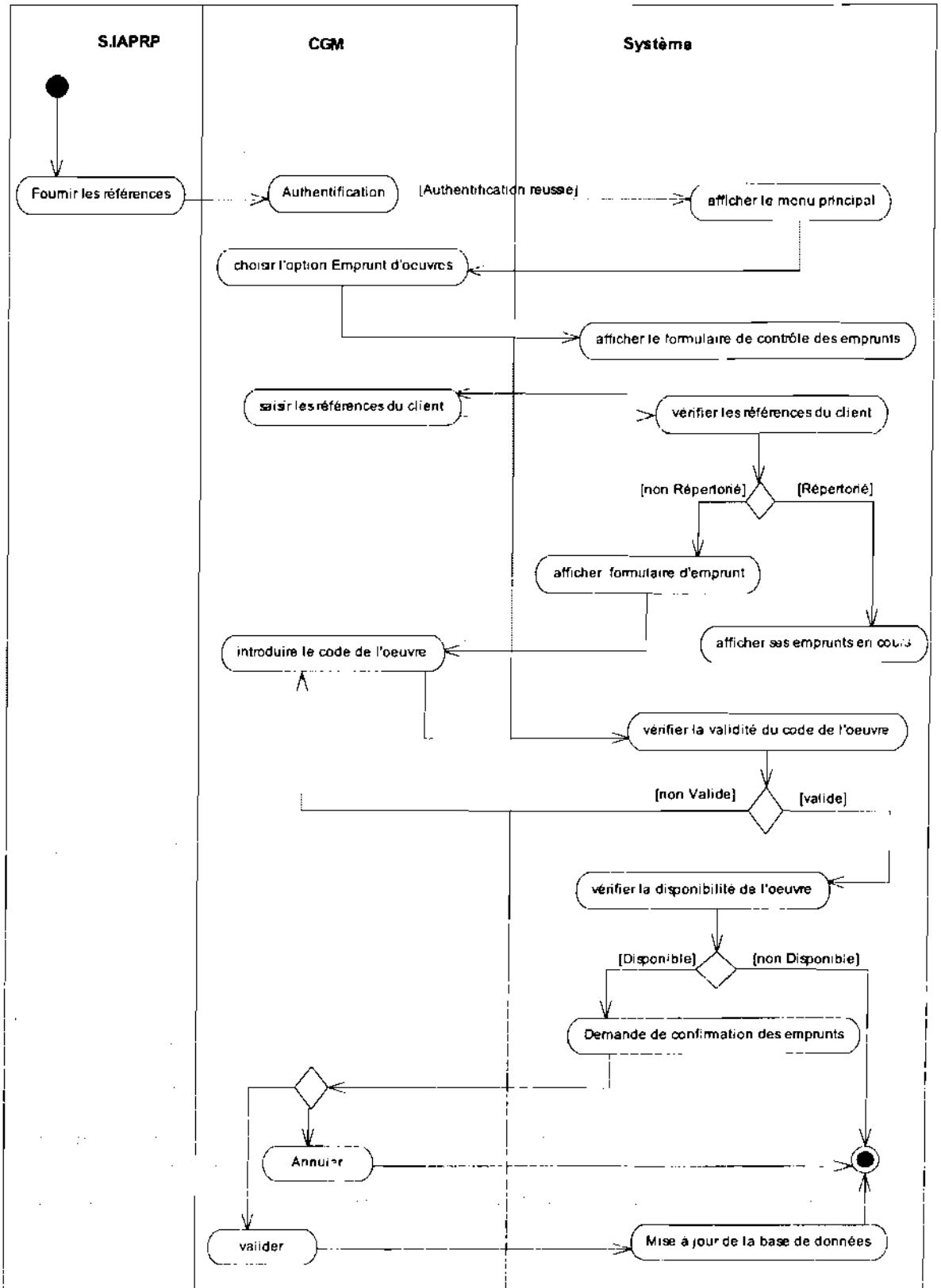


Diagramme4.31 Diagramme d'activité décrivant la procédure d'emprunt des œuvres

❖ Diagramme d'activité du CU Gérer les archives du SYAPRO

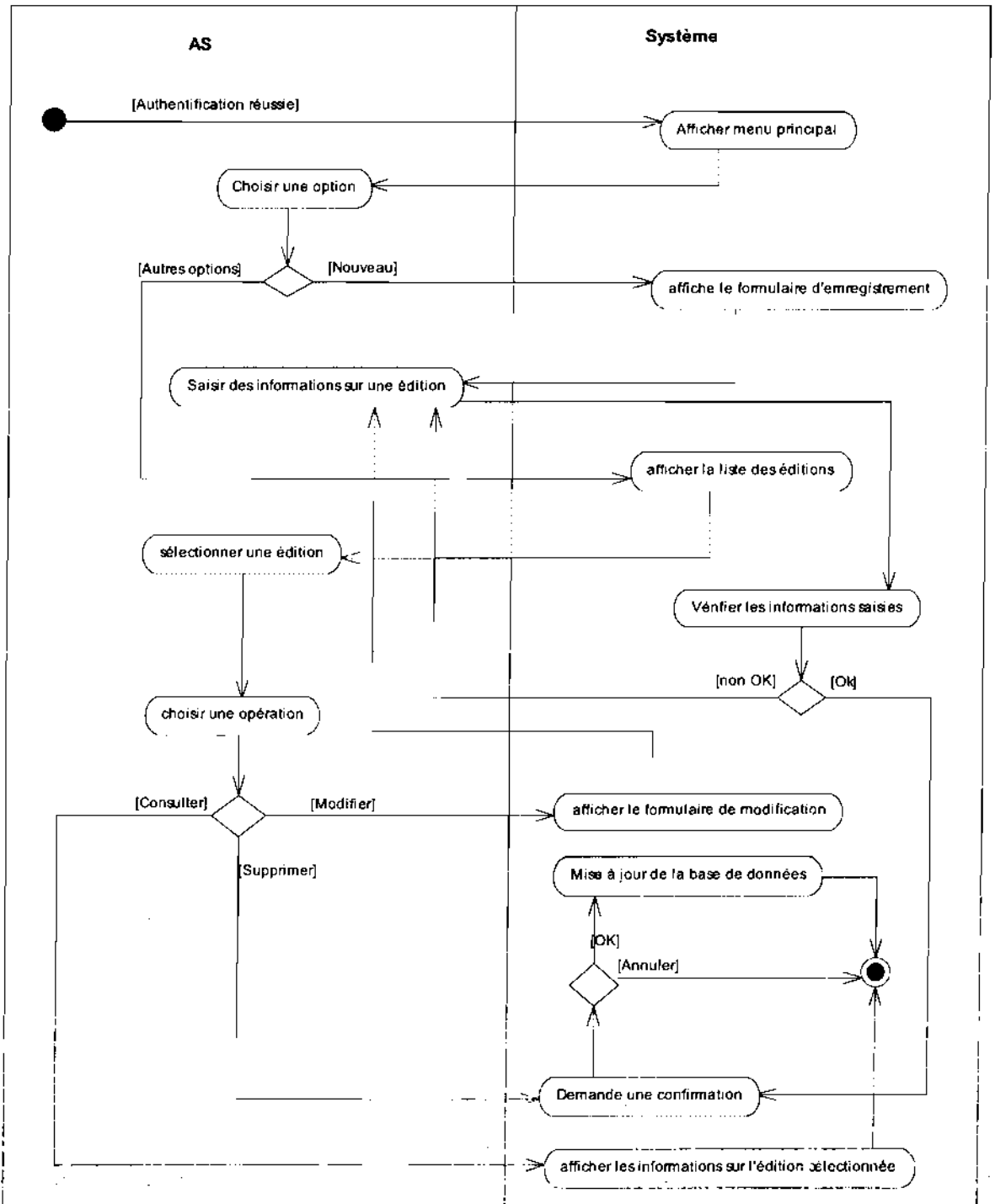


Diagramme4.32 Diagramme d'activité décrivant la procédure de gestion des archives du SYAPRO

❖ Diagramme d'activité du CU Gérer des archives du prix «santé et sécurité au travail » du FESPACO

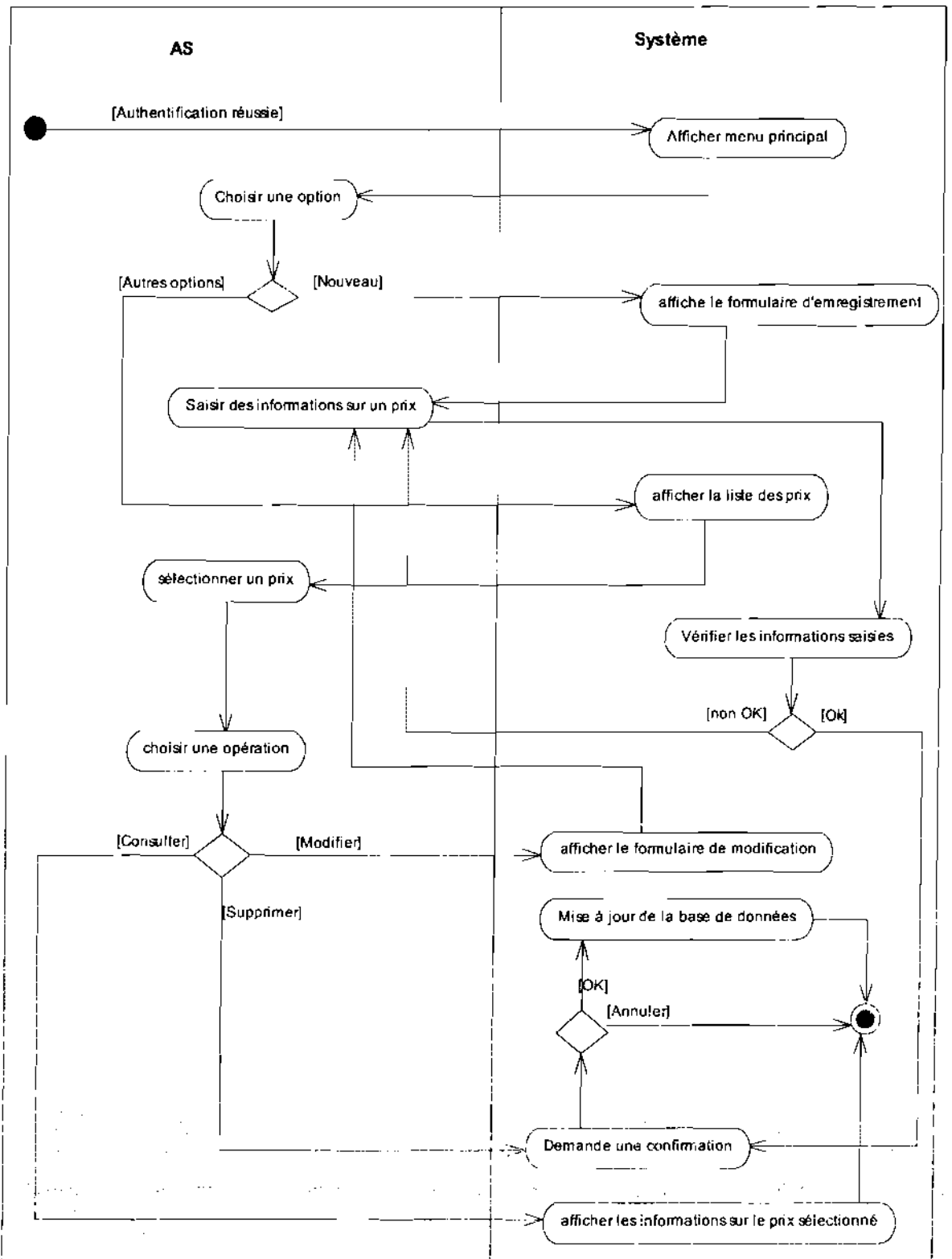


Diagramme4.33 Diagramme d'activité décrivant la procédure de gestion des archives du prix «santé et sécurité au travail » du FESPACO

❖ Diagramme d'activité du CU Gérer la formation

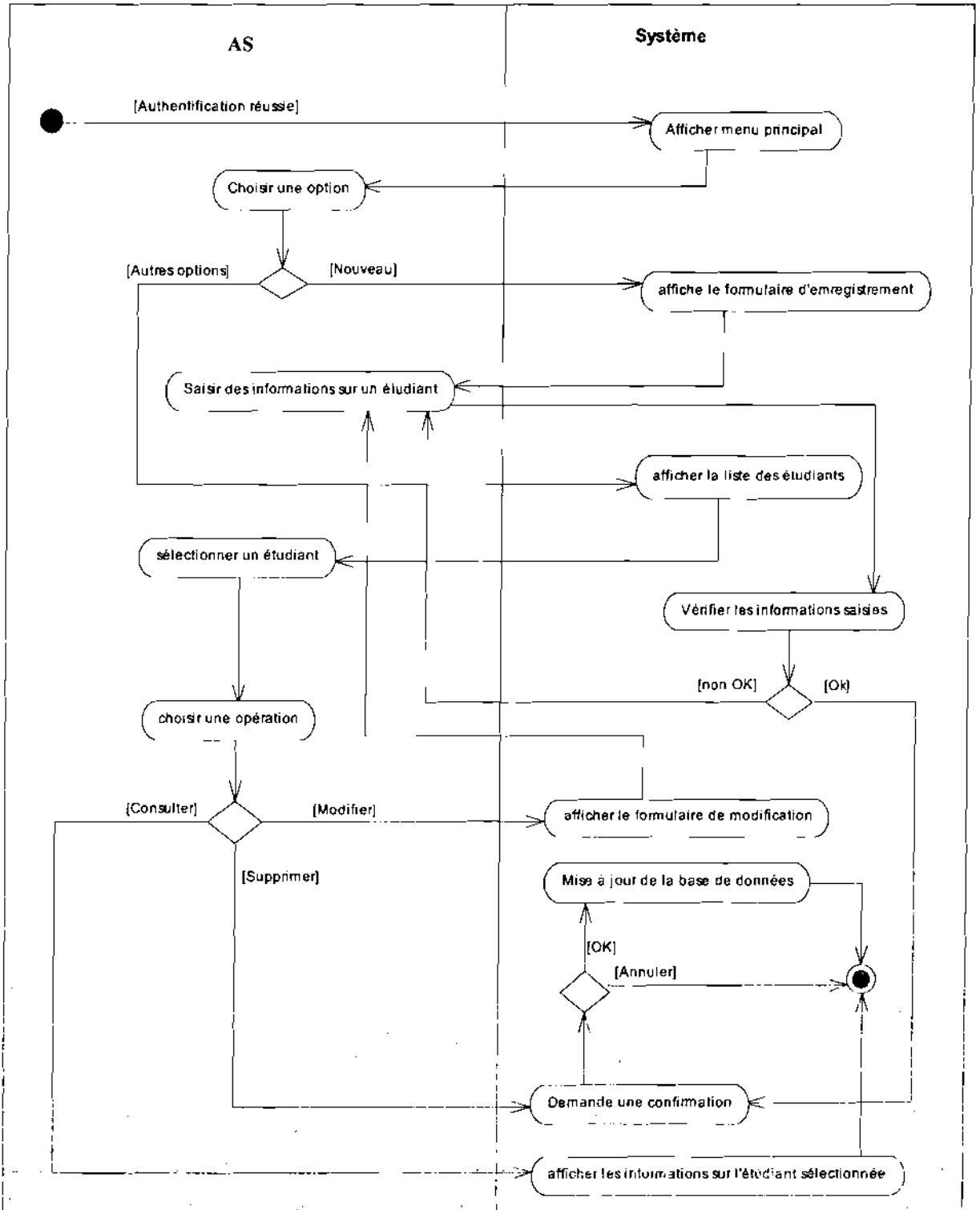


Diagramme4.34 Diagramme d'activité décrivant la procédure de gestion des formations

❖ Diagramme d'activité du CU Gérer les partenaires

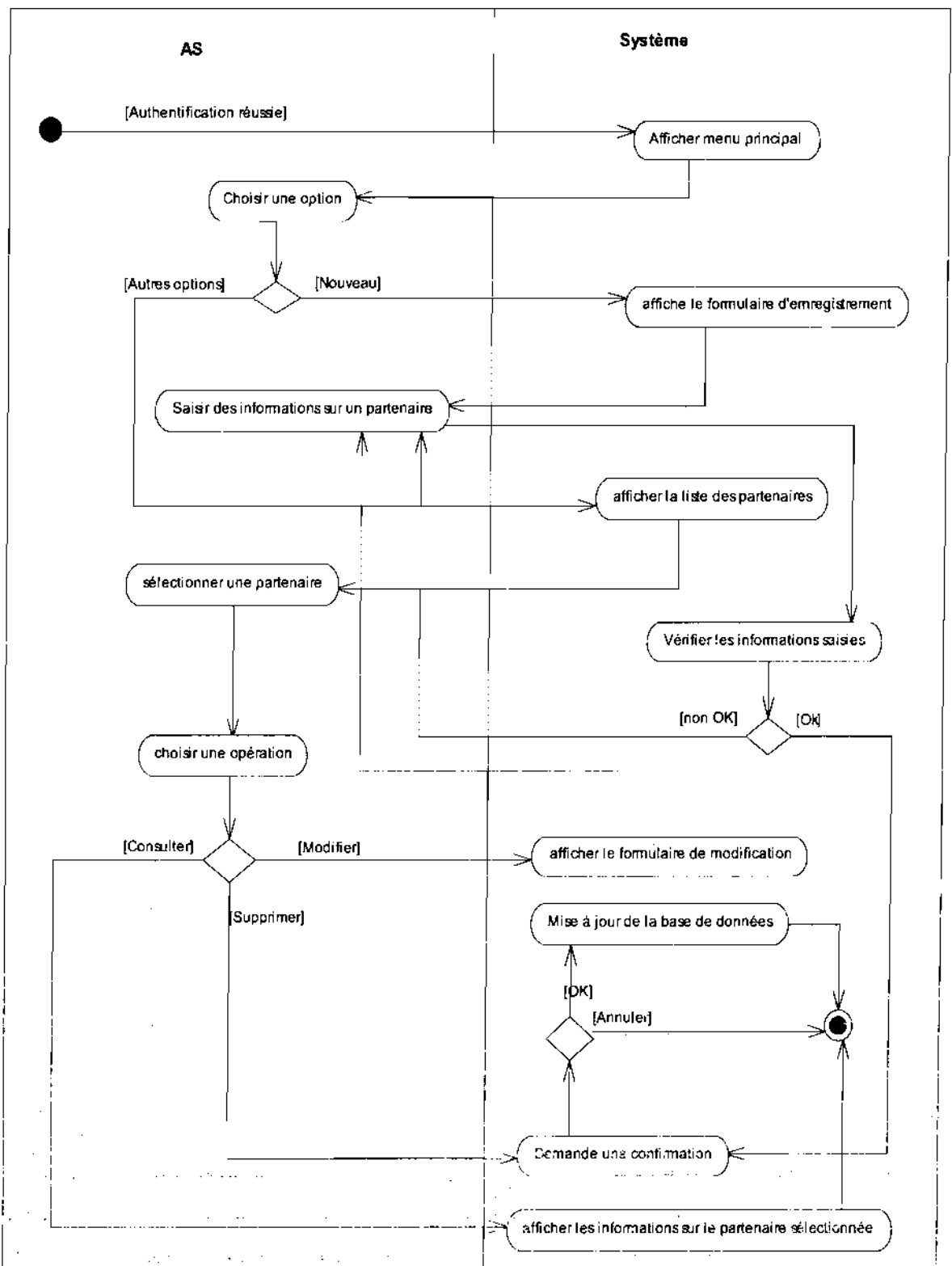


Diagramme4.35 Diagramme d'activité décrivant la procédure de gestion des partenaires

4.3.3 -Représentation de diagrammes de déploiement¹⁸

Le diagramme de déploiement montre la disposition physique des matériels et la répartition des composants sur ces matériels. Une ressource matérielle est représentée par un nœud.

Les nœuds sont connectés entre eux à l'aide d'un support de communication, la nature des lignes de communication et leurs caractéristiques peuvent être précisées.

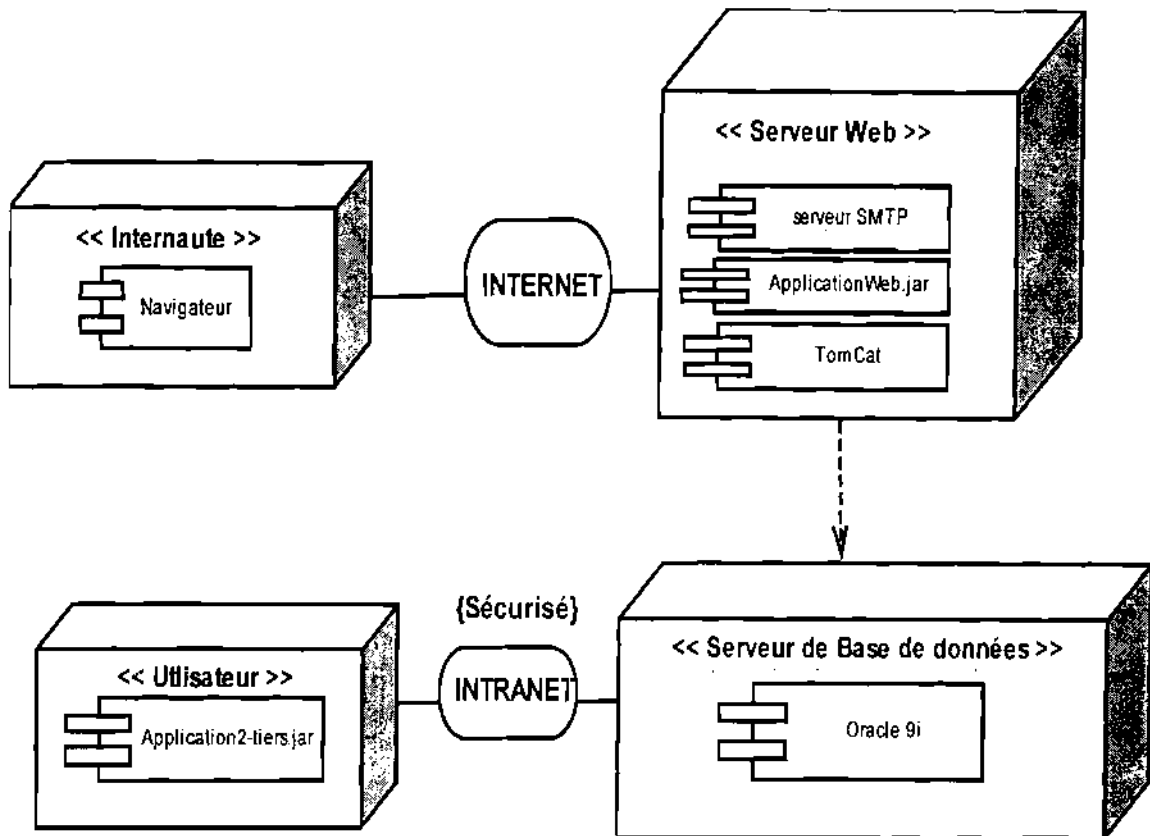


Diagramme 4.36 diagramme de déploiement du futur système

4.4-Procédures transitoire

Les procédures transitoires sont des tâches à exécuter pour passer du système actuel au système futur. Pour cela le système futur devra être soumis à une série de tests afin de s'assurer de son adéquation avec les besoins et les exigences exprimés par les utilisateurs. Les éventuelles défaillances décelées au cours de ces tests seront progressivement corrigées jusqu'à l'obtention d'une application conforme aux besoins. En effet l'application sera mise en service pour une période transitoire de quatre (4) mois pour s'assurer qu'il répond entièrement aux besoins exprimés.

4.5-Politique de sécurité

C'est une déclaration formelle des règles auxquelles doivent se conformer les personnes recevant un droit d'accès au capital technologique et informatique d'une entreprise. L'avantage de définir une politique de sécurité est :

¹⁸ Les concepts et le formalisme sont décrits en Annexes partie 6

- De situer la responsabilité du personnel ;
- De constituer une base pour tout acte juridique.

L'objectif de la sécurité des systèmes d'information est :

- d'empêcher des personnes non-autorisées d'agir sur le système ;
- d'empêcher les utilisateurs d'effectuer des opérations involontaires capable de nuire le système ;
- de sécuriser les données en prévoyant les pannes ;
- de garantir la non-interruption du système.

4.5.1-Protection contre les catastrophes

Les catastrophes susceptibles d'endommager les systèmes d'informatiques sont essentiellement les incendies, les inondations, les effondrements de bâtiments. Pour se prémunir des désastres sus cités, nous prévoyons des sauvegarde journalière de la base de donnée sur une bande magnétique ou sur CDRW (Compact Disk ReWriable).

Ces supports de stockages seront conservés dans des coffres ignifuges et les placés dans des bâtiments différents. Les données devront être restaurées après une catastrophe.

4.5.2-Protection contre les virus

Un virus un programme capable de provoquer la destruction des données et de réduire la fiabilité des résultats produits par le système. La protection consiste à installer un anti-virus éprouvé (pour avoir des versions révisées à moindre cout) au niveau des différents postes de travail.

En outre une restriction quant à l'utilisation de tout support externe (CD-RM, clé USB, disquette,...) d'origine douteuse pourrait renforcer la sécurité.

L'application de ces mesures passe une politique de sensibilisation des acteurs quant aux méfaits engendrés par les virus.

4.5.3-Protection contre les coupures d'électricité

Pour garantir la non-interruption du système nous préconisons l'utilisation d'onduleurs et d'un groupe électrogène en cas coupure prolongée.

4.5.4- Confidentialité des données

La confidentialité est l'assurance que seules les personnes autorisées ont accès à l'information. En effet, elle sera assurée par la définition d'un droit d'accès. Ainsi donc, les utilisateurs seront identifiés par un nom de connexion et authentifiés par un mot de passe (susceptible d'être modifié régulièrement). De cette façon, l'accès à la base de données sera restreint aux personnes qui seront autorisées tout en contrôlant les opérations qu'ils sont en mesure d'effectuer sur les informations la base de donnée.

Un journal d'historique permettra de retracer les différentes connexions (dates et heures) afin d'identifier les responsables de toute opération.

4.6-Procédure de secours

4.6.1- Poste de travail indisponible

Nous préconisons de faire appel a un service de maintenance en cas de panne d'ordinateur. Au cas où le poste est irréparable, l'utilisateur du poste en panne, avec le consentement des autres utilisateurs, effectuera sur leurs postes les traitements qui lui sont assignés avant l'acquisition d'un nouveau poste.

4.6.2- Panne de serveur

En cas de panne de serveur, nous préconisons de déplacer l'un de ses disques durs vers un autre poste de travail (on indique que chaque serveur sera équipé de 2 disques durs dont l'un est le miroir de l'autre) afin de transformer ce poste en serveur temporaire.

4.6.3- Indisponibilité généralisée du système

En cas de panne généralisée du système, nous suggérons de recourir à l'ancien système. En somme, les traitements se feront manuellement pendant la durée d'indisponibilité du système.

Chapitre 5 Réalisation

La construction est le moment où l'on construit le produit. L'architecture de référence se métamorphose en produit complet. Le produit contient tous les cas d'utilisation que le groupe de projet, en accord avec les utilisateurs ont décidé de mettre au point pour cette version.

Dans ce chapitre nous ferons une description des outils de développement utilisés ainsi qu'une présentation de quelques maquettes graphiques de l'application 3-tiers.

5.1-Infrastructure applicative

5.1.1 Outil d'analyse : Sybase Power AMC

PowerAMC permet de modéliser les processus métiers, les données et les objets à l'aide des modèles suivants :

- Modèle conceptuel de données
- Modèle physique de données
- Modèle de traitement Merise
- Modèle Orienté-Objet
- Modèle de processus métiers
- Révérenciel.

Tous les modèles PowerAMC sont documentés dans différents guides de l'utilisateur et didacticiels. Une introduction explique le contenu et la structure de la documentation PowerAMC.

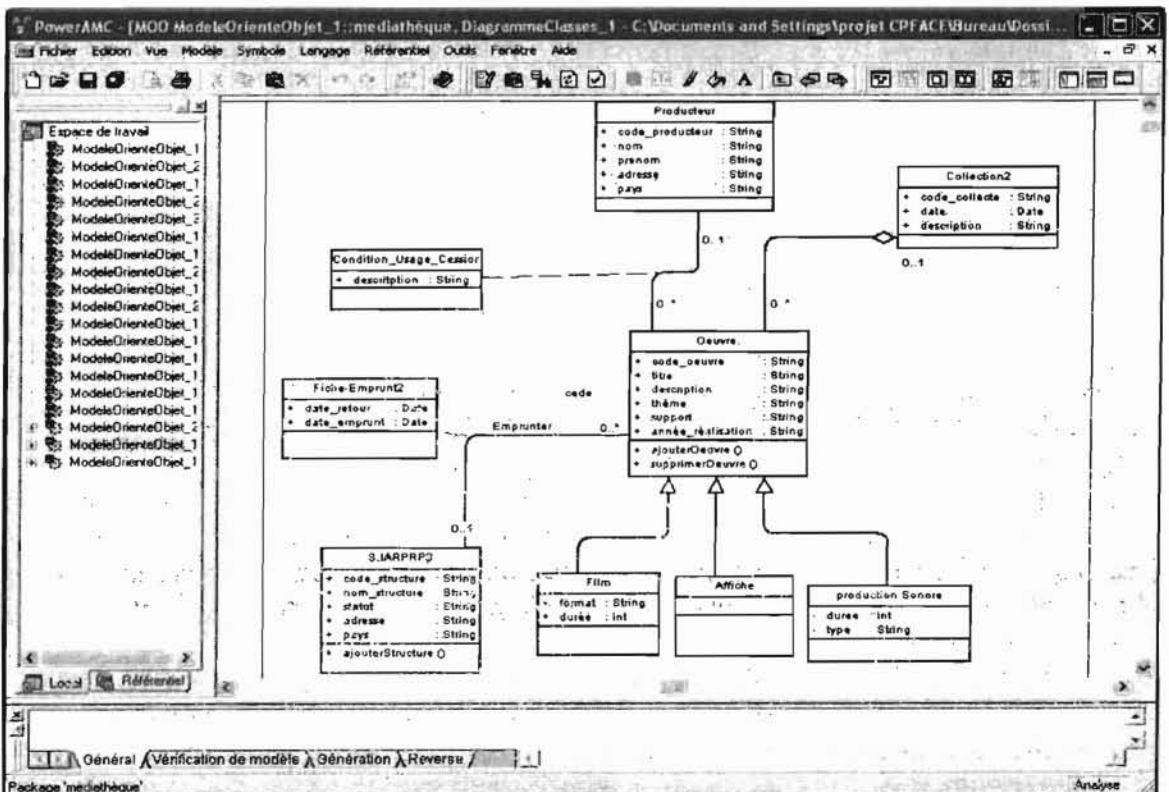


Figure 5.1 Environnement de travail de Sybase Power AMC

5.1.1-IDE : NetBeans 5.5.1

NetBeans est un environnement de développement fonctionnant sous plateforme Windows et Linux. Il permet de développer et déployer rapidement et gratuitement des applications graphiques Swing, des Applets, des JSP/Servlet, des architectures J2EE, dans un environnement fortement personnalisable.

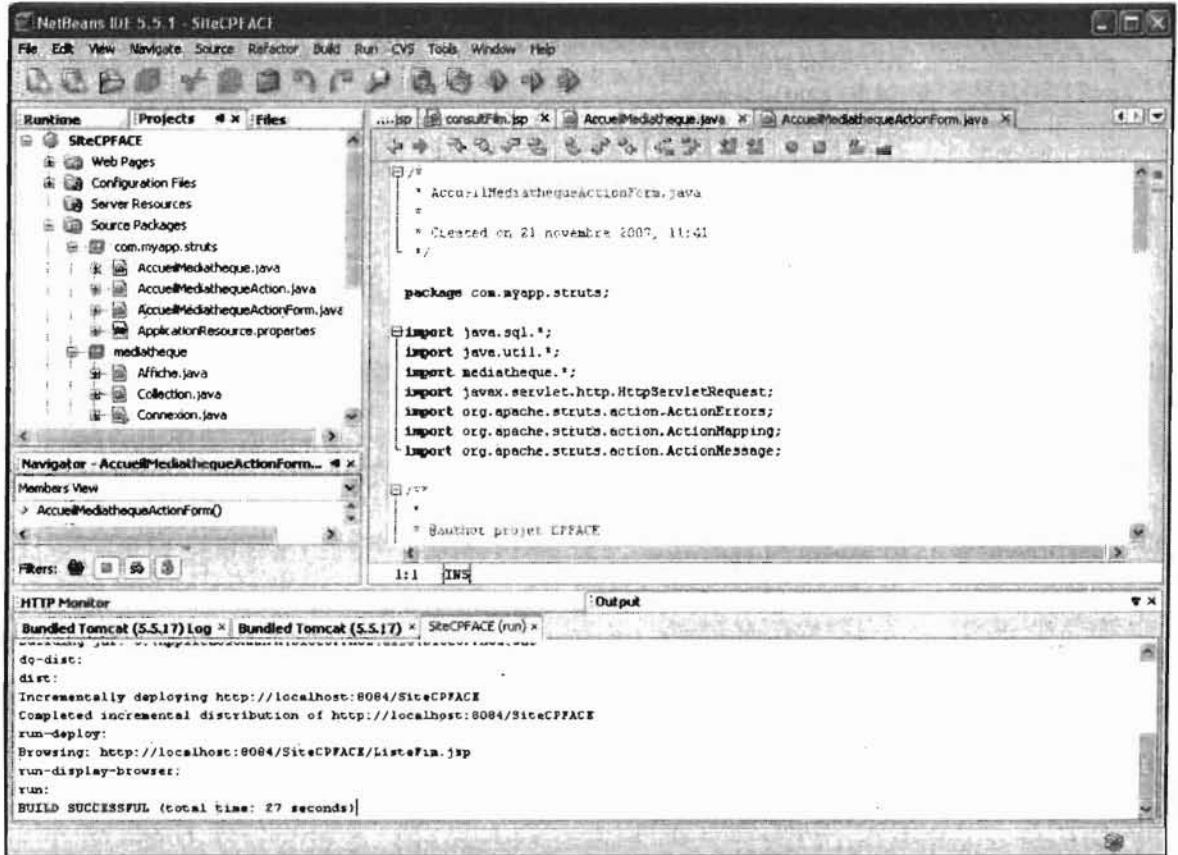


Figure 5.2 Environnement de développement NetBeans 5.5.1

5.1.2- Serveur d'application Web : Tomcat

Tomcat est un conteneur Web J2EE, hébergeant des applications Web composées de servlets et/ou JSP. Il peut également faire office de serveur Web grâce à une servlet dédiée au service de fichiers. On pourra lui adjoindre un serveur Web en amont (Apache Web Server par exemple) pour le décharger de cette tâche de fourniture des ressources (pages HTML, images, etc.) statiques.

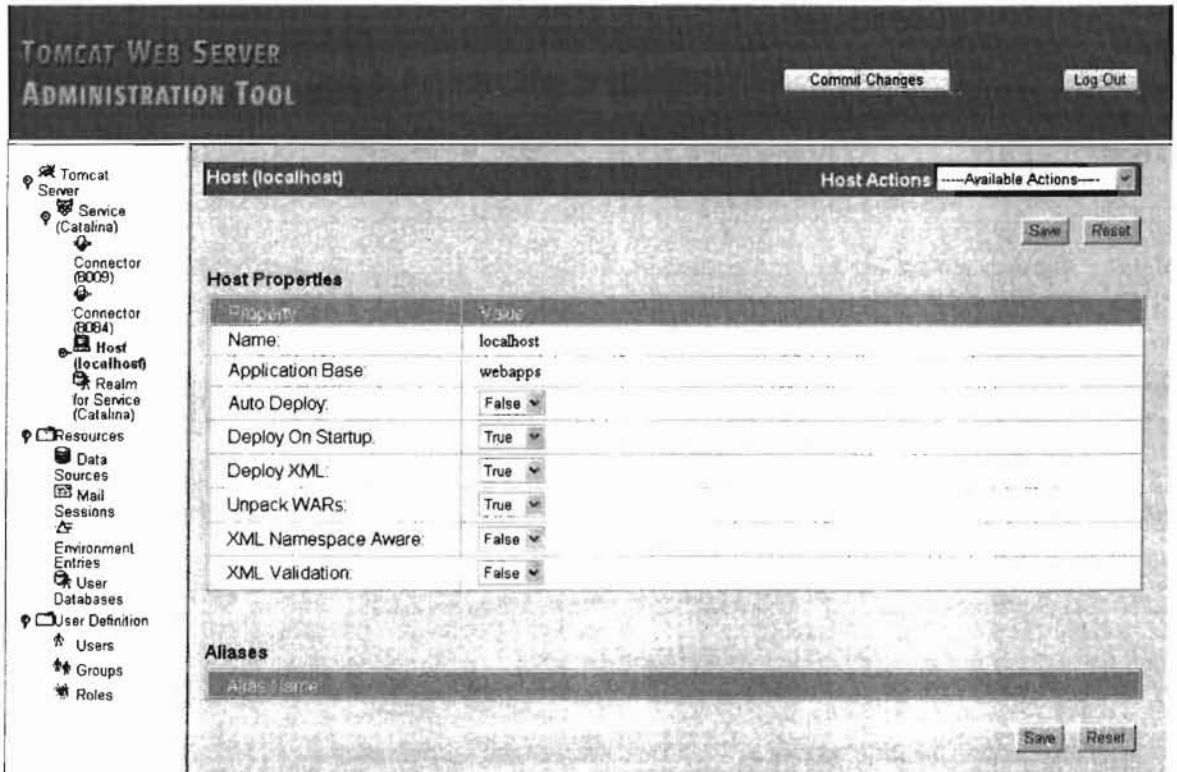
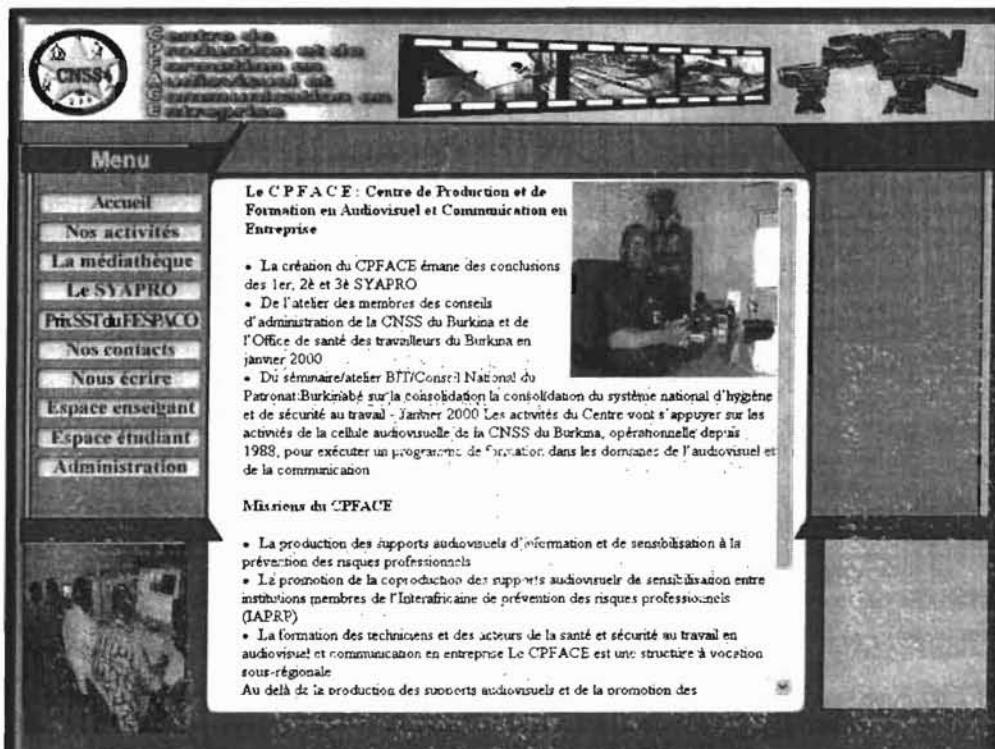
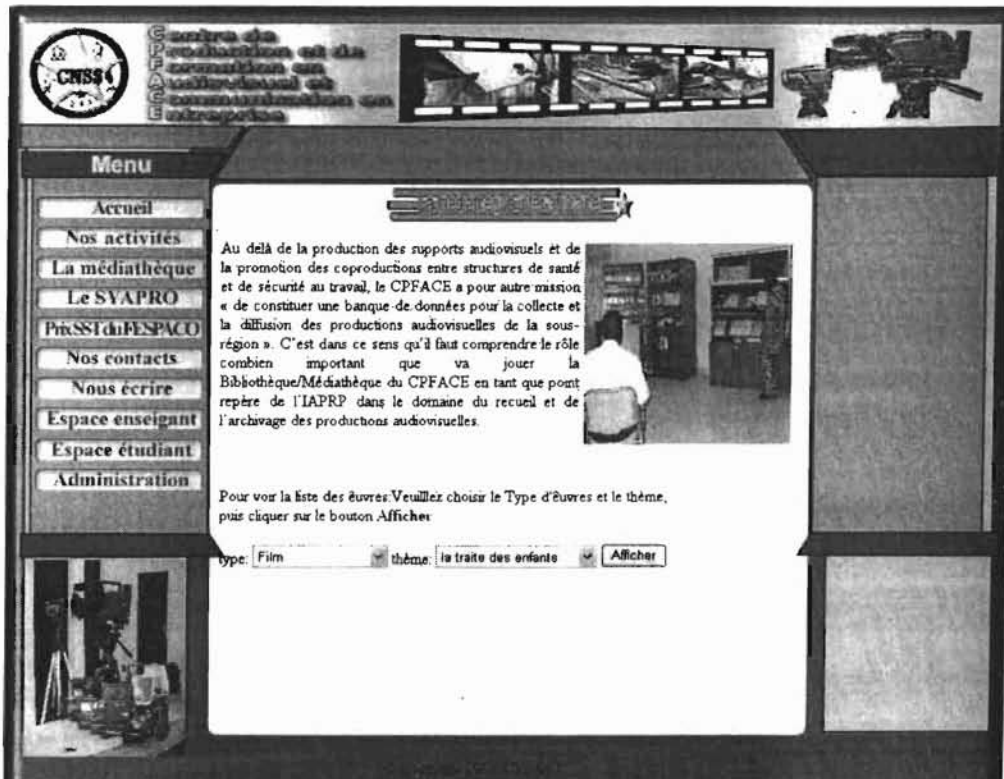


Figure 5.3 Environnement web : Serveur web Tomcat

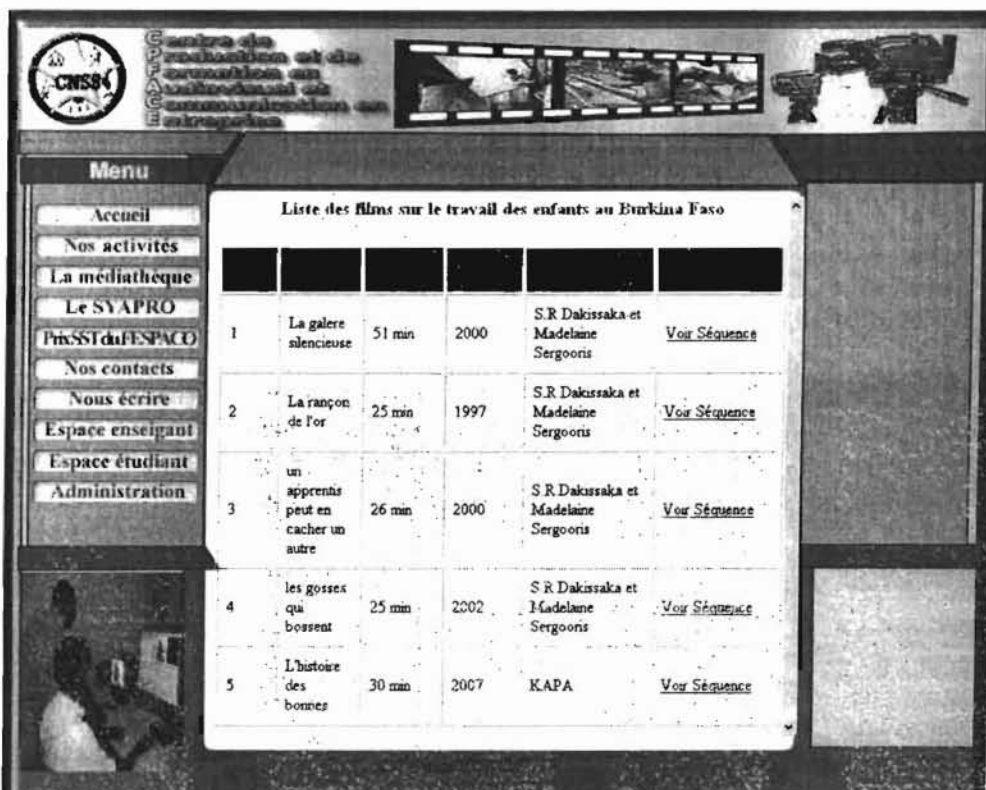
5.2-Présentation de quelques maquettes graphiques



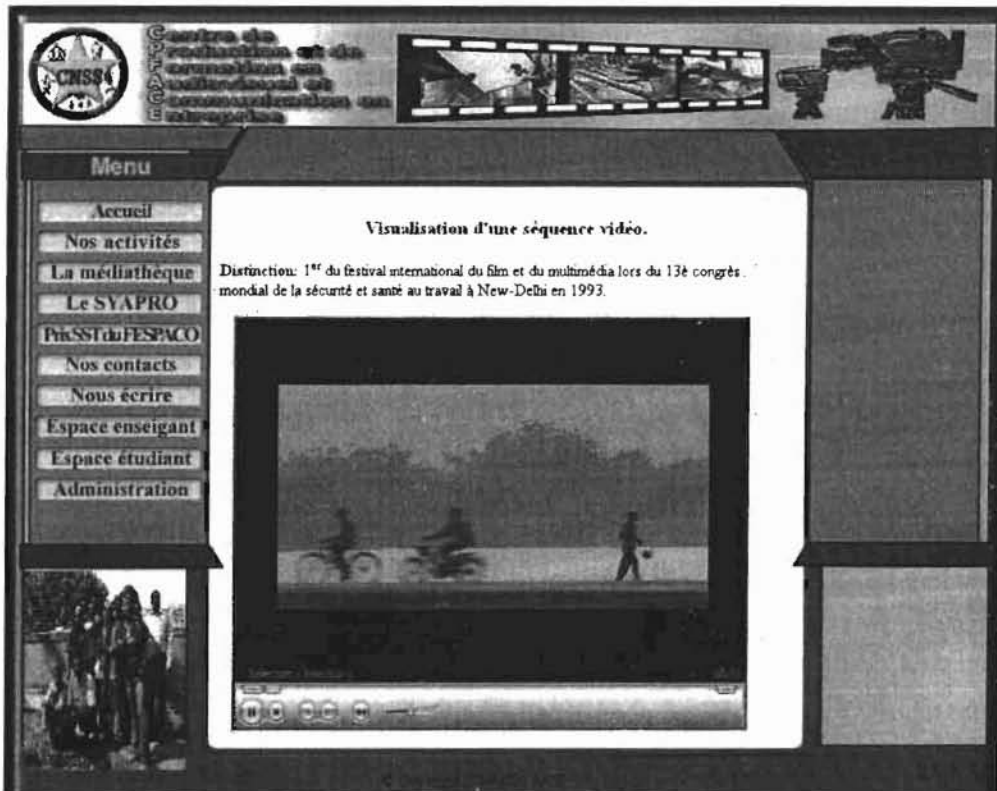
Maquette 5.1 la vue de la page d'accueil



Maquette 5.2 la vue de la page médiathèque



Maquette 5.3 la vue de la liste des films par thème



Maquette 5.4 la vue de la page de visualisation d'une séquence vidéo

Annexes

1-Organigramme de la Casse Nationale de Sécurité Sociale

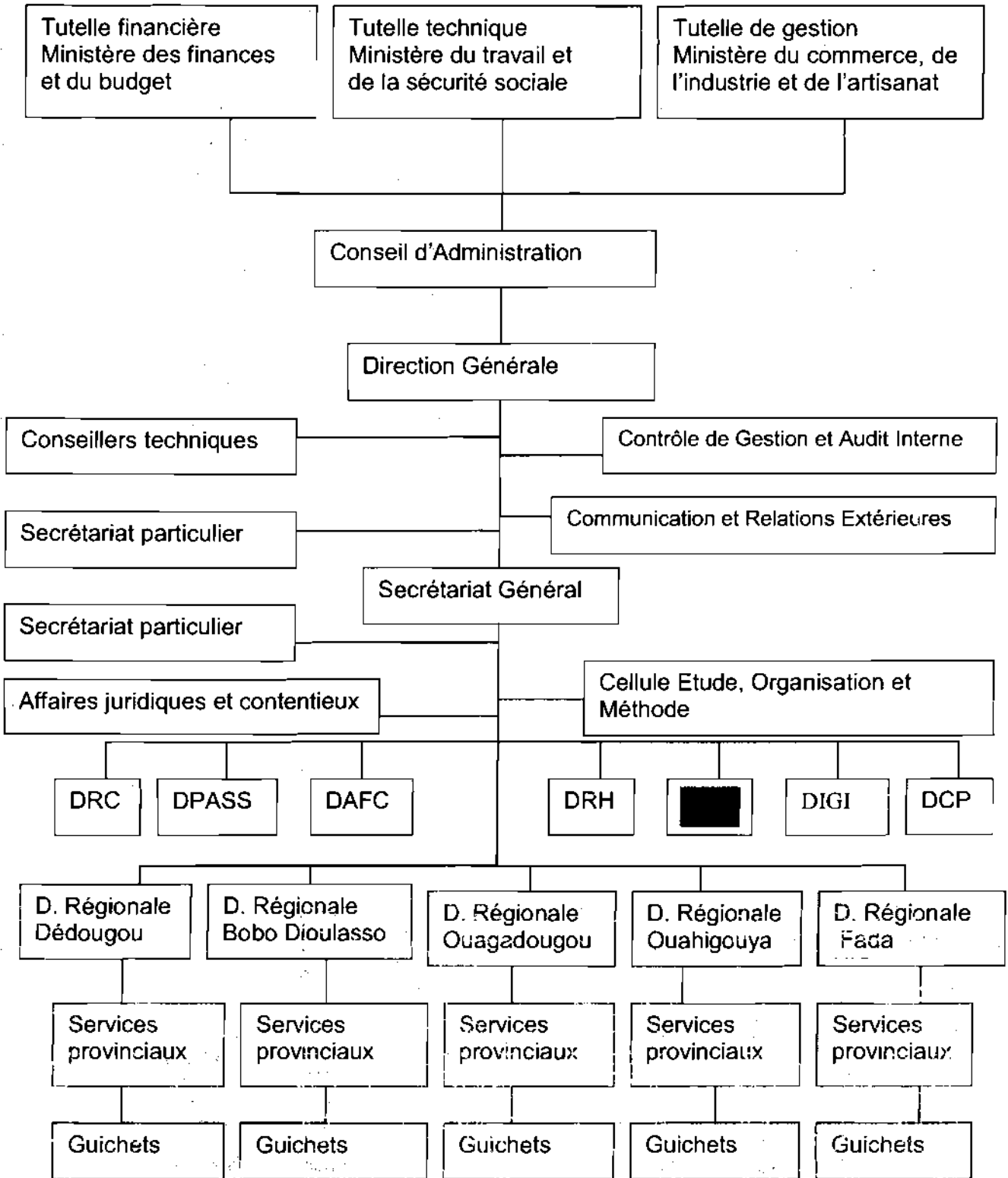


Figure 1 Organigramme de la CNSS

2-Diagramme des cas d'utilisation

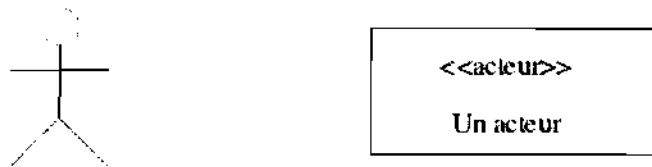
Bien souvent, la maîtrise d'ouvrage et les utilisateurs ne sont pas des informaticiens. Il leur faut donc un moyen simple d'exprimer leurs besoins. C'est précisément le rôle des diagrammes de cas d'utilisation qui permettent de recueillir, d'analyser et d'organiser les besoins, et de recenser les grandes fonctionnalités d'un système. Il s'agit donc de la première étape UML d'analyse d'un système.

Un diagramme de cas d'utilisation capture le comportement d'un système, d'un sous-système, d'une classe ou d'un composant tel qu'un utilisateur extérieur le voit. Il scinde la fonctionnalité du système en unités cohérentes, les cas d'utilisation, ayant un sens pour les acteurs. Les cas d'utilisation permettent d'exprimer le besoin des utilisateurs d'un système, ils sont donc une vision orientée utilisateur de ce besoin au contraire d'une vision informatique.

☞ **Diagramme de cas d'utilisation:** représentent les cas d'utilisation du système, les acteurs et les relations existant entre eux.

Les diagrammes de cas d'utilisation décrivent sous la forme d'actions et de réactions, le comportement d'un système du point de vue d'un utilisateur. Ils permettent de définir les limites du système et les relations entre le système et l'environnement. Ce type de diagrammes interviennent tout au long du cycle de développement, depuis le cahier des charges jusqu'à la fin de la réalisation.

☞ **acteur:** représente un rôle joué par une personne ou une chose qui interagit avec un système.



Un acteur

Figure 2. Représentation d'un acteur

Un acteur peut également participer à des relations de généralisation/spécialisation :

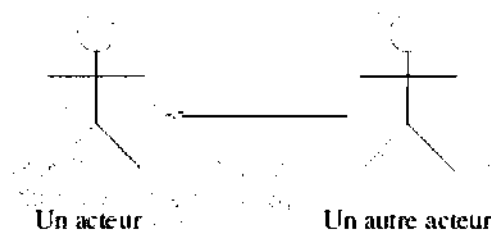


Figure 3. Généralisation entre acteurs

☞ **cas d'utilisation:** est une modélisation d'une fonctionnalité ou d'une classe.

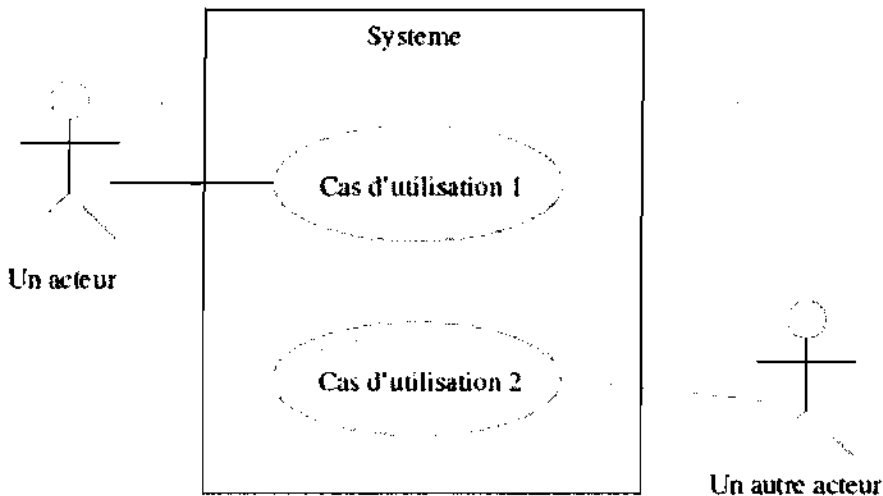


Figure 4. Représentation d'un cas d'utilisation

Les cas d'utilisation se déterminent en observant et en précisant, acteur par acteur, les séquences d'interaction du point de vue de l'utilisateur. Ils se décrivent en termes d'informations échangées et d'étapes dans la manière d'utiliser le système. Un cas d'utilisation regroupe une famille de scénarios d'utilisation selon un critère fonctionnel. Ils décrivent des interactions potentielles, sans entrer dans les détails de l'implémentation.

Il existe trois types de relations entre cas d'utilisation :

- La relation **de généralisation** : le cas d'utilisation enfant est une spécialisation du cas d'utilisation parent.

- La relation **d'inclusion** : le cas d'utilisation source comprend également le comportement de son cas d'utilisation destination. Cette relation a un caractère obligatoire (à la différence de la généralisation) et permet ainsi de décomposer des comportements partageables entre plusieurs cas d'utilisation différents.

- La relation **d'extension** : le cas d'utilisation source ajoute son comportement au cas d'utilisation destination.

L'extension peut-être soumise à condition. Cette relation permet de modéliser des variantes de

Comportement d'un cas d'utilisation.

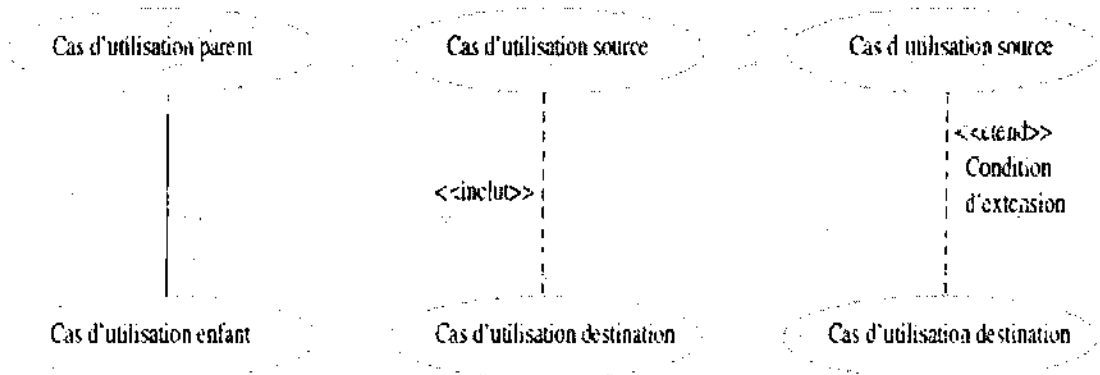


Figure5. Les trois relations entre cas d'utilisation

3-Diagramme de séquence

☞ **Diagramme de séquence:** montre des interactions entre objets selon un point de vue temporel. Ce type de diagramme sert à modéliser les aspects dynamiques des systèmes temps réels et des scénarios complexes mettant en œuvre peu d'objets.

Dans ce type de diagrammes, l'accent est mis sur la chronologie des envois de messages. La représentation se concentre sur l'expression des interactions et non pas sur l'état ou le contexte des objets. Ce type de diagramme est usuellement utilisé pour illustrer les diagrammes de cas d'utilisation.

☞ **Interactions:** modélisent un comportement dynamique entre objets. Elles se traduisent par l'envoi de messages entre objets. Un diagramme de séquence représente une interaction entre objets, en insistant sur la chronologie des envois de messages.

Dans un diagramme de séquence, les objets sont associés à une ligne de vie. La dimension verticale de celle-ci représente l'écoulement du temps (du haut vers le bas). Notons que la disposition des objets sur l'axe horizontal n'est pas importante dans ce type de diagrammes.

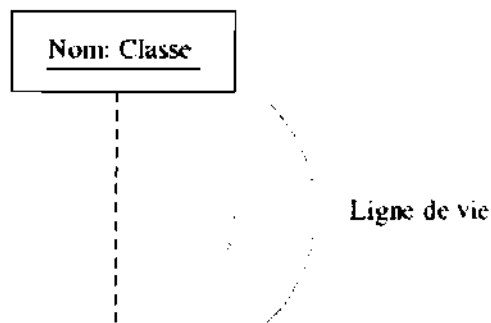


Figure 6. Un objet et sa ligne de vie

☞ **Message:** représentation d'une communication au cours de laquelle des informations sont échangées.

Les messages sont représentés par des flèches et leur ordre est donné par leurs positions sur la ligne de vie. Ils représentent toute forme de communication entre objets : appels de procédures, signaux, interruptions matérielles...

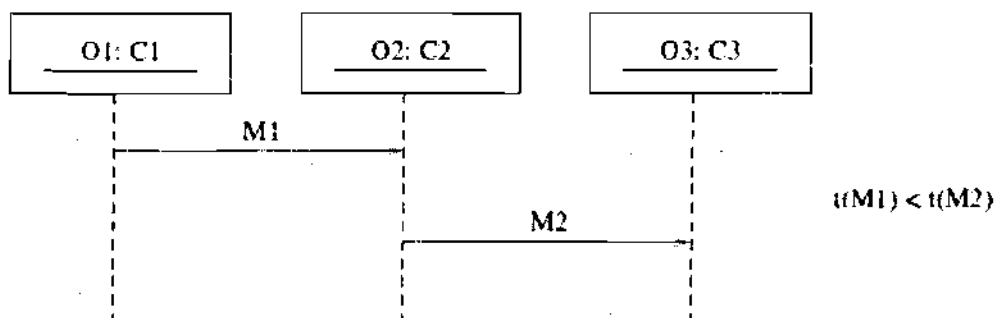


Figure 7. Agencement de messages

☞ **Période d'activation:** correspond au temps pendant lequel un objet effectue une action, soit directement, soit par l'intermédiaire d'un autre objet.

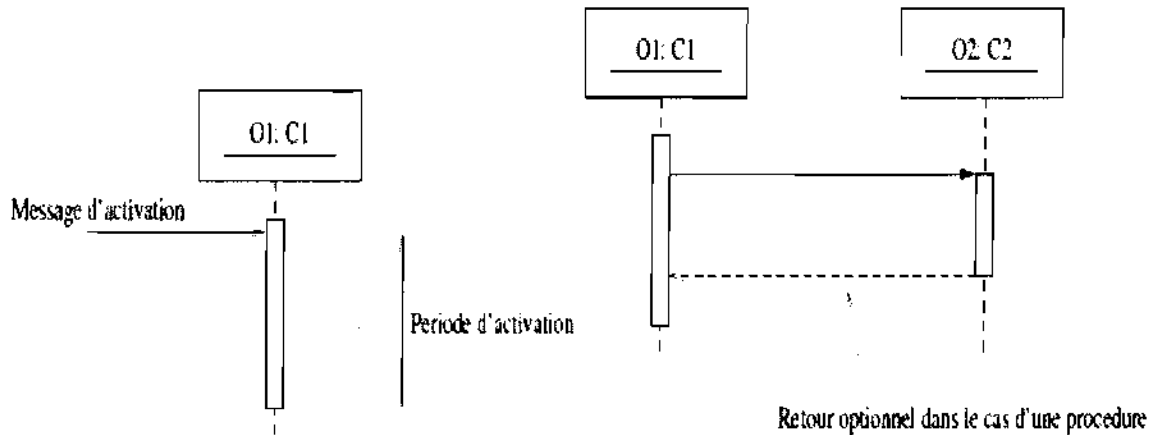


Figure 8. Activations d'un objet de manière simple et par l'intermédiaire d'un autre objet

Si l'on considère le second schéma de la Figure 6.5, O1 active O2. La période d'activation de O1 recouvre celle de O2. Dans le cas d'un appel de procédure, O1 est bloqué jusqu'à ce que O2 lui redonne la main.

Notons que la période d'activation d'un objet n'a aucun rapport avec sa création ou sa destruction. Un objet peut être actif plusieurs fois au cours de son existence.

Il existe deux catégories d'envoi de messages : Les flots de contrôle à plat et les flots de contrôle emboîtés.

☞ **Flots de contrôle à plat:** simple progression vers la prochaine étape d'une séquence.

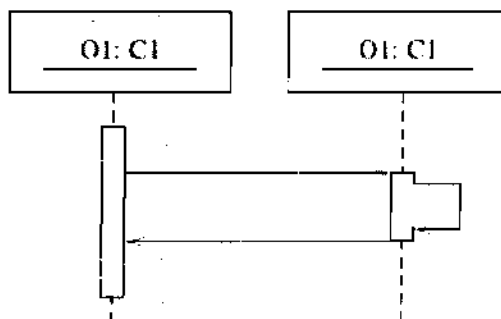


Figure 9. Un flot de contrôle à plat

☞ **Flots de contrôle emboîté:** la séquence emboîtée doit se terminer pour que la séquence englobante reprenne le contrôle. Ce type de flot est utilisé pour représenter des appels de procédures avec un retour de valeur optionnel.

Les diagrammes suivants présentent quelques constructions communes : récursivité, réflexion, création et destruction,...

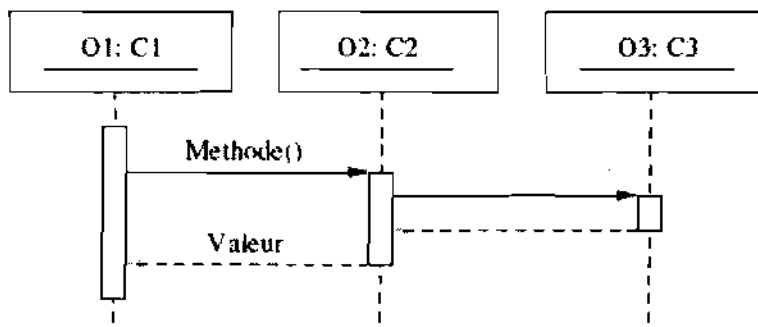


Figure 10. Un flot de contrôle emboîté

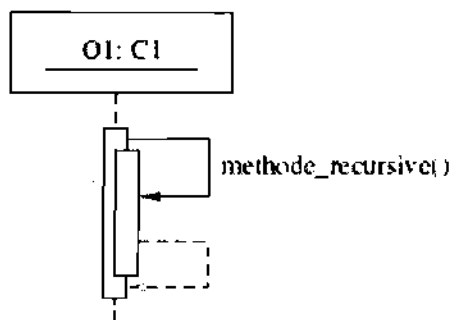


Figure 11. Représentation de la récursivité

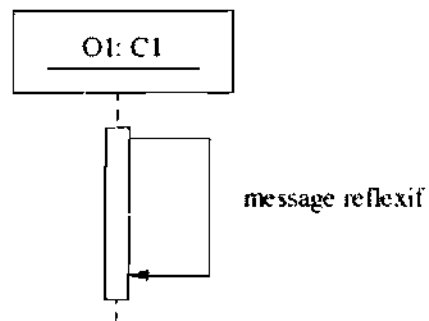


Figure12. Représentation de la réflexion

Dans le cas d'un objet composite, cette construction peut indiquer un point d'entrée dans une activité de plus bas niveau qui s'exerce au sein de l'objet :

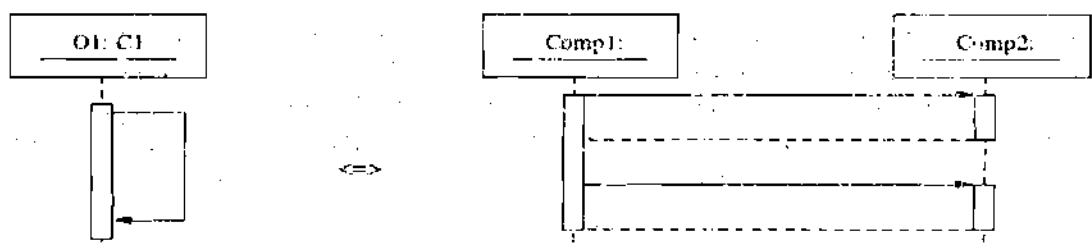


Figure 13. Masquage d'informations par utilisation de la réflexion

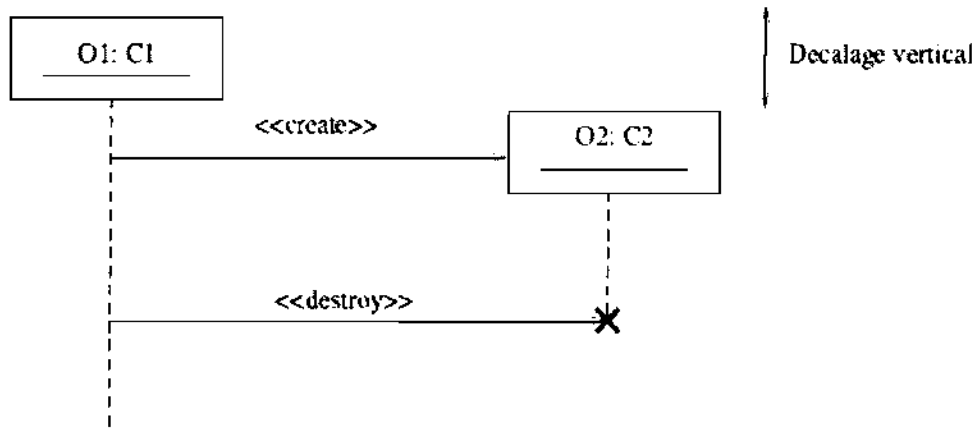


Figure 14. Création et destruction d'objets

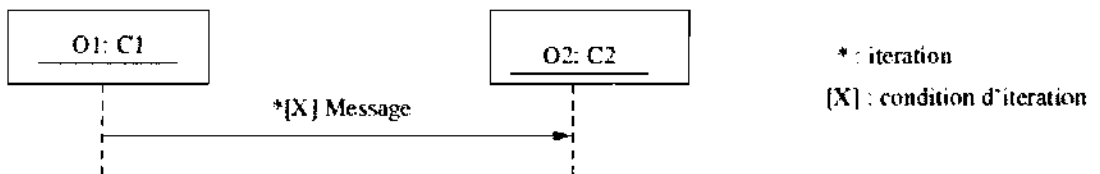


Figure 15. La boucle « while »

La séquence emboîtée doit se terminer pour que la séquence englobante reprenne le contrôle. Ce type de flot est utilisé pour représenter des appels de procédures avec un retour de valeur optionnel

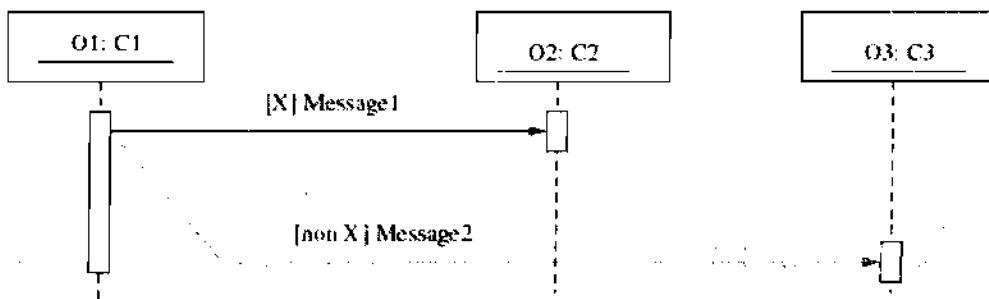


Figure 16. Le branchement conditionnel « if...then...else... »

4-Diagramme de classe

☞ **Diagrammes de classe:** expriment de manière générale la structure statique d'un système, en termes de classes et de relations entre ces classes.

☞ **paquetage (package):** mécanisme d'ordre général qui permet d'organiser les éléments en groupes.

Les packages permettent de définir des sous-systèmes formés d'éléments ayant entre eux une certaine logique.

Leurs caractéristiques sont les suivantes :

- . Ils regroupent des éléments de modélisation selon des critères purement logiques ;
- . Ils permettent d'encapsuler des éléments de modélisation par l'intermédiaire d'interfaces ;
- . Ils permettent de structurer un système en catégories ou sous-systèmes ;
- . Ils servent de « briques » réutilisables dans la conception d'un logiciel.

Chaque package doit avoir un nom différent de celui des autres packages et peut être composé d'autres éléments, y compris d'autres packages. Les éléments contenus sont en fait « possédés » (au sens UML du terme), ce qui signifie que la destruction d'un package implique la destruction de tous ses éléments.

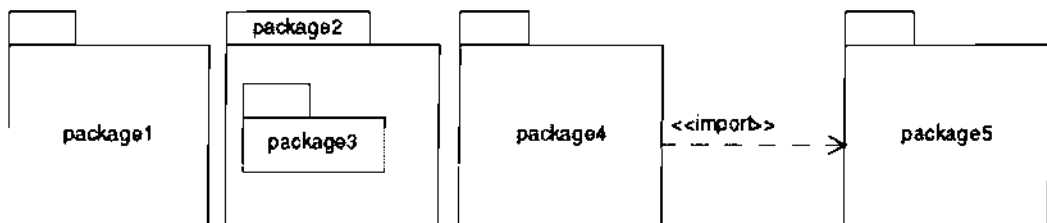


Figure 17 Représentation graphique d'un package

☞ **Classe:** représentation d'un ensemble d'éléments partageant les mêmes attributs, les mêmes opérations, les mêmes relations et les mêmes sémantiques.

En programmation orientée objet, une classe définit une abstraction, un type abstrait qui permettra d'instancier

des objets. Graphiquement, une classe décrite en UML peut être plus ou moins précise :

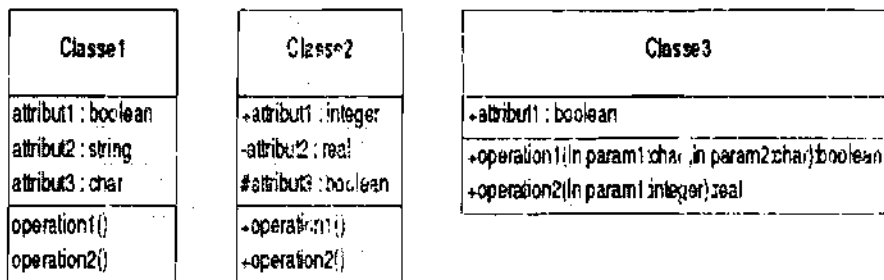


Figure 18 Représentation graphique d'une classe

La première classe (classe1) est dite à visibilité réduite, tandis que les deux autres (classe2 et classe3) sont dites à visibilités détaillées.

☞ **Visibilité d'une caractéristique:** détermine si d'autres éléments peuvent l'utiliser.

Trois niveaux de visibilité sont possibles pour les attributs et les opérations :

- « + » : visibilité public. La caractéristique peut être utilisée par n'importe quelle instance ayant une visibilité sur les instances de la classe complète.
- « - » : visibilité private : La caractéristique ne peut être utilisée que par des instances de la classe elle-même.
- « # » : visibilité protected : La caractéristique ne peut être utilisée que par des instances de la classe elle-même ou bien par les descendants directs de cette classe.

☞ **Classe abstraite:** classe ne pouvant pas être instanciée directement. Une telle classe sert de spécification pour des objets instances de ses sous-classes.

☞ **Relation d'association:** relation structurelle précisant que les objets d'un élément sont reliés aux objets d'un autre élément.

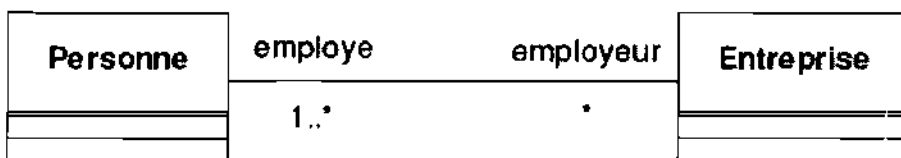


Figure 19 Représentation graphique d'une association

☞ **Relation de dépendance:** relation sémantique entre deux éléments selon laquelle un changement apporté à l'un peut affecter la sémantique de l'autre.

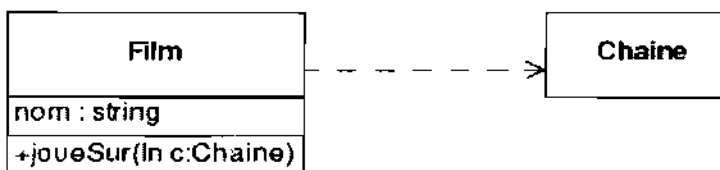


Figure 20 Représentation graphique d'une dépendance

☞ **Relation de généralisation:** relation entre un élément général et un élément dérivé de celui-ci, mais plus spécifique (désigné par sous-élément ou élément fils).

Le plus souvent, la relation de généralisation est utilisée pour représenter une relation d'héritage.

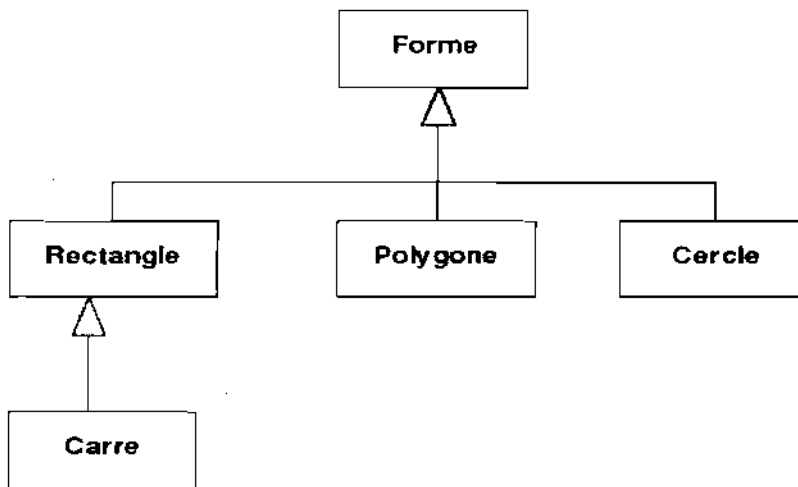


Figure 21 Représentation graphique d'une généralisation

☞ **Relation d'agrégation:** relation «tout-partie» dans laquelle une classe représente un élément plus grand (le «tout») composé d'éléments plus petits (les «parties»). La relation d'agrégation est souvent implémentée en utilisant des membres privés.



Figure 22 Représentation graphique d'une agrégation

☞ **Relation de composition:** relation d'agrégation mettant en avant une notion de propriété forte et de coïncidence des cycles de vie. Les «parties» sont créées et détruites en même temps que le «tout».

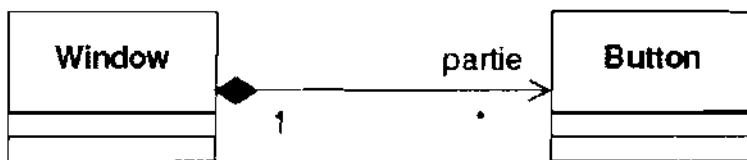


Figure 6.23 Représentation graphique d'une composition

5-Diagramme d'activités

Les diagrammes d'activités permettent de mettre l'accent sur les traitements. Ils sont donc particulièrement adaptés à la modélisation du cheminement de flots de contrôle et de flots de données. Ils permettent ainsi de représenter graphiquement le comportement d'une méthode ou le déroulement d'un cas d'utilisation.

Une action est le plus petit traitement qui puisse être exprimé en UML. Une action a une incidence sur l'état du système ou en extrait une information. Les actions sont des étapes discrètes à partir desquelles se construisent les comportements. La notion d'action est à rapprocher de la notion d'instruction élémentaire d'un langage de programmation (comme C++ ou Java).

☞ **Une action** peut être, par exemple :

- une affectation de valeur à des attributs ;
- un accès à la valeur d'une propriété structurelle (attribut ou terminaison d'association) ;
- la création d'un nouvel objet ou lien ;
- un calcul arithmétique simple ;
- l'émission d'un signal ;
- la réception d'un signal ;

☞ **Une activité** définit un comportement décrit par une séquence organisée d'unités dont les éléments simples sont les actions. Le flot d'exécution est modélisé par des noeuds reliés par des arcs (transitions).

Le flot de contrôle reste dans l'activité jusqu'à ce que les traitements soient terminés.

Une activité est un comportement (behavior en anglais) et à ce titre peut être associée à des paramètres.

☞ **Un groupe d'activités** est une activité regroupant des noeuds et des arcs. Les noeuds et les arcs peuvent appartenir à plus d'un groupe. Un diagramme d'activités est lui-même un groupe d'activités

☞ **Noeud d'activité** (activity node)

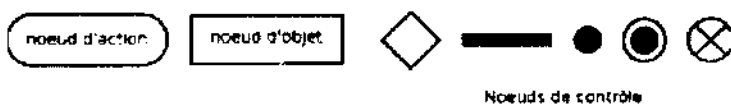


Figure 24 Représentation graphique des noeuds d'activité. De la gauche vers la droite, on trouve : le noeud représentant une action, qui est une variété de noeud exécutable, un noeud objet, un noeud de décision ou de fusion, un noeud de bifurcation ou d'union, un noeud initial, un noeud final et un noeud final de flot.

☞ **Un noeud d'activité** est un type d'élément abstrait permettant de représenter les étapes le long du flot d'une activité. Il existe trois familles de noeuds d'activités :

- les noeuds d'exécutions (executable node en anglais) ;
- les noeuds objets (object node en anglais) ;

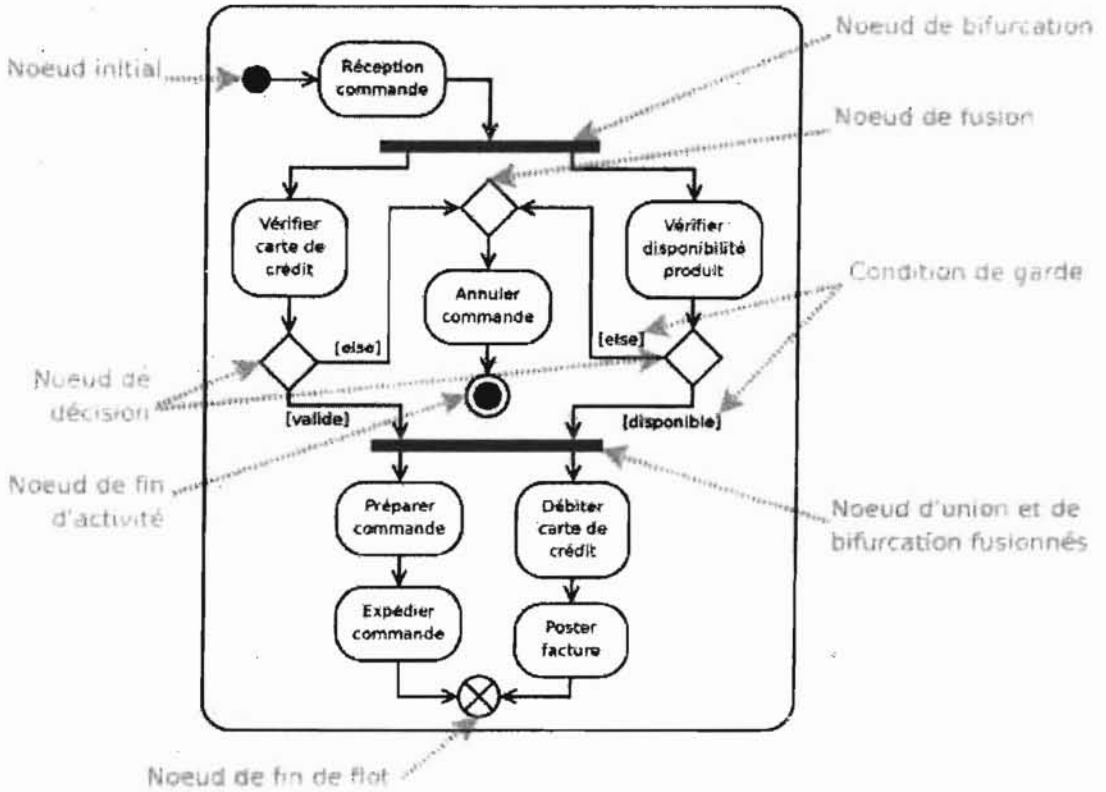


Figure 25 Exemple de diagramme d'activité illustrant l'utilisation de nœuds de contrôle. Ce diagramme décrit la prise en compte d'une commande.

Le passage d'une activité vers une autre est matérialisé par une transition. Graphiquement les transitions sont représentées par des flèches en traits pleins qui connectent les activités entre elles. Elles sont déclenchées dès que l'activité source est terminée et provoquent automatiquement et immédiatement le début de la prochaine activité à déclencher (l'activité cible). Contrairement aux activités, les transitions sont franchies de manière atomique, en principe sans durée perceptible.

☞ **Les transitions** spécifient l'enchaînement des traitements et définissent le flot de contrôle.

Nœud exécutable (executable node)

☞ **Un nœud exécutable** est un nœud d'activité qu'on peut exécuter (i.e. une activité). Il possède un gestionnaire d'exception qui peut capturer les exceptions levées par le nœud, ou un de ses nœuds imbriqués.

☞ **Nœud d'action**

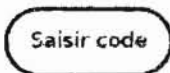


Figure 6.26 Représentation graphique d'un nœud d'action.

6-Diagramme de déploiement

Diagrammes de déploiement: permet de montrer la disposition physique des matériels qui composent le système, ainsi que la répartition des composants sur ces matériels représentés par des nœuds. Ce type de diagramme est utilisé principalement pour la modélisation de trois types de systèmes : les systèmes embarqués, les systèmes client/serveur et les systèmes totalement répartis. Deux notations sont possibles pour montrer qu'un composant réside sur un nœud :

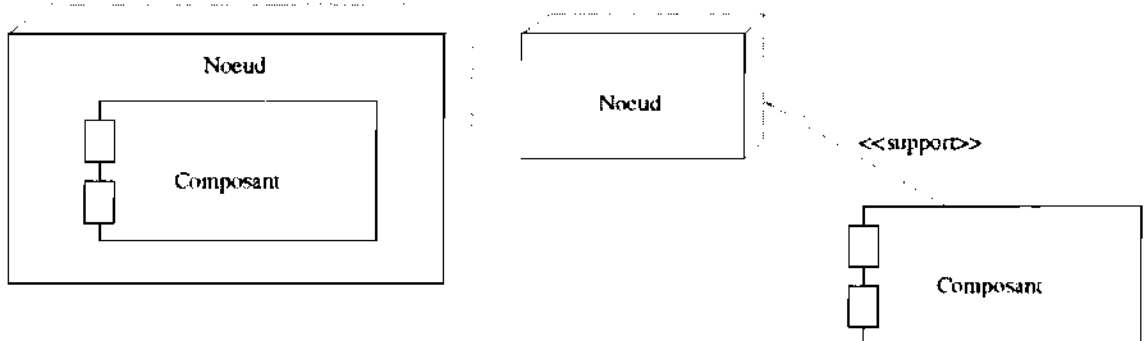


Figure 27 Représentation d'un composant à l'intérieur d'un nœud

La migration et la copie d'un composant d'un nœud vers un autre sont effectuées respectivement en utilisant les stéréotypes <<devient>> et <<copie>>.

Un diagramme de déploiement permet également de représenter les relations entre différents nœuds.

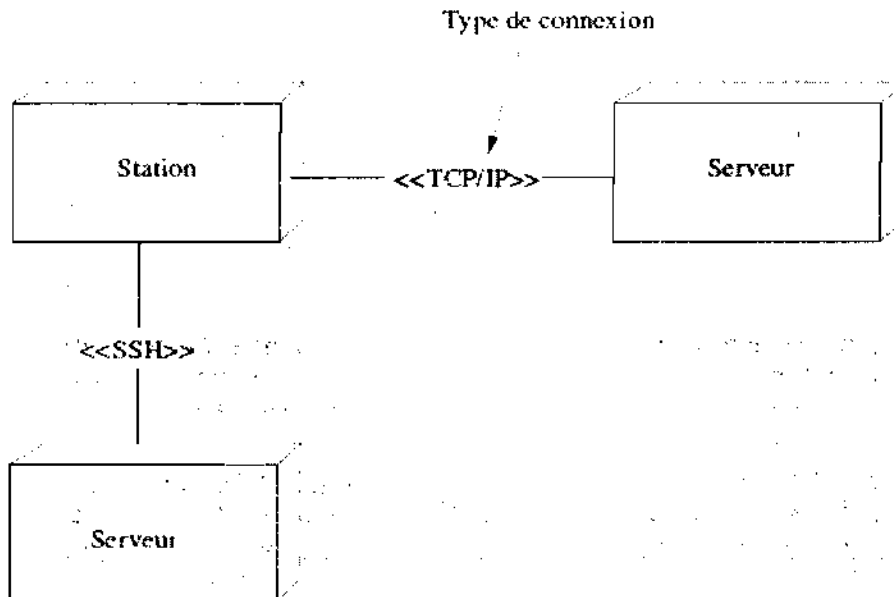


Figure 28 Représentation de relations entre nœuds

Conclusion

De l'étude qui précède il paraît évident que l'informatisation de la gestion des différentes prestations sera un apport considérable pour la bonne marche du CPFACE dans la mesure où elle permettra une ouverture de celui-ci dans la sous région et un meilleur suivi de ses activités.

Dans ce document, il était question de décrire le futur système à l'aide des phases et activités préconisées par la démarche UP. Après avoir déterminé les avantages et les imperfections du système, nous avons proposé des solutions pour pallier à ces insuffisances et pour atteindre les résultats attendus de l'informatisation.

Il s'agissait concrètement de proposer une application qui puisse d'une part donner une visibilité au Centre de Production et de Formation en Audiovisuel et Communication en Entreprise aux yeux de ses partenaires et organes de l'IAPRP et d'autre part permettre une gestion des activités internes du centre. A cet effet une solution a été proposée, celle de la mise en place d'un *site web interactif* pour assurer la visibilité du centre et d'une *application 2-tiers de gestion des activités internes* (gestion de la bibliothèque/médiathèque, gestions des archives du SYAPRO et du prix « SST »,...)

Nous voici aux termes de notre travail. Les difficultés, nous en avons rencontrées certes ; mais elles ne faisaient que nous motiver car en cherchant à les surmonter, nous ciblons à coup sûr le but à atteindre : le travail bien fait. Qu'il nous soit donc permis de nous réjouir, non pas pour avoir fait un travail de perfection mais pour avoir mis à nu nos lacunes. Grâce à vos remarques, critiques et recommandations, combien instructives, dont nous attendons avec joie, nous sommes sûr de sortir vainqueur des difficultés.

BIBLIOGRAPHIE

- Laurent DEBRAUWER et Fien VAN DER HEYDE, « UML2 Modélisation des objets », Edition ENI, Octobre 2007 ;
- Laurent AUDIBERT, « UML 2.0 (IUT, département informatique, 1^{ère} année) »,
- CONCOURS DE PRODUCTIONS AUDIOVISUELLES SUR LA PREVENTION DES RISQUES PROFESSIONNELLES (5^e SYAPRO, 26 au 29 septembre 2006), « règlement intérieur » ;
- Rapport général et financier de l'organisation du prix santé et sécurité au travail du 18^{ème} FESPACO du 22 février au 1^{er} mars 2003.
- Hawaiï Team, <http://www.hile.free.fr>
- <http://www.developpez.com>