

UNIVERSITE POLYTECHNIQUE DE BOBO-DIOULASSO
(UPB)

ECOLE SUPERIEURE D'INFORMATIQUE
(ESI)

CYCLE DES INGENIEURS DE CONCEPTION
EN INFORMATIQUES
(CICI)

3^{eme} année



INTEGRATION D'UN MODULE DE SCAN DANS UNE APPLICATION WEB

Présenté par **TRAORE Lamine** en vue de l'obtention du

Diplôme d'Ingénieur de Conception en Informatiques

Maitre de stage

M. KONDE Hyppolite,
Manager du Pôle Développement et
Intégration Logiciel à SOFTNET
BURKINA

Superviseur

Dr SOME Borli Michel,
Enseignant-Chercheur à l'Ecole
Supérieure d'Informatique

DEDICACES

Au tout puissant **Dieu**, que ta grâce et ta miséricorde puisse
être mes compagnons de tous les jours.

A feu mon **père**, Monsieur TRAORE Bakari.

A ma défunte **mère** madame TRAORE née TRAORE
Fatoumata dite Tata.

A toutes mes **sœurs**, vos soutiens divers et combien
inestimables me fortifie et me guide dans tout ce que je fais.

REMERCIEMENTS

Nous ne saurons commencer ce présent travail sans remercier toutes les personnes qui nous ont apporté leur soutien et leur aide. Ainsi nous tenons à remercier particulièrement :

- Le Président Directeur Général, de SOFTNET GROUP pour nous avoir accordé ce stage ;
- Monsieur OUEDRAOGO Abdoul Karim, Directeur Technique de SOFTNET BURKINA ;
- Monsieur KONDE Hyppolite, manager du Pôle Développement et Intégration Logiciel à SOFTNET BURKINA ;
- Dr SOME Borlli Michel, mon superviseur ;
- Dr OUATTARA/COMPAORE Anastasie, responsable des stages à l'Ecole Supérieure d'Informatique ;
- Tout le corps enseignant de l'Ecole Supérieure d'Informatique ;
- L'Ecole Supérieure d'Informatique pour la formation reçue ;
- Tous ceux qui de près ou de loin ont apporté leur contribution et leur aide dans la réalisation de ce travail.

RESUME

La problématique posée dans cette présente étude est de trouver des moyens et des solutions pour permettre la mise en œuvre de la gestion de scanner dans un environnement web.

Aussi, notre travail a consisté à trouver des moyens et des outils à même de permettre d'utiliser un scanner dans un environnement web avec la possibilité de faire la reconnaissance de caractère. Il s'agit de voir comment insérer les fonctions d'un scanner dans un navigateur web et de pouvoir récupérer le document scanné sur un serveur ainsi que de pouvoir récupérer les informations sur le document afin de les insérer dans une base de données.

Le résultat doit pouvoir s'utiliser dans un environnement web soit indépendamment ou avec une application existante.

Ainsi, dans le cadre de l'étude nous nous sommes rendu compte que des ébauches de solutions existent pour le problème mais les réflexions se poursuivent toujours afin de trouver une solution pérenne et durable à cette problématique. Il existe en effet plusieurs entreprises qui se sont fait la leur la question et qui mène des réflexions quotidiennes afin de prendre en compte toutes les dimensions du problème.

Pour notre part, nous nous sommes basé sur le fruit de leur travail pour obtenir des résultats à mettre à notre actif et qui permettrons pourquoi pas à guider d'autres personnes à leur tour pour booster le domaine de la connaissance en la matière

Mots clés : scanner, scan, OCR, WebAPI.

ABSTRACT

The issues raised in this study is to find ways and solutions to enable the implementation of scanner management in a web environment.

So, our job consists to find ways and tools to be able to permit to use a scanner in a web environment with the ability to do the character recognition. It's about to insert the functions of a scanner in a web browser and permits to save the scanned document on a server and put information in the document into a database.

The result must be used in a web environment either independently or with an existing application.

Thus, as part of the study we found that solutions exist to draft the problem but the reflections are ongoing to find a lasting and sustainable solution to this problem. Indeed, there are several companies that have made their own the issue and leads daily reflections to take into account all the dimensions of the problem.

For our part, we based on the results of their work to get results to put to our credit, and why not allow other people to lead in turn to boost the field of knowledge on the subject

Keywords: scanner, scan, OCR, WebAPI.

SOMMAIRE

DEDICACES	I
REMERCIEMENTS	II
RESUME	III
ABSTRACT	IV
SOMMAIRE	V
LISTE DES FIGURES	VII
LISTE DES TABLEAUX	VIII
SIGLES ET ABREVIATIONS	IX
PREAMBULE	X
INTRODUCTION GENERALE	1
CHAPITRE I. PRESENTATION GENERALE	3
I. Présentation de la structure d'accueil	3
1. SOFTNET-GROUP	3
2. Vision	3
3. Organisation	3
4. Organigramme	4
II. Présentation du thème	6
III. Problématique de l'étude	6
1. Description du projet	6
2. Objectifs	7
3. Résultats attendus	7
CHAPITRE II. DEMARCHES POUR L'ETUDE	8
I. Méthodes d'étude et de conception	8
1. UML	8
2. Modèle en V	10
II. Outils utilisés	10
1. Environnement de développement intégré : Visual Studio 2012.....	10
2. Outils de modélisation : Microsoft Office Visio 2007	13
3. Serveur de déploiement	14
III. Acteurs du projet	15
IV. Planning de déroulement global	16
CHAPITRE III. ETAT DE L'ART	17
I. Historique	17
II. Définitions et concepts	18
III. Principe de fonctionnement	18
IV. Quelques solutions de gestion des scanners	20

V. Intérêt de l'étude.....	21
CHAPITRE IV. ETUDE DE L'EXISTANT	23
I. Etude détaillée	23
II. Etude technique	24
CHAPITRE V. ETUDE DU FUTUR SYSTEME.....	26
I. Délimitation du projet.....	26
II. Modélisation du système	26
1. Diagramme de cas d'utilisation du système.....	27
2. Diagramme de classes.....	33
3. Diagrammes de séquence.....	36
4. Diagrammes d'activités	37
5. Diagramme d'état transition	39
III. Architecture réseau.....	40
IV. Etude comparative des différentes scenarii	43
1. Premier scenario : utilisation des modules natifs inclus dans les scanners .	43
2. Deuxième scénario : acquisition d'un module de gestion de scan.....	43
3. Troisième scenario : développement du plugin et d'une interface applicative	45
4. Choix d'un scénario	45
CHAPITRE VI. REALISATION DU SYSTEME.....	47
I. Etude détaillée	47
1. Modèle MVC.....	47
2. Projet de scan universel : TWAIN Initiative	48
3. Reconnaissance des caractères.....	48
4. Service REST	49
II. Choix techniques de réalisation	51
1. Architecture logique	51
2. ASP.Net MVC	51
3. Base de Données : MS SQL Server.....	52
III. Mise en œuvre.....	54
1. Outils et Technologies	54
2. Résultats obtenus	57
3. Présentation du prototype	58
CONCLUSION GENERALE ET PERSPECTIVES	61
ANNEXE.....	B
BIBLIOGRAPHIE	B

LISTE DES FIGURES

Figure 01 : Organigramme de SOFTNET.....	5
Figure 02 : Illustration du cycle en V	10
Figure 03 : Architecture de .Net Framework	12
Figure 04 : Planning prévisionnel.....	16
Figure 05 : Principe de fonctionnement d'un scanner	19
Figure 07 : Diagramme de classe du système	35
Figure 08 : Diagramme de séquence : Lancer un scan	36
Figure 09 : Diagramme de séquence : Reconnaître les caractères.....	37
Figure 10 : Diagramme d'activité : Lancer scan	38
Figure 11 : Diagramme d'activité : Reconnaître les caractères.....	39
Figure 12 : Diagramme d'état-transition du système	40
Figure 13 : Architecture réseau de l'application.....	41
Figure 14 : Modèle-Vue-Contrôleur	47
Figure 15 : Architecture logique	51
Figure 16 : Ecran 1 : Page d'accueil pour le démarrage du scan.....	58
Figure 17 : Ecran 2 : Page de sélection du scanner.....	59
Figure 18 : Ecran 3 : Page d'affichage des options de numérisation.....	59
Figure 19 : Ecran 4 : Page d'enregistrement sur un serveur.....	60

LISTE DES TABLEAUX

Tableau 01 : Différents diagrammes utilisés dans le projet.....	9
Tableau 02 : Acteurs du projet	15
Tableau 03 : Scénario nominal : Initialisation	29
Tableau 04 : Scénario alternatif : Initialisation.....	30
Tableau 05 : Scénario nominal : Lancer un scan.....	30
Tableau 06 : Scénario alternatif : Lancer un scan	31
Tableau 07 : Scénario nominal : Enregistrer sur un serveur.....	31
Tableau 08 : Scénario alternatif : Enregistrer sur un serveur	32
Tableau 09 : Scénario nominal : Reconnaître les caractères	32
Tableau 10 : Scénario alternatif : Reconnaître les caractères	33
Tableau 11 : Scénario d'exception : Reconnaître les caractères.....	33
Tableau 12 : Formalisme de l'architecture réseau	42
Tableau 13 : Etude comparative de quelques entreprises évoluant dans le domaine du scan utilisant une interface web.....	44
Tableau 14 : Etude comparative des scénarii	46

SIGLES ET ABREVIATIONS

API	Application Programming Interface
ASMO	Arabie Standard Metrology Organisations
CICI	Cycle des Ingénieurs en Conception Informatiques
CLR	Common Langage Runtime
CLS	Common Language Specification
CU	Cas d'Utilisation
ESI	Ecole Supérieure d'Informatique
FTP	File Transfer Protocol
HTTP	HyperText Transfer Protocol
IDE	Integrated Development Environment
IHM	Interface Homme Machine
IIS	Internet Information Services
MSDE	Microsoft SQL Server Desktop Engine
MSIL	MicroSoft Intermediate Language
MVC	Modèle Vue Contrôleur
OCR	Optical Character Recognition
SCSI	Small Computer System Interface
SDK	Software Development Kit
SGBD	Système de Gestion de Base de Données
SGBDR	Système de Gestion de Base de Données Relationnelles
SMTP	Simple Mail Transfer Protocol
UML	Unified Modeling Language
UPB	Université Polytechnique de Bobo-Dioulasso
URI	Unified Resource Identifier
USB	Universal Serial Bus
XML	eXtended Markup Language

PREAMBULE

L'Université polytechnique de Bobo-Dioulasso (UPB), jadis Centre Universitaire Polytechnique de Bobo-Dioulasso (CUPB), a été créée le 19 septembre 1995 par décret N° 95-340/PRES/MESSRS. Elle est située dans la région de l'Ouest du Burkina-Faso, plus précisément dans le village de Nasso à une quinzaine de kilomètres de Bobo-Dioulasso, seconde ville du pays. L'Université Polytechnique de Bobo-Dioulasso est un Etablissement Public de l'Etat à caractère Scientifique, Culturel et Technique (EPSCT), chargé d'enseignement supérieur et de recherche scientifique. C'est une université nationale qui a pour mission l'élaboration et la transmission de la connaissance pour la formation des hommes et des femmes afin de répondre aux besoins de la Nation. Elle est composée de six (06) établissements et facultés :

- Ecole Supérieure d'Informatique (ESI) ;
- Institut du Développement Rural (IDR) ;
- Institut Universitaire de Technologie (IUT) ;
- Institut des Sciences de la Santé (INSSA) ;
- Unité de Formation et de Recherche en Sciences et Technologies (UFR/ST) ;
- Unité de Formation et de Recherche en Sciences Juridiques, Politiques, Economiques et de Gestion (UFR/SJPEG).

L'Ecole Supérieure d'Informatique où nous avons suivi notre formation, a une organisation pédagogique qui s'articule autour des deux (02) axes suivants :

- Le Cycle des Ingénieurs de Travaux Informatiques (CITI) options Analyse et Programmation (AP) et Réseau et Maintenance Informatiques (REMI) ;
- Le Cycle des Ingénieurs de Conception en Informatique (CICI) ;

La formation au Cycle des Ingénieurs de Conception en Informatique d'où nous sommes issus a pour objectif de former des cadres supérieurs opérationnels et évolutifs qui sont aptes à participer efficacement à l'analyse et à la conception (mise en œuvre) de solutions informatiques.

INTRODUCTION GENERALE

INTRODUCTION GENERALE

Dans un monde en perpétuelle évolution où les défis majeurs pour les organisations ont toujours été de s'adapter à leur environnement, l'informatique par définition étant la science du traitement rationnel de l'information se veut un canal garantissant cet état de fait.

En effet, les évolutions techniques et technologiques ont bouleversé nos habitudes et notre manière de travailler tant sur le plan du rendement que sur les aspects métiers. Aussi, sommes-nous tenté de dire que, de plus en plus, l'outil informatique est indispensable dans la plupart des activités humaines surtout au niveau des entreprises.

Dans cet élan de remplacement des tâches fastidieuses par la machine, un fait de l'heure, il s'agit de la question de la dématérialisation. Aussi, face à la rareté du papier et vue la question de la protection de l'environnement, les entreprises adoptent progressivement l'aspect écologique dans leur politique et procédures de gestion.

C'est ainsi que nous avons été sollicité par le Groupe SOFTNET, solutions informatiques, pour traiter sur le sujet relatif à la gestion des scanners dans un environnement Web avec comme thématique la problématique de l'intégration de scanner dans un environnement web. Il s'agit de pouvoir intégrer un module de scan dans une application Web cela en vue de faciliter la gestion du traitement des documents dans un processus de dématérialisation.

Dans ce qui suit, nous allons dans un premier temps faire une présentation de la problématique à l'étude avec une vue sur la structure d'accueil ; puis nous nous attèlerons successivement à la présentation des méthodes et des outils utilisés dans le cadre de l'étude, des acteurs à prendre en compte pour le présent projet ainsi que du planning de déroulement global ; s'en suivra ensuite une vue approfondie de la notion de scanner et de quelques aspects techniques liés aux scanners ; par la suite nous présenterons l'existant en matière de gestion de scanner, puis nous déclinerons notre

INTRODUCTION GENERALE

solution en vue de gérer le scan dans une environnement web et nous terminerons par une présentation du prototype.

CHAPITRE I. PRESENTATION GENERALE

L'objectif de ce chapitre est de poser les bases pour une étude ultérieure de la thématique soumise à notre étude. Il s'agira de présenter dans un premier temps la structure qui nous a donné l'opportunité de pouvoir mener l'étude ; puis nous présenterons le thème de notre étude ainsi que la problématique à résoudre.

I. Présentation de la structure d'accueil

1. SOFTNET-GROUP

Créé en 2003 SOFTNET GROUP a capitalisé une somme d'expériences et de compétences reconnues qui l'ont positionné comme l'un des acteurs majeurs des solutions intégrées IT en Afrique de l'Ouest et du Centre.

SOFTNET GROUP intervient principalement dans les domaines du Cloud computing, des Infrastructures Réseaux et Télécommunications de l'Ingénierie Logicielle, des Infrastructures Systèmes et Sécurité, des Services à Valeur Ajoutée ainsi que dans celui des Formations et du Consulting.

2. Vision

SOFTNET-GROUP participe à l'amélioration des performances et de la compétitivité des organisations publiques et des entreprises d'Afrique en offrant l'ensemble de ses capacités et des services pour accompagner l'essor des pays où elle opère afin de devenir l'un des acteurs majeurs de leurs dynamiques de croissance.

3. Organisation

SOFTNET, une entreprise dynamique au service de sa clientèle est composée ainsi que suit :

- la société mère basée à Ouagadougou ainsi que sa filiale du Burkina connues sous les noms respectifs de SOFTNET GROUP et SOFTNET BURKINA,
- et de trois (03) filiales qui sont SOFTNET COTE D'IVOIRE, SOFTNET MALI et SOFTNET NIGER.

CHAPITRE I. PRESENTATION GENERALE

Pour la filiale du Burkina, elle dispose de deux (02) représentations dont une à Ouagadougou qui fait office de siège du Burkina et du Groupe et l'autre à Bobo-Dioulasso.

Pour de plus amples performances et compte tenu des objectifs et des nouveaux défis qui s'offrent à elle, SOFTNET s'est doté d'un nouvel organigramme. Il s'agit de l'organisation en pôles de compétences. Pour ce faire on distingue quatre pôles de compétences. Le pôle Datacenter et Productivité ; le pôle Réseaux et Télécoms ; le pôle Formation et Support et le pôle Développement et Intégration Logiciel. C'est dans ce département que nous avons été accueillis pour ce présent stage.

4. Organigramme

Ci-contre l'organigramme de SOFTNET qui donne un aperçu global de son fonctionnement.

Voir Figure 01 à la page suivante

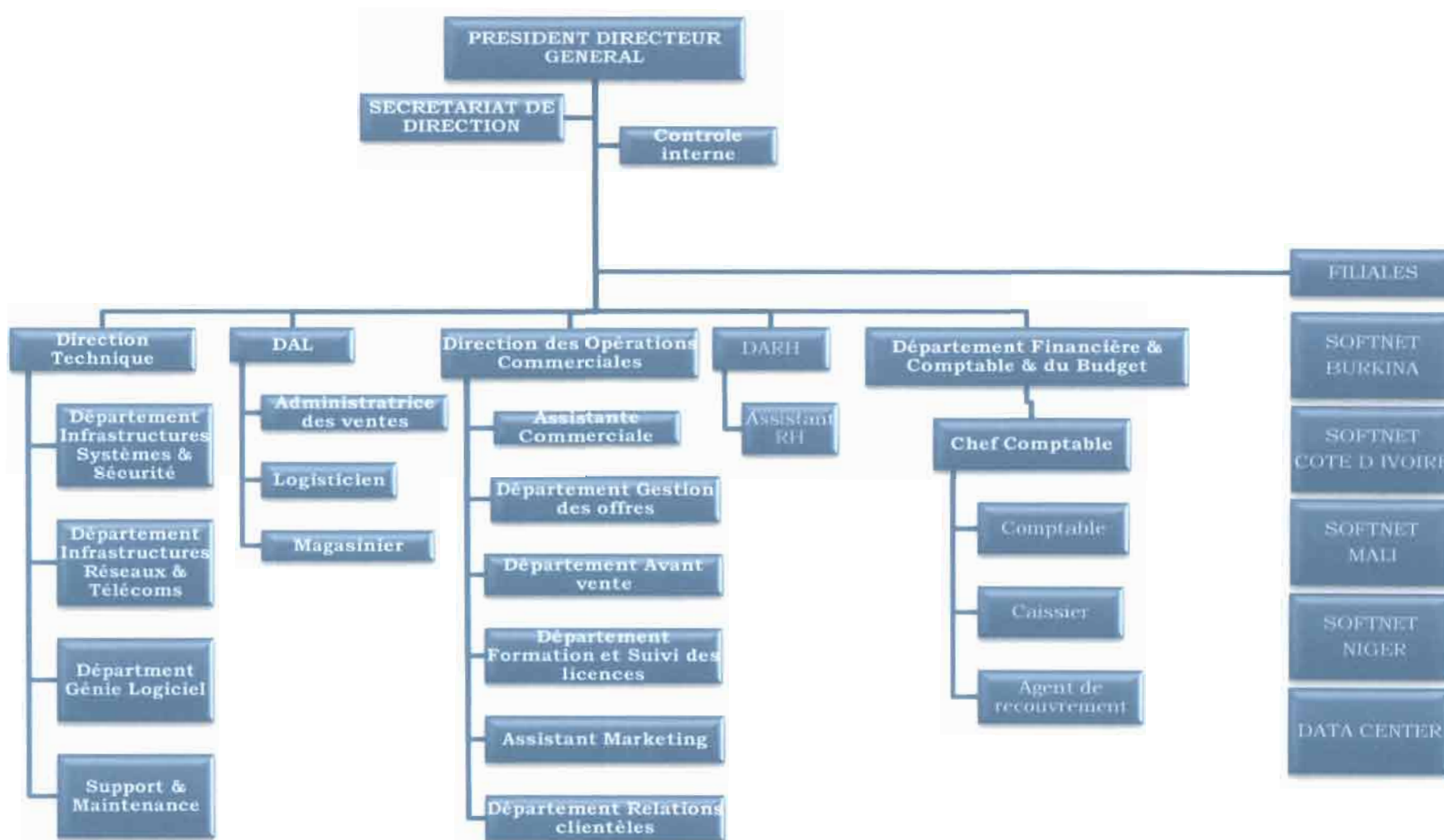


Figure 01 : Organigramme de SOFTNET

II. Présentation du thème

L'informatique est devenue aujourd'hui l'outil indispensable pour les entreprises tant du point de vue des résultats à obtenir que de la performance. Au-delà de ces aspects c'est aussi un outil au service de ces entreprises pour faciliter leur travail au quotidien.

La question de la dématérialisation est une question de l'heure. Pour les entreprises, il s'agit de mettre en place des applications qui sont à même de pouvoir résoudre l'épineuse problématique de pouvoir réduire la quantité des documents à produire et/ou à fournir lors des démarches pour les procédures dans les administrations ou les entreprises.

Il s'agit ici, de proposer une solution à même de donner la possibilité de pouvoir scanner un document en local et l'enregistrer sur un serveur distant et cela de façon transparente pour l'utilisateur avec la possibilité de pouvoir faire la reconnaissance des caractères scannés étant donné que les scanners sont pour la plupart conçus de manière à ce que le scan se fasse en local.

III. Problématique de l'étude

1. Description du projet

Le projet de gestion d'un scanner est né de la volonté de la société SOFTNET en réponse aux nombreuses sollicitations de ses partenaires pour leur doter d'une solution simple et efficace de gestion et de prise en charges de la gestion des scans. Cette solution devra être indépendante des modules déjà fonctionnels et surtout pourvoir au besoin suscité étant donné que les entreprises sont confrontées à un sérieux problème à ce niveau.

Au niveau par exemple des institutions financières, comment faire pour scanner et un chèque et l'injecter dans le circuit normal sans passer par l'étape de saisie du montant initial marqué par un stylo sur le chèque ?

Pour les administrations, comment scanner un document, l'enregistrer sur un serveur distant et utiliser les informations y afférentes ?

2. Objectifs

- **Général**
 - Simplifier la procédure de scan d'un document en diminuant le nombre d'étapes lors d'un processus de scan ;
- **Spécifiques**
 - Diminution du temps de traitement d'un processus de gestion automatisé basé sur la dématérialisation ;
 - Diminution de la quantité de papiers ;
 - Permettre de faire la reconnaissance des caractères saisis.
 - Faciliter l'interfaçage avec une application tierce.

3. Résultats attendus

- Gestion des scans à travers une interface web ;
- Gestion de l'enregistrement d'un document sur un serveur ;
- Utilisation de la fonction OCR (reconnaissance des caractères) ;
- Sauvegarde des données sur une Base de Données.

Ce chapitre nous a permis d'avoir un aperçu général sur le thème de notre étude qui s'intitule l'intégration d'un module de scan dans un environnement web. Il s'est agi de donner la problématique de l'étude qui s'il faut le rappeler est de permettre les scans de documents dans un environnement web.

CHAPITRE II. DEMARCHES POUR L'ETUDE

En vue de mener à bien une étude, il est convenable de donner les bases techniques sur lesquelles elle se fonde. Ainsi dans ce chapitre, nous présenterons les méthodes pour l'étude, les outils utilisés, les acteurs du projet ainsi que le planning de déroulement global ;

I. Méthodes d'étude et de conception

1. UML

UML est un langage de modélisation, utilisé pour la compréhension et la description des besoins des utilisateurs, la spécification et l'établissement d'une architecture logicielle d'un système. C'est le langage de modélisation le plus utilisé de nos jours.

Il est né de la fusion de trois (03) méthodes de référence : OMT, BOOCH, OOSE. Il n'est ni une méthode ni un processus.

UML n'impose pas une démarche particulière pour l'analyse d'un système mais préconise d'adopter une démarche ayant les caractéristiques suivantes:

- interactive et incrémentale ;
- guidée par le besoin des utilisateurs du système ;
- centrée sur l'architecture logicielle.

UML, dans sa version UML2 définit treize (13) diagrammes que l'on peut regrouper en trois (03) modèles qui sont :

- **Les modèles statiques:** Ces diagrammes permettent de visualiser, spécifier, de construire et documenter l'aspect statique ou structurel du système d'information. Il s'agit des diagrammes de classes, d'objets, de déploiement, de composants, de paquetage et de structure composite.
- **Les modèles dynamiques :** Ils modélisent les aspects dynamiques du système, c'est-à-dire les différents éléments qui sont susceptibles de subir des modifications. Parmi eux, on distingue les diagrammes de

CHAPITRE II. DEMARCHES POUR L'ETUDE

cas d'utilisation, d'états-transitions et d'activités.

- **Les modèles d'interactions** : Ils représentent les interactions entre le système lui-même et les différents acteurs du système, mais aussi la façon dont les différents objets contenus dans le système communiquent entre eux. Il s'agit des diagrammes de séquence, d'interaction globale et de temps.

Cependant, tous les diagrammes ne sont pas nécessaires et/ou indispensables pour la description d'un système logiciel donné.

En effet, de la même manière qu'il n'est pas judicieux d'utiliser tous les éléments de la langue française pour écrire un article, il n'est pas non plus obligatoire d'utiliser tous les concepts du langage UML dans un projet.

Ainsi dans le cadre de ce projet, nous nous limiterons à l'exploitation des diagrammes de classe, de séquence, de cas d'utilisation, d'état transition, de collaboration et d'activité.

Tableau 01 : Différents diagrammes utilisés dans le projet

Diagramme	Description
Classes	Exprime de manière générale la structure statique du système en termes de classes et de relations entre ces classes.
Cas d'utilisation	Décrit le comportement du système du point de vue de l'utilisateur. Permet de définir les limites du système et les relations entre le système et son environnement.
Séquence	Permet de représenter la hiérarchie des modules d'un projet.
Activité	Décrit les fonctionnalités d'un cas d'utilisation. Permet de bien comprendre les différentes activités réalisées et leur enchaînement dans le temps.
Etat-transition	Représente la façon dont évoluent (cycle de vie) les objets appartenant à une même classe. La modélisation du cycle de vie est essentielle pour représenter et mettre en forme la dynamique du système.

2. Modèle en V

La nécessité d'une application qui réponde aux besoins des utilisateurs définis le type de modèles à mettre en place pour l'atteinte des objectifs. Cela passe nécessairement par des modèles de gestion qui peuvent en tout temps subir des modifications et répondre au mieux aux besoins des utilisateurs. Pour notre présente étude, nous préconisons le cycle en V qui est un cycle de développement dont le principal atout reste sans doute la possibilité de faire des tests tout au long du cycle de développement en partant des tests unitaires jusqu'aux tests d'intégration. Pour notre part, ce seront les tests unitaires qui constitueront l'essentiel car l'application sera créée et gérée de façon autonome quoiqu'elle devrait pouvoir interagir avec des applications tierces existantes. Le modèle en V est représenté de façon imagée comme la figure ci-contre (cf. **Figure 02**)

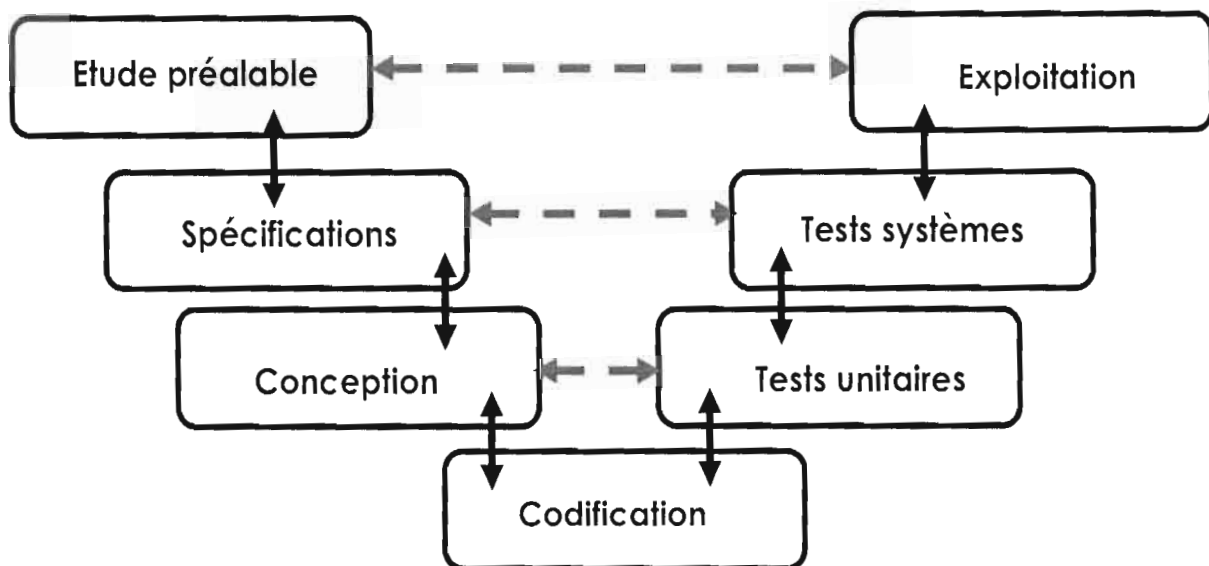


Figure 02 : Illustration du cycle en V

II. Outils utilisés

1. Environnement de développement intégré : Visual Studio 2012

CHAPITRE II. DEMARCHES POUR L'ETUDE

Visual Studio est un ensemble complet d'outils de développement permettant de générer des applications Web ASP.NET, des Services Web XML, des applications bureautiques et des applications mobiles. Visual Basic, Visual C++, Visual C# et Visual J# utilisent tous le même environnement de développement intégré (IDE : Integrated Development Environment), qui leur permet de partager des outils et facilite la création de solutions faisant appel à plusieurs langages. Par ailleurs, ces langages permettent de mieux tirer parti des fonctionnalités du Framework .NET, qui fournit un accès à des technologies clés simplifiant le développement d'applications Web ASP et de Services Web XML grâce à Visual Web Developer.

Microsoft Visual Studio est une suite de logiciels de développement pour Windows conçue par Microsoft. La dernière version s'appelle Visual Studio 2013.

Visual Studio 2012, introduit le développement d'applications sur l'environnement Windows RT. Le SDK Windows Phone 8.0 est également davantage mis en avant (s'il n'est pas inclus avec l'environnement, il est tout de même proposé au téléchargement). Il introduit la version 4.5 du .NET Framework et les versions compatibles de Windows sont Windows 7 et Windows 8 (et les déclinaisons serveur de ces éditions).

Le schéma ci-contre décrit le principe de fonctionnement de la plateforme .Net sur lequel est basé l'EDI Visual Studio. La plateforme comporte plusieurs couches les unes abstraites, les autres en code exécutable:

Voir Figure 03 à la page suivante

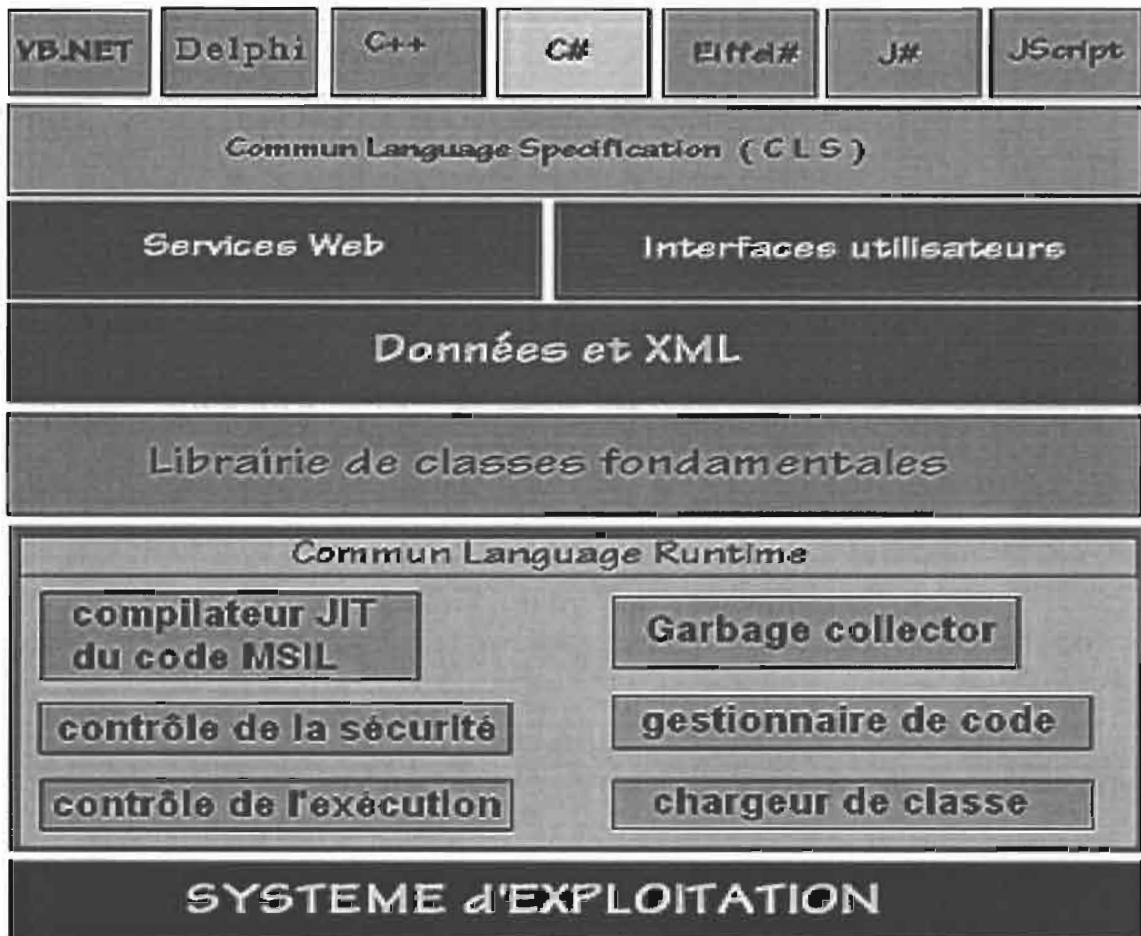


Figure 03 : Architecture de .Net Framework

- **La première couche CLS** (Common Language Spécification) est composée des spécifications communes à tous les langages qui veulent produire des applications **.NET** qui soient exécutables dans cet environnement et les langages eux-mêmes. Le CLS est une sorte de sous-ensemble minimal de spécifications autorisant une interopérabilité complète entre tous les langages de **.NET** les règles minimales (il y en a en fait 41) sont :
 - Les langages de **.NET** doivent savoir utiliser tous les composants du CLS
 - Les langages de **.NET** peuvent construire de nouvelles classes, de nouveaux composants conformes au CLS

- La **seconde couche** est un ensemble de composants graphiques qui permettent de construire des interfaces homme-machine orientées Web (services Web) ou bien orientées applications classiques avec IHM.
Les données sont accédées dans le cas des services Web à travers les protocoles qui sont des standards de l'industrie : HTTP, XML et SOAP.
- **La troisième couche** est constituée d'une vaste librairie de plusieurs centaines de classes.

Toutes ces classes sont accessibles telles quelles à tous les langages de .NET et cette librairie peut être étendue par adjonction de nouvelles classes. Cette librairie a la même fonction que la bibliothèque des classes de Java.

La librairie de classe de .NET Framework est organisée en nom d'espace hiérarchisé, exemple ci-dessous de quelques espaces de nom de la hiérarchie System.

- La **quatrième couche** forme l'environnement d'exécution commun (CLR : Common Language Runtime) de tous les programmes s'exécutant dans l'environnement .NET. Le CLR exécute un bytecode écrit dans un langage intermédiaire (MSIL : MicroSoft Intermediate Language).

2. Outils de modélisation : Microsoft Office Visio

Microsoft Visio (officiellement Microsoft Office Visio) est un logiciel de diagrammes et de synoptique pour Windows qui fait partie de la suite bureautique Microsoft Office mais se vend séparément. On peut ainsi créer des diagrammes de Gantt, des réseaux de PERT ou encore des diagrammes IDEFO. Dans Visio, les graphiques utilisés pour créer des diagrammes sont vectoriels.

Les versions Standard et Professionnel de l'édition 2007 partagent la même interface, mais cette dernière permet de faire des diagrammes plus avancés, grâce à des modèles supplémentaires. Cette version offre également une fonctionnalité unique qui permet aux utilisateurs de relier facilement

CHAPITRE II. DEMARCHES POUR L'ETUDE

leurs diagrammes à un grand nombre de sources de données et d'afficher les informations recueillies graphiquement.

Visio 5 est devenu le produit « Microsoft Visio 2000 » quand Microsoft a acquis la société Visio Corporation le 7 janvier 2000 pour 1,5 milliard de dollars.

Un plugin Visio est disponible permettant de seulement lire les documents depuis Internet Explorer.

3. Serveur de déploiement

Internet Information Services, communément appelé IIS (prononcé "2 i s"), est le logiciel de serveur services Web (ou FTP, SMTP, HTTP etc.) de la plateforme Windows NT.

Internet Information Services (IIS) 8 et 8.5 est le rôle Web Server (IIS) dans Windows Server® 2012 et Windows Server® 2012 R2.

Un serveur Web (appelé aussi Serveur HTTP) est un programme basé sur le modèle client/serveur et le protocole HyperText Transfer Protocol (HTTP).

Un serveur Web fournit les fichiers qui constituent les pages Web aux clients Web. Le rôle Serveur Web (IIS) offre une plateforme à la fois extensible, modulaire, sécurisée et simple d'emploi pour héberger de manière fiable des sites Web, des services et des applications Web.

Avec IIS, vous pouvez partager des informations avec des utilisateurs sur Internet, un intranet ou un extranet...

IIS est une plateforme Web unifiée qui intègre les services ASP.NET, FTP, PHP et Windows Communication Foundation (WCF).

IIS supporte les protocoles HTTP, HTTPS, FTP, FTPS, SMTP, ... Tout ordinateur sur Internet (ou dans un Intranet) qui contient un site dispose d'un programme Web.

III. Acteurs du projet

On désigne par acteur les personnes ressources à même d'impacter d'une manière ou d'une autre le projet. On distingue trois (03) groupes d'acteurs consignés dans le tableau ci-dessous (Cf. **Tableau 02**).

Tableau 02 : Acteurs du projet

Acteur	Rôles	Membres
Groupe de pilotage	Le groupe de pilotages prend les décisions relatives aux objectifs recherchés et fixe les orientations générales, les délais à respecter et définit également les moyens à mettre en place pour la réalisation du projet.	<ul style="list-style-type: none">- M. OUEDRAOGO Abdoul Karim, Directeur Technique/SOFTNET- M. KONDE Hippolyte Chef du pôle Développement et Intégration Logiciel/SOFTNET- Dr SOME Michel, enseignant-chercheur à l'ESI.
Groupe de projet	Le groupe de projet est chargé de l'exécution du projet, de la conception et la réalisation du projet.	M. TRAORE Lamine
Groupe des utilisateurs	Le groupe des utilisateurs à un rôle consultatif, il est chargé de fournir toutes les informations nécessaires à la bonne conduite du projet. Il intervient également dans la validation des dossiers d'étude produit par le groupe de projet et également pour les tests.	Tous les utilisateurs et utilisateurs potentiels

CHAPITRE II. DEMARCHES POUR L'ETUDE

IV. Planning de déroulement global

Tout projet et en particulier un projet informatique doit se dérouler dans le temps avec des objectifs clairement établis en vue de sa réalisation dans un certain délai. Aussi, cette section nous permet de fixer des délais de réalisation du projet avec en appoint les différentes tâches consignées.

N°	TACHES	DATE DEBUT	DATE FIN	DUREE	nov. 2015	déc. 2015	janv. 2016	févr. 2016
1	Phase d'insertion	03/11/2015	17/11/2015	2,2w	■			
2	Phase de recherche documentaire	03/11/2015	02/03/2016	17,4w	■	■	■	■
3	Analyse et conception	18/11/2015	16/12/2015	4,2w	■			
4	Realisation	16/12/2015	14/01/2016	4,4w		■		
5	Tests	16/12/2015	01/03/2016	11w		■	■	■
6	Exploitation et corrections des bugs	18/01/2016	02/03/2016	6,6w			■	■

Figure 04 : Planning prévisionnel

Dans ce chapitre, nous nous sommes focalisés sur la démarche considérée pour la présente étude ainsi que des différents outils et environnement utilisés. Il s'agit de fixer les bases techniques pour la conduite du projet et la réalisation technique.

CHAPITRE III. ETAT DE L'ART

L'état de l'art consiste à déterminer de façon temporelle ce qui a été mené dans un domaine de recherche précis avec si possible des références bibliographique et de l'internet qui étaye cet état de fait. Aussi, dans ce chapitre nous donnerons une vue globale pour ce qui est des scanners et de leur principe de fonctionnement et nous terminerons par la présentation de quelques solutions dans le domaine du scan dans un environnement web.

I. Historique

Le premier scanner est le bélinographe, inventé en 1908. On fixe le document à reproduire sur un cylindre tournant. Une cellule photoélectrique se déplace lentement sur un chariot mu par une vis solidaire du cylindre qui la fait avancer d'une fraction de millimètre par tour. Le bélinographe analyse ainsi le document ligne par ligne ((en) scan). Le signal électrique obtenu peut se transmettre par téléphone. À l'autre bout, le signal module une lumière qui impressionne un film photographique.

Les premiers scanners pour l'exploitation informatique de l'image, qui apparaissent dès les années 1960, fonctionnent selon le même principe.

Le scanner à plat, sur lequel on peut poser des livres et des cahiers sans séparer les feuilles, apparaît dans les années 1970. Les scanners professionnels resteront rotatifs pendant une dizaine d'années. Il est en effet plus facile alors de fabriquer une mécanique de précision, n'utilisant qu'une cellule dont les caractéristiques restent stables pendant la durée de l'analyse du document, que des ensembles de capteurs photo-électriques de caractéristiques identiques.

La politique de numérisation des bibliothèques et archives a entraîné la fabrication de scanners permettant la reproduction de livres sans les écraser sur une vitre, grâce à un système optique permettant de garder la ligne de capteurs à distance.

II. Définitions et concepts

Un scanner (anglicisme pour le mot français «numériseur») est un périphérique d'acquisition permettant de numériser des documents, c'est -à-dire de transformer un document papier en image numérique.

On distingue généralement trois (03) catégories de scanners :

- Les **scanners à plats** permettant de numériser un document en le plaçant à plat contre une vitre. Il s'agit du type de scanner le plus courant.
- Les **scanners à main** possédant une taille réduite. Ces scanners doivent être déplacés manuellement (ou semi manuellement) sur le document, par bandes successives, afin de le numériser en entier.
- Les **scanners par défilement** faisant défiler le document devant une fente lumineuse fixe afin de le numériser, à la manière des télécopieurs (fax). Ce type de scanner est de plus en plus intégré dans des appareils tels que des imprimantes multifonctions.

Il existe également des scanners capables de numériser des supports particuliers tels que les diapositives.

III. Principe de fonctionnement

Le principe du fonctionnement d'un scanner est le suivant :

- Le scanner parcourt le document ligne par ligne ;
- Chaque ligne est décomposée en «points élémentaires », correspondant à des pixels ;
- Un capteur analyse la couleur de chacun des pixels ;
La couleur de chaque pixel est décomposée selon trois (03) composantes (rouge, vert, bleu) ;

CHAPITRE III. ETAT DE L'ART

Chacune des composantes de couleur est mesurée et représentée par une valeur. Pour une quantification sur 08 bits, chacune des composantes aura une valeur comprise entre 0 et 255.

- Le document est posé contre une vitre. Sous cette vitre, un miroir et une source de lumière effectuent ensemble un passage. La source lumineuse éclaire le document au travers de la vitre. La lumière réfléchie par le document est renvoyée par le miroir mobile à un système optique qui le transmet à une batterie de capteurs de type CCD¹ (Charged Couple Device). L'ordinateur se charge ensuite de recomposer l'image obtenue.

Puisque la lumière traverse deux (02) fois la vitre, les images souffrent d'une légère diffusion de la lumière et d'une relative perte de luminosité. Il existe cependant d'excellents scanners à plat depuis la fin des années 1990.

En général, les numériseurs sont contrôlés par l'ordinateur via l'interface de programmation TWAIN ou WIA sous Windows ou SANE sous Linux.

La figure ci-contre illustre le fonctionnement d'un scanner :

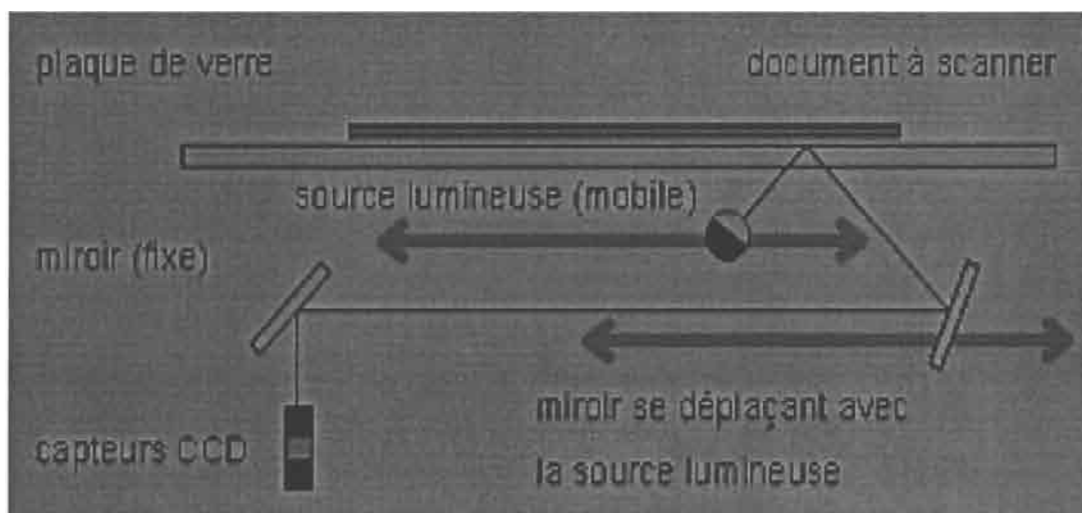


Figure 05 : Principe de fonctionnement d'un scanner

¹ Il existe aussi un autre type de capteurs : les **capteurs CMOS** (*Complementary Metal Oxyd Semi-conductor*, ou encore *MOS complémentaires*). On parle alors de technologie **CIS** (*Contact Image Sensor*). Ce type de dispositif utilise une rampe de LED (*Light Emitting Diode*) pour l'éclairage du document et requiert une distance très étroite entre les capteurs et le document. En contrepartie, la technologie CIS est bien moins consommatrice d'énergie.

IV. Quelques solutions de gestion des scanners

1. Atalasoft

Atalasoft est une entreprise spécialisée dans l'ECM d'imagerie qui fait sur le Web Document Imaging facile à ajouter dans toute entreprise à travers ses boîtes à outils .NET Developer.

Depuis 2002, Atalasoft fourni des bibliothèques d'imagerie pour aider les développeurs de logiciels à faire la numérisation, l'affichage et le traitement des documents dans leurs applications.

Atalasoft est également dans les technologies d'imagerie à zéro impact, ce qui permet aux applications d'être plus compétitives, plus faciles à utiliser et 100% sur le Web.

Pendant deux ans consécutifs, Atalasoft a été classé parmi les 100 sociétés de logiciels de croissance rapide.

Source : Atalasoft, Décembre 2015, <http://www.atalasoft.com>

2. Dynamsoft

Fondée en septembre 2003 dans le but d'être le centre dynamique de développeurs de logiciels, Dynamsoft fournit des logiciels de contrôle de version de classe entreprise, kits TWAIN de développement de logiciels (SDK) et autres SDK d'imagerie de documents, avec de nombreuses générations pour chaque produit. Aujourd'hui, de nombreuses entreprises Fortune 500, y compris HP, IBM, Intel et Siemens font confiance aux solutions Dynamsoft pour le contrôle de version et TWAIN développement SDK de balayage.

Pour rester en contact avec les dernières tendances de la technologie afin de fournir les meilleures solutions à ses clients, la société a été actif dans la communauté du logiciel en tant que membre associé du groupe de travail TWAIN et Gold Certified Partner de Microsoft.

Dynamsoft est situé à Vancouver, Canada et, notre mission est de continuer à évoluer nos produits logiciels pour faciliter le travail des développeurs de logiciels facile et agréable.

CHAPITRE III. ETAT DE L'ART

Dynamsoft comprend ce qu'il faut pour réussir dans un environnement commercial difficile technologique d'aujourd'hui. Avec l'expérience de l'industrie, des compétences et des solutions logicielles éprouvées, Dynamsoft Corporation offre à ses clients une grande valeur lorsque la productivité, la performance et le temps rapide sur le marché sont essentiels pour le succès.

A partir du premier SDK de numérisation TWAIN, Dynamsoft a élargi sa gamme de produits pour différents domaines :

- Web & .NET Imaging SDKs - TWAIN control, Barcode generator/reader...;
- Contrôle de version, à la fois sur site et des solutions hébergées...

Source : traduit par Google traduction (<http://www.translate.google.com/>).

Version originale (anglais)

Source : DynamSoft, Décembre 2015,

<http://www.dynamsoft.com/Company/AboutDynamsoft.aspx>

3. TwainX

TwainX est un projet open source de gestion de scan basé sur le projet Twain.

Ce projet Open source dont l'objectif est de faire progresser la recherche en matière d'OCR vise à rendre le scan indépendant du système d'exploitation et cela à travers l'utilisation des Web service mais aussi à rendre le traitement transparent pour l'utilisateur.

Source : TwainX, Décembre 2015, <http://twainx.sourceforge.net/index.html>

V. Intérêt de l'étude

Comme nous l'avons constaté dans l'historique (**Cf. CHAPITRE III, I**), l'utilisation des scanners remontent à plusieurs années.

CHAPITRE IV. ETUDE DE L'EXISTANT

- ☞ SCSI⁶ : Interface de prédilection pour le scanner à la fin des années 90, le standard SCSI est actuellement délaissé au profit du Firewire et de l'USB 2.0 ;
- ☞ Port parallèle : Lent par nature, ce type de connectique est de moins en moins utilisé et à éviter si l'ordinateur possède une des connectiques précédentes ;

⁶ *Small Computer System Interface* en anglais, est un standard définissant un bus informatique reliant un ordinateur à des périphériques ou à un autre ordinateur.

CHAPITRE V. ETUDE DU FUTUR SYSTEME

Après la phase de présentation de l'existant en matière de gestion du scanner, nous présentons dans cette section le futur système qui a pour but de donner les éléments techniques d'appréciation du système. Cela passe par les étapes de délimitation du projet des différents modèles (diagrammes) de l'architecture matériel et nous terminerons par le choix d'un scénario préalablement établi.

I. Délimitation du projet

Les limites de ce projet sont fixées par le fonctionnement même d'un scanner qui se résume en quatre (04) points essentiels :

- La phase d'initialisation du scanner ;
- La phase de scan à proprement parlé;
- L'enregistrement des informations dans un fichier sur un disque ;
- Et la lecture des informations stockées sur le fichier (fonction OCR).

II. Modélisation du système

L'objectif à atteindre lors de cette étude est de mieux percevoir les aspects fonctionnel et organisationnel du futur système; c'est-à-dire de comprendre le rôle de chaque acteur et la manière dont ils vont interagir pour atteindre les finalités du domaine d'étude. Pour les détails des diagrammes (voir Annexes (2) DIAGRAMME DE CAS D'UTILISATION, (3) DIAGRAMME DE SEQUENCES, (4) DIAGRAMME D'ACTIVITES, (5) DIAGRAMME DE CLASSES).

1. Diagramme de cas d'utilisation du système

a. Identification des cas d'utilisation

Après analyse les cas d'utilisations (CU) suivants ont été décelés :

CU01 : Initialisation

CU02 : Choisir un scanner

CU03 : Lancer le scan

CU04 : Lancer la fonction de scan multiple

CU05 : Enregistrer sur le serveur

CU06 : Reconnaître les caractères

CU07 : Sauvegarder sur une Base de Données

b. Cas d'utilisation du système

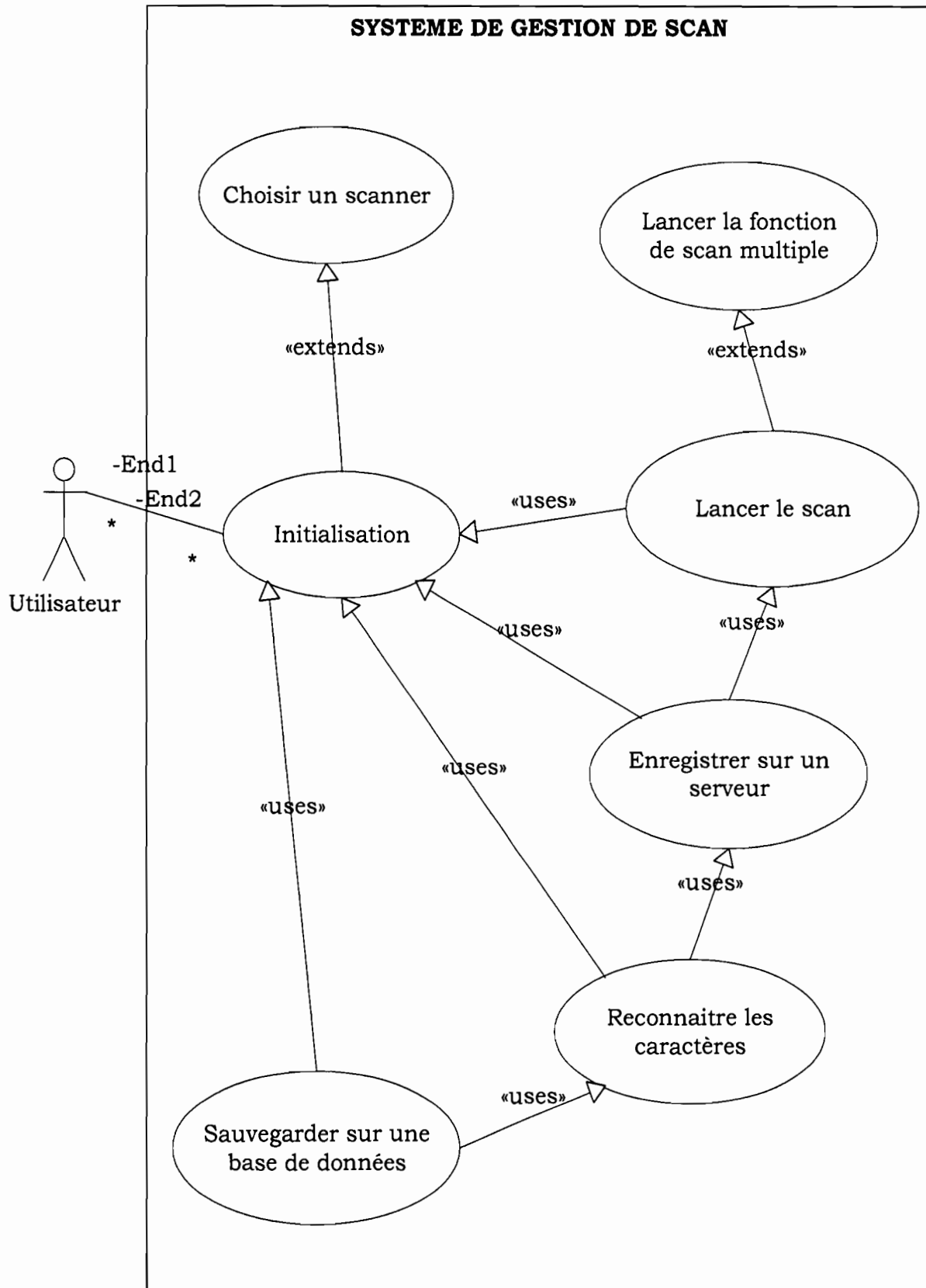


Figure 06 : Diagramme de cas d'utilisation du système

c. Description des cas d'utilisation

La description textuelle des cas d'utilisation permet de décrire de façon détaillée les cas d'utilisations en mettant l'accent sur les interactions entre le système et les utilisateurs dans les scénarii. Dans la description des cas d'utilisations on distingue 3 types de scénarii :

- Le scénario nominal qui montre un déroulement normal ;
- Le scénario alternatif qui est une variante du scénario normal ;
- Le scénario exceptionnel qui montre un déroulement anormal.

Ci-dessous, voici détaillé quelques cas d'utilisation décrits avec les scénarii nominal et alternatif :

Tableau 03 : Scénario nominal : Initialisation

Scénario nominal	Folio 1/2
CU1 : Initialisation	
Résumé : Ce cas permet d'initialiser le scanner. Il s'agit de sélectionner le scanner utilisé pour le scan.	
Acteurs :	
DESCRIPTION DU SCENARIO NOMIMAL <Début> 01 : L'utilisateur clique sur le bouton d'initialisation du scanner. 02 : Une fenêtre lui demande de choisir le scanner dans une liste de scanner 03 : L'utilisateur choisi le scanner souhaité et valide sa saisie. (A1) <Fin>	
Pré conditions : Le scanner est connecté à la machine et est alimenté	
Post conditions : Le scanner est choisi.	

CHAPITRE V. ETUDE DU FUTUR SYSTEME

Tableau 04 : Scénario alternatif : Initialisation

Scénario alternatif	Folio 2/2
CU1 : Initialisation	
DESCRIPTION DU SCENARIO ALTERNATIF	
A1 : Le scanner n'est pas répertorié dans la liste des scanners. On vérifie la disponibilité du scanner et on recommence l'étape 01	

Tableau 05 : Scénario nominal : Lancer un scan

Scénario nominal	Folio 1/2
CU3 : Lancer un scan	
Résumé : Il s'agit ici de lancer l'opération de scan qui permet de démarrer une action physique sur le scanner.	
Acteurs :	
DESCRIPTION DU SCENARIO NOMIMAL	
<Début>	
01 : L'utilisateur clique sur le bouton de scan du scanner.	
02 : Il lui est demandé de sélectionner des options supplémentaires. (A1)	
03 : Le scan peut commencer et se poursuivre jusqu'à sa fin. (A2)	
<Fin>	
Pré conditions : Le scanner à utiliser est sélectionné ;	
Post conditions : Le scan est terminé	

Tableau 06 : Scénario alternatif : Lancer un scan

Scénario alternatif	Folio 2/2
CU3 : Lancer un scan	
DESCRIPTION DU SCENARIO ALTERNATIF	
<Début>	
A1 : Il peut sauter cette étape en choisissant de continuer	
A2 : Au cas où le scan ne s'est pas terminé comme prévu, il revient à la phase de choix du clic sur le bouton du scanner	
<Fin>	

Tableau 07 : Scénario nominal : Enregistrer sur un serveur

Scénario nominal	Folio 2/2
CU5 : Enregistrer sur un serveur	
Résumé : Ce cas est transparent pour l'utilisateur. Au travers d'un service web, les informations scannées sont envoyées sur le serveur en vue d'y être stocké.	
Acteurs :	
DESCRIPTION DU SCENARIO NOMIMAL	
<Début>	
01 : Les informations sont encapsulées dans un format web (XML) pour être envoyées. (A1)	
02 : Les informations sont envoyées par la suite via un service web. (A2)	
03 : Les données arrivées sont stockées sur le serveur. (A3)	
<Fin>	
Pré conditions : Le scan s'est bien passé et les données sont disponibles.	
Post conditions : Les données sont stockées sur le serveur.	

CHAPITRE V. ETUDE DU FUTUR SYSTEME

Tableau 08 : Scénario alternatif : Enregistrer sur un serveur

Scénario alternatif	Folio 2/2
CU5 : Enregistrer sur un serveur	
DESCRIPTION DU SCENARIO ALTERNATIF	
A1 : L'encapsulation ne s'est pas bien passée, dans ce cas un message est affiché	
A2 : Le service web n'a pas pu aller jusqu'au bout, un message est affiché	
A3 : Le stockage ne s'est pas bien passé, un message est affiché	

Tableau 09 : Scénario nominal : Reconnaître les caractères

Scénario nominal	Folio 1/2
CU6 : Reconnaître les caractères	
Résumé : La reconnaissance des caractères est une fonction permettant à un scanner de pouvoir à partir d'un document scanné d'identifier un ensemble de caractères inscrits sur le document.	
Acteurs :	
DESCRIPTION DU SCENARIO NOMIMAL	
<Début>	
01 : Le système de reconnaissance initialise le document à reconnaître (E1)	
02 : L'image subi un prétraitement en vue de son analyse futur	
03 : L'image subi une analyse pour être stockée dans une catégorie	
04 : L'image est ensuite décomposée et reconnue (A1)	
<Fin>	
Pré conditions : Le document existe sur le scanner et respecte le formatage souhaité.	
Post conditions : Les caractères sont reconnus et identifiés.	

Tableau 10 : Scénario alternatif : Reconnaître les caractères

Scénario alternatif	Folio 2/2
CU6 : Reconnaître les caractères	
DESCRIPTION DU SCENARIO ALTERNATIF	
A1 : Les caractères sont renvoyée dans un modèle de lettre pour y subir une phase d'apprentissage avant la reconnaissance à proprement parlé.	

Tableau 11 : Scénario d'exception : Reconnaître les caractères

Scénario d'exception	Folio 3/3
CU6 : Reconnaître les caractères	
DESCRIPTION DU SCENARIO D'EXCEPTION	
E1 : Le document à reconnaître n'existe pas ou le chemin d'accès au fichier est introuvable. Un message d'erreur est affiché indiquant que le fichier est introuvable.	

2. Diagramme de classes

Le diagramme de classes représente le concept le plus important dans un développement orienté objet. Il représente la structure statique du système c'est à dire la partie du système qui sera toujours présente lors des interactions définies par les cas d'utilisation. Sur la branche fonctionnelle, ce diagramme est prévu pour développer la structure des petites entités connues des utilisateurs. En conception, ce diagramme représente la structure d'un objet, ou les modules du langage de développement. Il contient des classes et des associations. Ces dernières permettent de modéliser les entités du système. Les informations se rapportant à un domaine y sont structurées et regroupées dans des classes.

Pour ce module nous avons distingué essentiellement les classes suivantes en prenant en compte les documents de type titres de propriétés ou de jouissances et les documents de type pièces d'identités. Ce sont les classes :

CHAPITRE V. ETUDE DU FUTUR SYSTEME

Adresse, TypePiece, Piece, Usager, Civilite, Terrain, UsageTerrain, Localite, Arrondissement, CommuneDepartement, Province, Region, PiecePaye, qui est une extension de la classe Piece pour les pièces où on a un montant payé ou à encaisser.

CHAPITRE V. ETUDE DU FUTUR SYSTEME

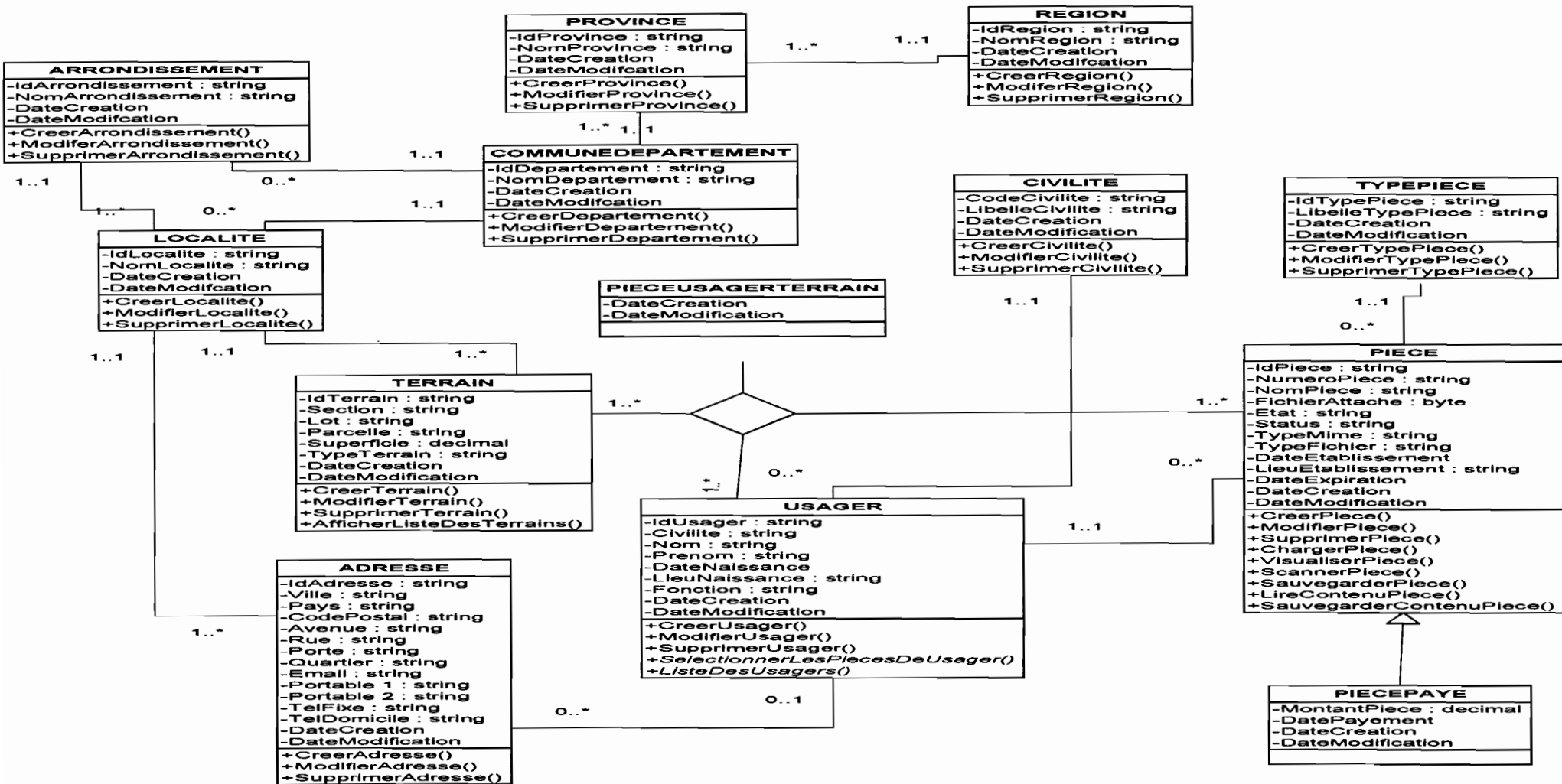


Figure 07 : Diagramme de classe du système

3. Diagrammes de séquence

Il s'agit dans le diagramme de séquence de décrire de façon chronologique les messages échangés entre les acteurs et le système. Pour cette section nous nous limiterons au cas d'utilisation « Lancer un scan » et «Reconnaitre les caractères».

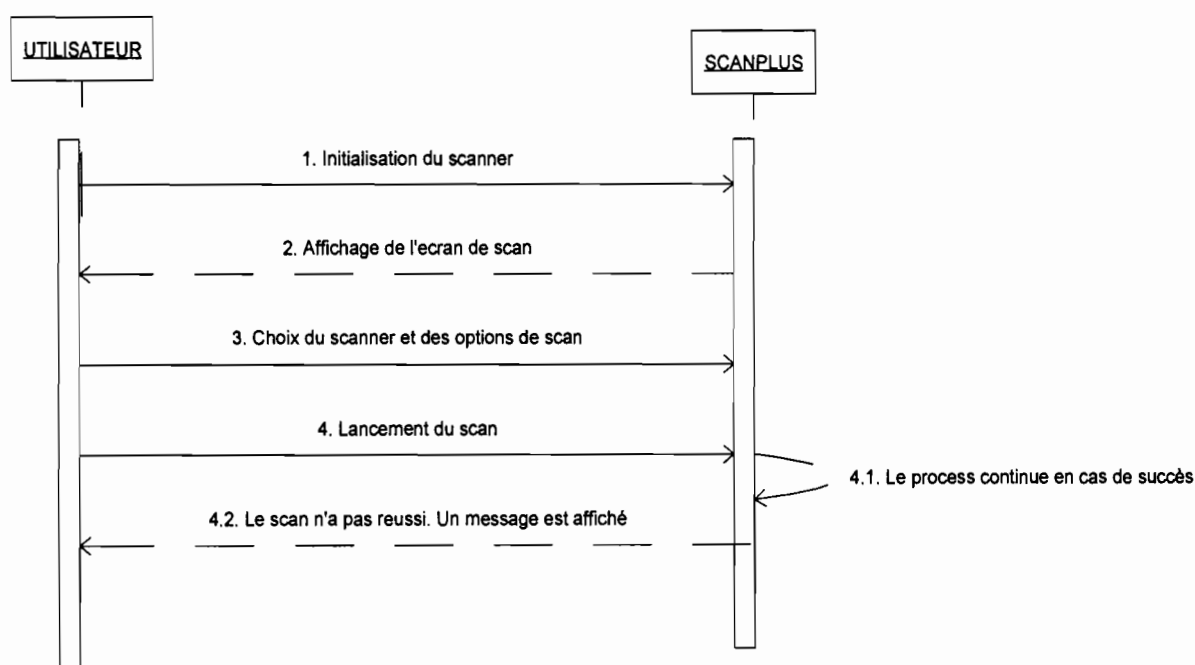


Figure 08 : Diagramme de séquence : Lancer un scan

Le diagramme de séquence « Reconnaitre les caractères » à lieu au niveau du serveur et fait suite à la séquence « Lancer un scan ».

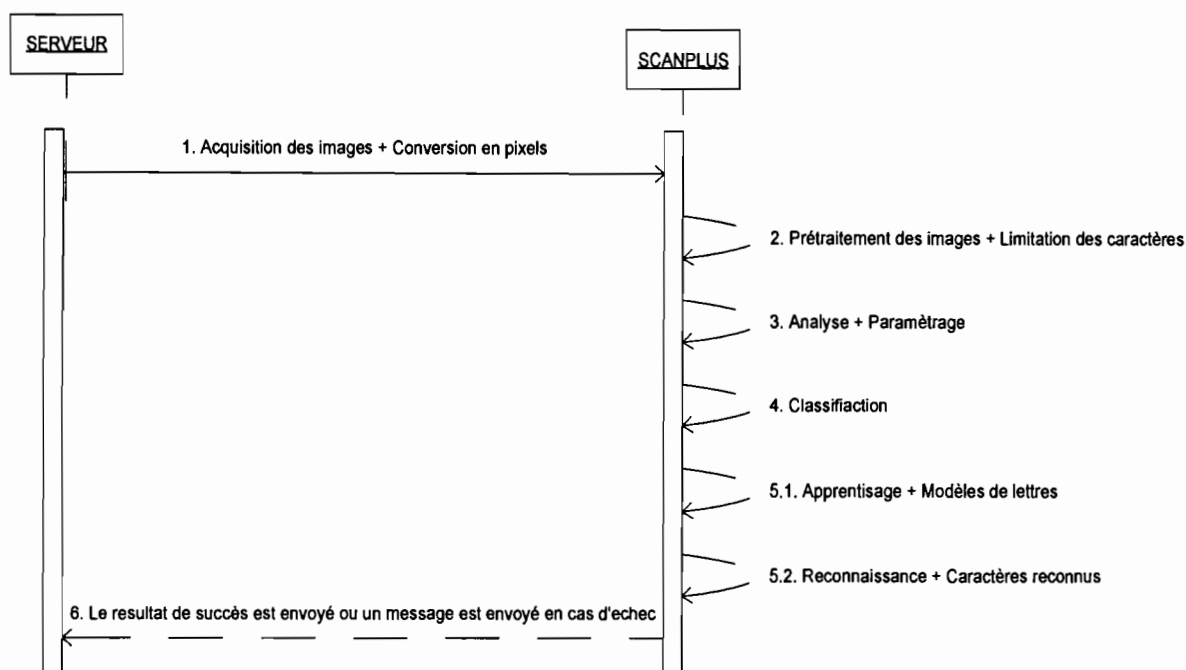


Figure 09 : Diagramme de séquence : Reconnaître les caractères

4. Diagrammes d'activités

Le diagramme d'activités décrit les fonctionnalités d'un cas d'utilisation. Il permet de bien comprendre les différentes activités réalisées et leur enchaînement dans le temps. Pour cette section nous nous limiterons au cas d'utilisation « Lancer un scan » et « Reconnaître les caractères ».

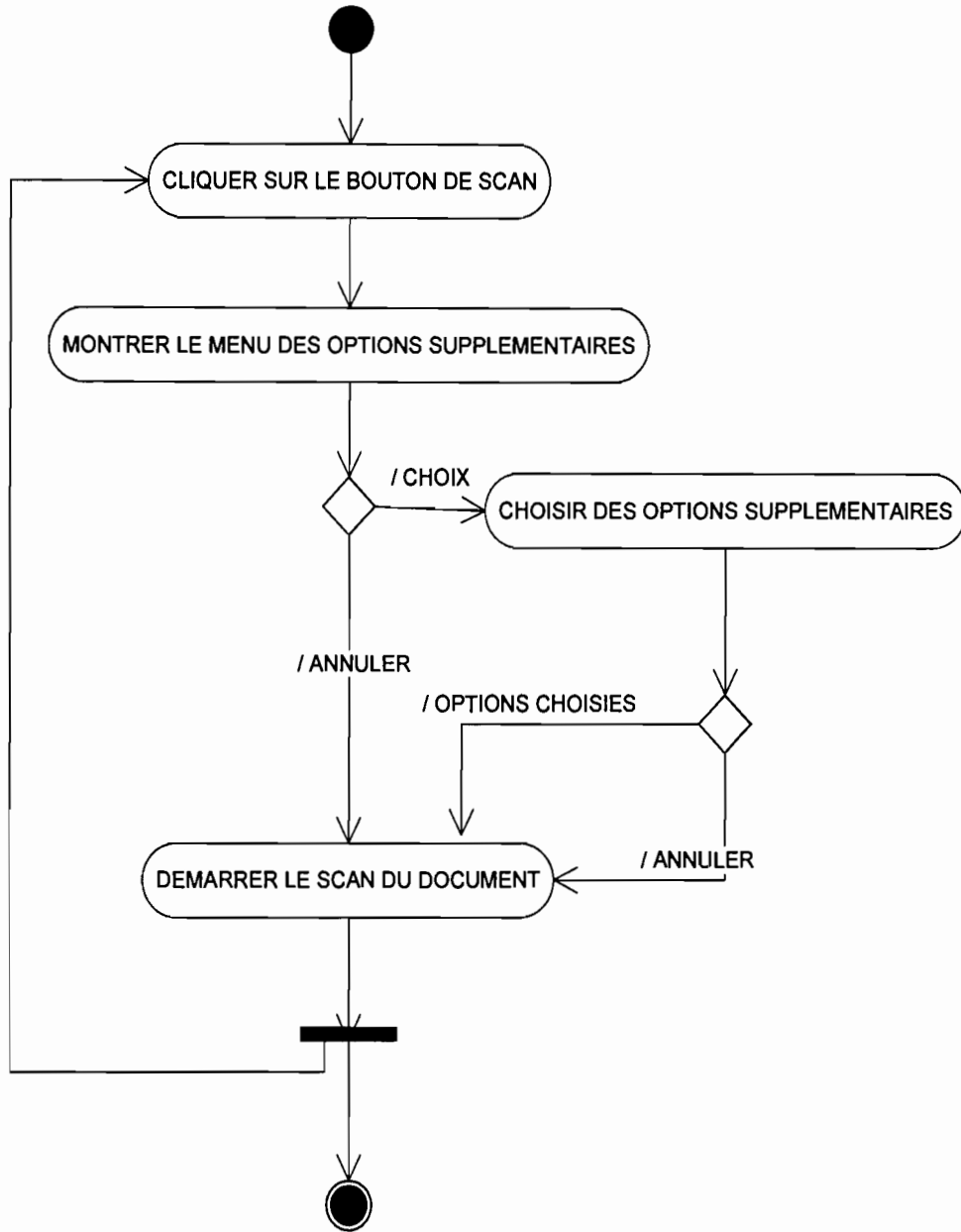


Figure 10 : Diagramme d'activité : Lancer scan

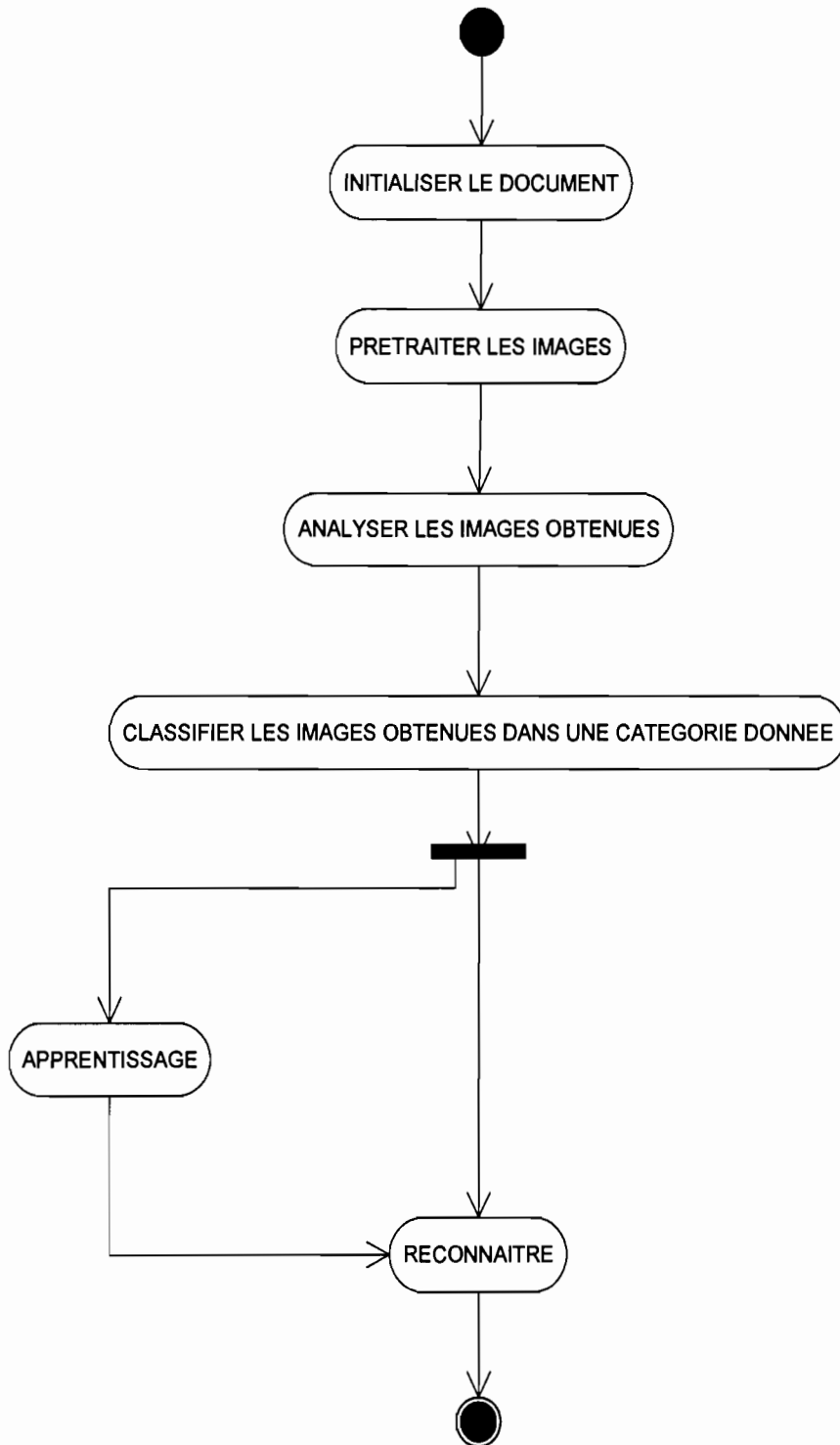


Figure 11 : Diagramme d'activité : Reconnaître les caractères

5. Diagramme d'état transition

Un diagramme d'état-transition décrit les différents états du système. Pour notre part, il s'agit de la gestion du scanner à proprement parlé et des différents étapes de la reconnaissance. Un document peut passer successivement par les étapes de encours de scan à scanné et continuer par

CHAPITRE V. ETUDE DU FUTUR SYSTEME

en cours de reconnaissance à reconnu ou passé directement en phase de reconnaissance si le document existe déjà. (Cf. Figure 17)

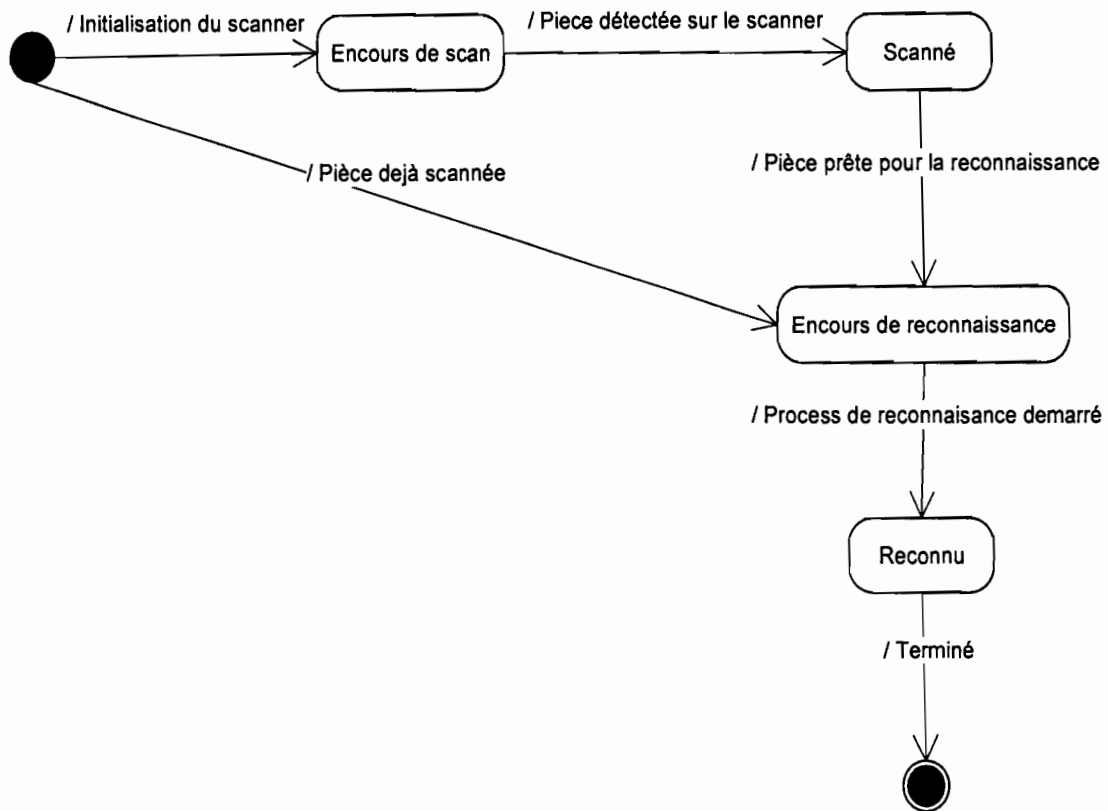


Figure 12 : Diagramme d'état-transition du système

III. Architecture réseau

L'architecture réseau décrit la logique dans lequel le réseau du système sera mis en place. Il permet donc de savoir comment de façon physique la composition de l'application. Il s'intègre à la logique applicative car il permet de s'assurer de la conformité de l'application dans son environnement d'exécution.

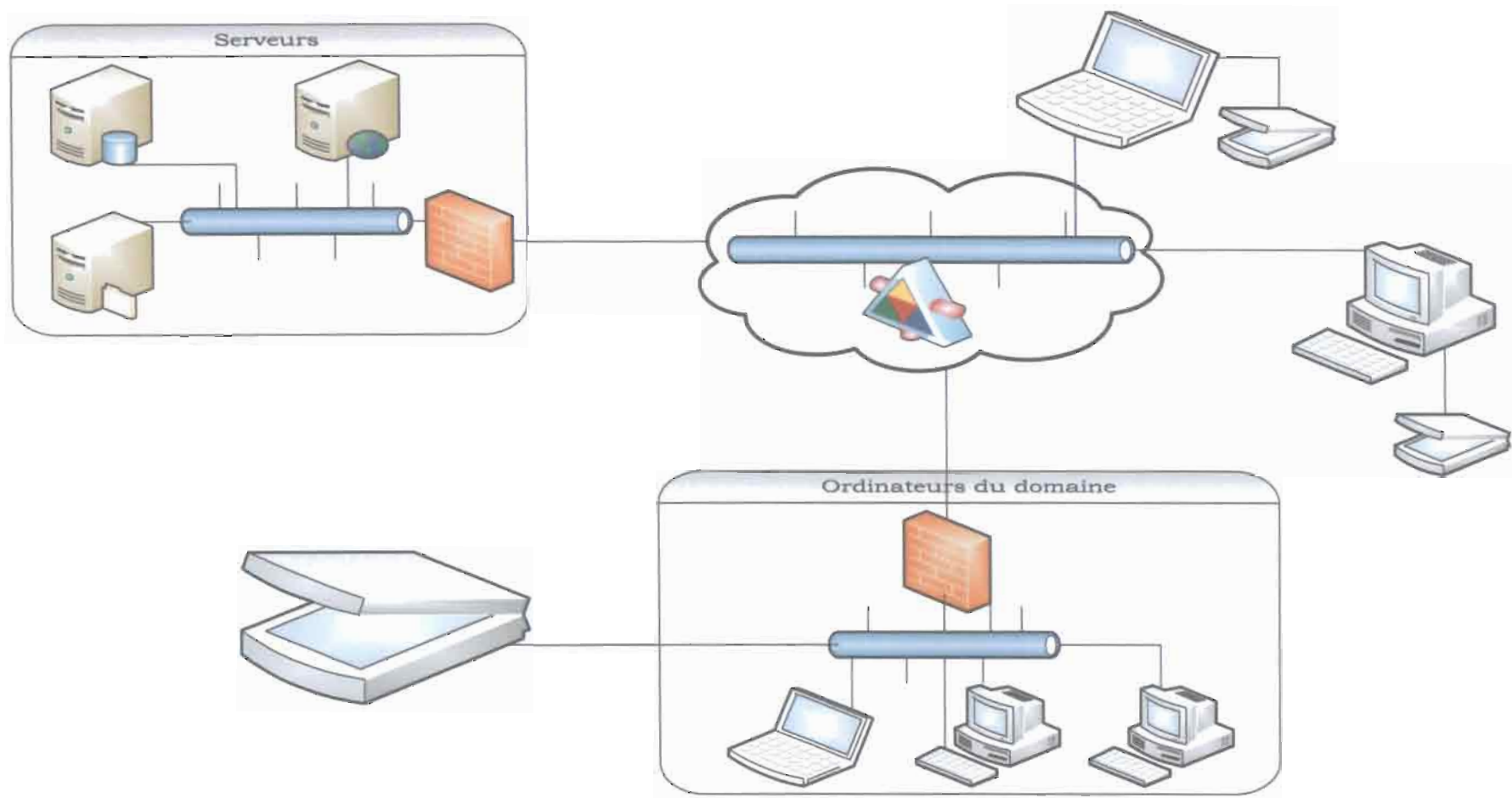












Figure 13 : Architecture réseau de l'application

CHAPITRE V. ETUDE DU FUTUR SYSTEME

Tableau 12 : Formalisme de l'architecture réseau

ELEMENT	DESCRIPTION
	Ordinateur client de l'application (Ordinateur de bureau)
	Ordinateur client de l'application (Ordinateur portable)
	Scanner utilisé par l'application pour scanné les documents dans le système.
	Serveur applicatif contenant les principaux traitements à réaliser sur la base de données.
	Serveur de Base de Données contenant l'ensemble des données.
	Serveur de fichier permettant de stocker les documents scannés et envoyé au système.
	Pare feu servant de protection pour les accès distants (ou local).
	Connexion par câbles Ethernet
	Service web utilisé pour l'appel distant des fonctions de scan du scanner.
	Utilisation d'un canal sécurisé pour les accès.

IV. Etude comparative des différents scenarii

Cette étape consiste à proposer les différents scenarii possible qui satisferont au mieux les objectifs visés par la mise en place du futur système. Il s'agit pour nous de spécifier les besoins en matériel et en logiciel, le coût de réalisation de chaque scénario, et éventuellement le scénario retenu pour la mise en œuvre du système futur.

1. Premier scenario : utilisation des modules natifs inclus dans les scanners

Il s'agit pour nous dans ce scenario d'utiliser les modules natifs déjà inclus dans les scanners et voir leur réutilisabilité dans la gestion des scans pour une application web. Pour ce scénario, il est à noter que les scanner pour la plupart sont conçus pour un fonctionnement en local.

2. Deuxième scénario : acquisition d'un module de gestion de scan

Ce scénario se base sur l'acquisition et la réadaptation d'un module déjà orienté web développé par une entreprise tierce. Dans les recherches, il nous est revenu que l'un des projets les plus avancés sur ce volet est le projet de la TWAIN Initiative, repris par plusieurs sociétés informatique. Ce projet est Open source (cf. **Chapitre VI.I.3**). Ci-dessous une étude comparative des quelques entreprises (non exhaustives) faisant ressortir leur forces et faiblesses dans ce domaine assortie du coût qui permet d'apprécier.

CHAPITRE V. ETUDE DU FUTUR SYSTEME

Tableau 13 : Etude comparative de quelques entreprises évoluant dans le domaine du scan utilisant une interface web

Nom de l'entreprise	Description	Avantages	Inconvénients
Atalasoftware	<p>Atalasoftware est une entreprise spécialisée dans l'ECM d'imagerie qui fait sur le Web Document Imaging facile à ajouter dans toute entreprise à travers ses boîtes à outils .NET Developer.</p> <p>http://www.atalasoftware.com</p>	<ul style="list-style-type: none"> - Expérience acquise dans le domaine - Solution orientée entreprise - Module disponible pour plusieurs environnements de développement 	<ul style="list-style-type: none"> - Coût (Plugin payant) 2 124 \$ ~ 1 100 000 F CFA par serveur de production
Dynamsoftware	<p>Fondée en septembre 2003 dans le but d'être le centre dynamique de développeurs de logiciels, Dynamsoftware fournit des logiciels de contrôle de version de classe entreprise, kits TWAIN de développement de logiciels (SDK) et autres SDK d'imagerie de documents, avec de nombreuses générations pour chaque produit. Aujourd'hui, de nombreuses entreprises, y compris HP, IBM, Intel et Siemens font confiance aux solutions Dynamsoftware pour le contrôle de version et TWAIN développement SDK de balayage.</p> <p>http://www.dynamsoftware.com</p>	<ul style="list-style-type: none"> - Expérience acquise dans le domaine - Solution orientée entreprise - Module disponible pour plusieurs environnements de développement 	<ul style="list-style-type: none"> - Coût (Plugin payant) 2 700 \$ ~ 1 400 000 F CFA par serveur de production
TwainX	<p>TwainX est un projet open Source de gestion de scan basé sur le projet Twain</p> <p>http://twainx.sourceforge.net/index.html</p>	<ul style="list-style-type: none"> - Open source 	<ul style="list-style-type: none"> - Nécessite du développement supplémentaire

3. Troisième scénario : développement du plugin et d'une interface applicative

La mise en place d'une solution faisant intervenir les aspects liés à l'OCR nécessite sans doute l'utilisation de plugin (ou contrôle active X selon le navigateur). Ce scénario consiste à développer un plugin incorporant un API web de gestion de l'interfaçage avec un serveur. Il s'agit plus explicitement de concevoir une interface web permettant d'atteindre un scanner en réseau. Cette solution présente l'avantage qu'elle est personnelle et personnalisable et qu'elle répond plus explicitement aux besoins exprimés dans le cahier de charges mais présente l'inconvénient qu'elle repose sur une approche temporelle. Autrement dit, cette solution nécessite plus de temps de développement.

4. Choix d'un scénario

Le choix d'un scénario repose sur des critères qui répondent au mieux aux exigences du cahier de charges. Il s'agit notamment de l'adaptabilité, de la portabilité, de la réutilisabilité et des coûts et temps de réalisation.

a. Description des différents critères de choix

- **L'adaptabilité** : aptitude à changer, à évoluer, flexibilité
- **L'interopérabilité**: l'interopérabilité est la capacité que possède un système, dont les interfaces sont intégralement connues, à fonctionner avec d'autres systèmes.
- **La réutilisabilité** : la réutilisabilité ou la réutilisation de code est l'action de reprendre un logiciel existant pour le réemployer, éventuellement en l'adaptant, dans un contexte présentant certaines similarités avec le programme d'origine. La réutilisation s'appuie fréquemment sur le concept de modularité. Par extension, ce terme désigne également l'ensemble des techniques informatiques proposées ou mises en œuvre pour faciliter cette réutilisation.
- **Le coût de réalisation** : le coût de réalisation se définit comme étant l'ensemble des acquisitions en matériel, en logiciel et bien sur des ressources

CHAPITRE V. ETUDE DU FUTUR SYSTEME

humaines employées pour la réalisation du projet. Ce critère est d'autant plus significatif qu'il permet d'évaluer qualitativement le projet.

- **Le temps de réalisation** : le temps est la durée de déroulement global du projet. Cette durée est un élément important pour évaluer un projet.

b. Etude comparative des scénarii

Ci-contre une étude comparative des différents scénarii.

Scénario	Adaptabilité	Interopérabilité	Réutilisabilité	Coût de réalisation	Temps de réalisation
Scénario 1	★	★★★	★	★★	★
Scénario 2	★★★	★★★★	★★★★	★★★★★	★★★
Scénario 3	★★★★	★★★★	★★★★★	★★★	★★★★

Tableau 14 : Etude comparative des scénarii

★ : Faible

★★ : Moyen

★★★ : Acceptable

★★★★ : Satisfaisant

★★★★★ : Élevé

c. Scénario retenu

Vu les critères liés à l'interopérabilité et à l'adaptabilité d'une part et d'autre part vu les exigences du cahier des charges (*cf.* Annexe (1)), notre choix s'est focalisé sur le scénario 3.

Ce chapitre nous a permis de décrire de façon schématique (à travers des modèles et diagrammes) le futur système ; ce qui nous a permis d'élaborer des scénarii et de faire un choix de réalisation du système.

CHAPITRE VI. REALISATION DU SYSTEME

C'est l'étape la plus importante de notre étude car nous allons présenter de façon concrète ce que nous avons pu réaliser. Dans un premier temps nous allons présenter de manière détaillée le choix techniques de réalisation après avoir fait une étude détaillée puis nous présenterons le prototype obtenu.

I. Etude détaillée

1. Modèle MVC

Le patron **Modèle-vue-contrôleur** (en abrégé **MVC**, de l'anglais *Model-View-Controller*), tout comme les patrons Modèle-vue-présentation ou Présentation, abstraction, contrôle, est un modèle destiné à répondre aux besoins des applications interactives en séparant les problématiques liées aux différents composants au sein de leur architecture respective. Ce paradigme regroupe les fonctions nécessaires en trois (03) catégories :

1. un modèle (modèle de données),
2. une vue (présentation, interface utilisateur)
3. un contrôleur (logique de contrôle, gestion des événements, synchronisation)

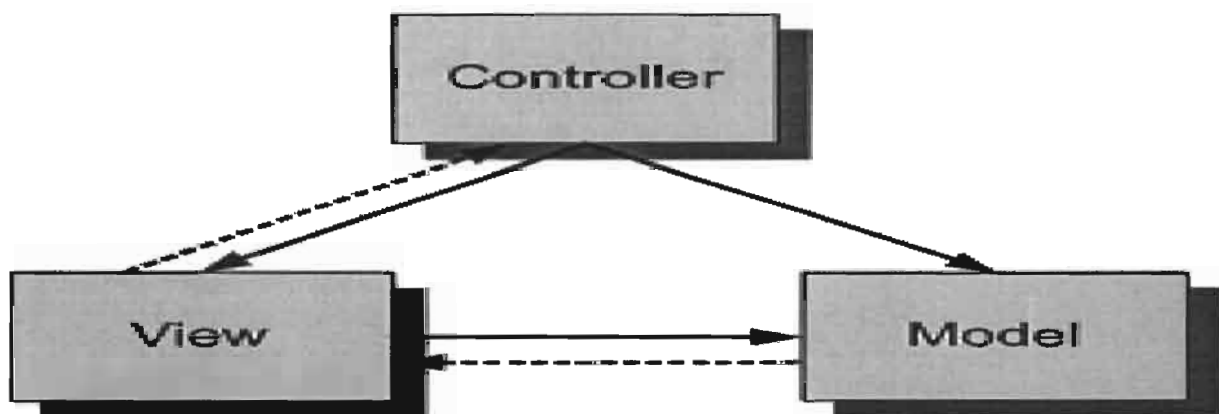


Figure 14 : Modèle-Vue-Contrôleur

Le schéma de cette figure résume les différentes interactions entre le modèle, la vue et le contrôleur. Les lignes pleines indiquent une dépendance (importation) tandis que les pointillés sont des événements

CHAPITRE VI. REALISATION DU SYSTEME

2. Projet de scan universel : TWAIN Initiative

Twain est créé par le petit groupe de compagnies software et hardware comme réponse au besoin de spécification proposée par l'industrie d'imagerie. Le but du groupe a été de fournir une solution multiplateforme et ouvertes pour interconnecter les différents outils d'acquisition avec les applications. Le groupe original est représenté par les cinq (05) compagnies : Aldus, Caere, Eastman Kodak, Hewlett-Packard, et Logitech.

Trois (03) autres compagnies, Adobe, Howtek, et Software Architects ont aussi contribuées significativement.

Le design de TWain a commencé en janvier, 1991. Une revue du ToolKit a été faite par les développeurs d'Avril 1991 à Janvier 1992. La coalition TWAIN inclut approximativement 300 individus représentant 200 compagnies pour influencer et guider le développement de TWain.

La version courante de TWain est écrite par les 11 membres courants du Groupe du Travail TWain. Les membres inclus: Adobe, Canon, Eastman Kodak Company, Fujitsu Computer Products of America, Genoa Technology, Inc., Hewlett-Packard Company, Intel Corporation, J.F.L. Peripherals, Kofax Image Products, Ricoh Corporation, et Xerox.

En Mai 1998, un agrément est annoncée entre Microsoft et le Groupe de Travail TWain ce qui fournit l'inclusion de TWain Data Source Manager en Microsoft Windows 98 et Microsoft Windows NT 5.0.

3. Reconnaissance des caractères

La reconnaissance de l'écriture relève du domaine de la reconnaissance des formes qui s'intéresse aux formes de caractères. Le but est d'attribuer à une forme un identifiant des prototypes de référence déterminés préalablement.

La reconnaissance optique de caractères (OCR) est une opération informatique rapide permettant de réaliser la transformation d'un texte écrit sur papier en un texte sous forme d'un fichier informatique en représentation

symbolique (par exemple pour les écritures latines, le codage opéré est le code ASCII (American Standard code for information interchange), tandis que pour l'arabe on utilise généralement le code ASMO (Arabie Standard Metrology Organisations).

Pour la mise en œuvre de la reconnaissance des caractères, plusieurs algorithmes ont été mis en place.

4. Service REST

REST (REpresentational State Transfer) est un style d'architecture, pour les systèmes hypermédia distribués, pour concevoir un service web, utilisant au maximum les possibilités de HTTP. Ses principales caractéristiques sont les suivantes :

- **application client/serveur** : le transport sur le réseau est assuré par HTTP (essence même d'un web service)
- **interface uniforme** : tout élément offert à la manipulation par l'application est nommé ressource et est identifié de manière unique par les Identifiants de Ressource Uniforme (URI ci-après) suivant le schéma : http URL = "http:" "//" host [":" port]

Deux (02) types de schéma d'URI sont distingués :

- **URI member** qui désigne une seule ressource.
- **URI collection** qui désigne une liste de ressources de même type.
- **la sémantique** des messages du client au serveur est **celle de HTTP** :
 - **GET URI** : récupérer la représentation d'une ressource (URI member), ou d'une liste de ressources (URI collection)
 - **POST URI** (collection) : rajouter une ressource à une liste de ressources existantes, donc création de ressource.
 - **PUT URI** (member) : modifier une ressource existante ou créer une nouvelle ressource.

CHAPITRE VI. REALISATION DU SYSTEME

- **DELETE URI** : destruction d'une (URI member) ou plusieurs (URI collection) ressources.
- **les réponses** du serveur aux clients utilisent également les messages HTTP, en particulier les codes d'erreur suivants :
 - 200 Ok
 - 404 Ressource non trouvée sur le serveur
- **représentation des ressources** : le format avec lequel une ressource est représentée est obtenu par négociation grâce aux entêtes HTTP. Le client indique ce qu'il souhaite comme format de représentation avec l'entête "Accept", le client et le serveur utilisent l'entête "Content-type" pour décrire le format de représentation de la ressource incluse dans le message. Les formats les plus fréquemment rencontrés sont XML et JSON.

II. Choix techniques de réalisation

1. Architecture logique

Pour ce projet, l'architecture retenue est une architecture à trois (03) niveaux i.e. une architecture basée sur le concept du 3-tiers. Dans ce type d'architecture, lorsque qu'un client demande une ressource, cette ressource est transmise au serveur Web via le réseau, le serveur réalise les traitements nécessaire et transmet à son tour au serveur de BD qui lui est chargé de renvoyer les données nécessaires au traitement ; enfin le résultat de la requête est transmis au client en réponse à la ressource demandée. (Cf. **Figure 15** ci-contre).

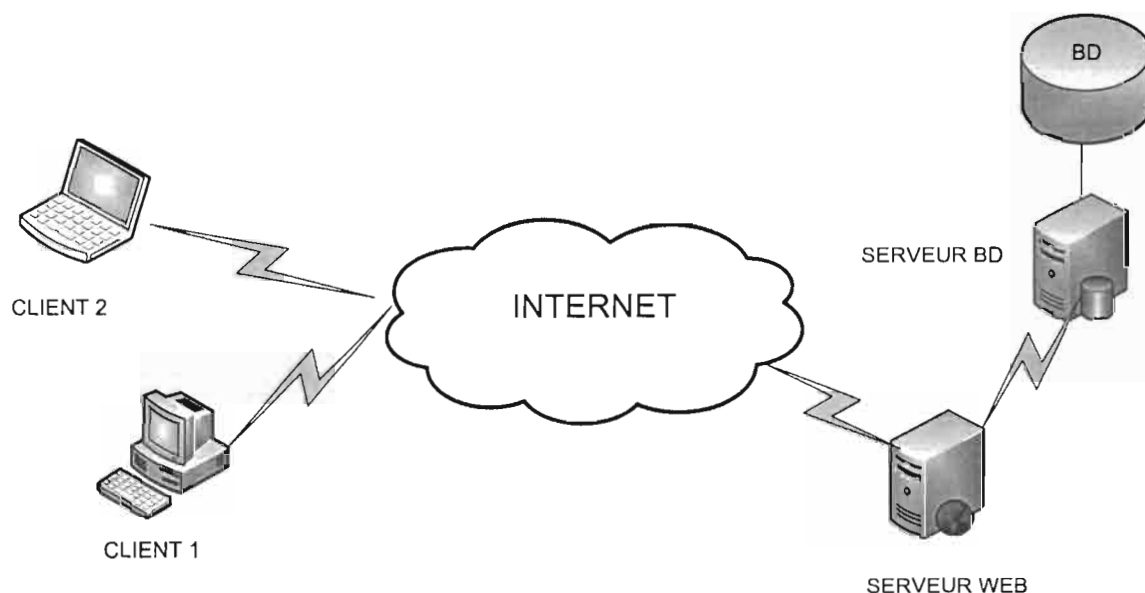


Figure 15 : Architecture logique

2. ASP.Net MVC

ASP.NET est un modèle de développement Web unifié qui offre les services dont le développeur a besoin pour générer des applications Web d'entreprise avec un minimum de codage.

Les applications peuvent être créées dans tout langage compatible avec le Common Language Runtime (CLR), comme Microsoft Visual Basic et C#. (Cf. **Chap. II. II.1. Environnement de développement intégré : Visual Studio 2012**)

CHAPITRE VI. REALISATION DU SYSTEME

Ces langages permettent de développer des applications ASP.NET qui bénéficient du Common Language Runtime, de la sécurité de type, de l'héritage, etc. Les pages Web ASP.NET sont intégralement orientées objet.

L'infrastructure des pages et des contrôles ASP.NET permet d'encapsuler les fonctionnalités courantes d'interface utilisateur dans des contrôles réutilisables et faciles à utiliser.

Cette infrastructure fournit également des fonctionnalités de contrôle de l'apparence globale de l'application ou site web via des thèmes et des apparences. On a la possibilité de les définir soi-même.

Tout le code ASP.NET est compilé, ce qui permet, entre autres avantages, le typage fort, l'optimisation des performances et la liaison anticipée. Une fois le code compilé, le CLR compile ensuite le code ASP.NET en code natif, ce qui améliore les performances.

ASP.NET contient un compilateur qui compile tous les composants d'application, y compris les pages et les contrôles, dans un assembly que l'environnement d'hébergement ASP.NET peut utiliser ensuite pour répondre aux demandes des utilisateurs.

ASP.NET MVC aide les développeurs Web à générer des applications Web standard faciles à maintenir, dans la mesure où il diminue la dépendance dans les couches d'application à l'aide du modèle MVC (Model-View-Controller). Il fournit également le contrôle complet sur le balisage de page. ASP.NET MVC améliore la testabilité en prenant en charge le développement axé sur des tests (TDD, Test Driven Development).

3. Base de Données : MS SQL Server

Microsoft SQL Server est un système de gestion de base de données (abrégé en SGBD ou SGBDR pour « Système de Gestion de Base de Données relationnelles ») développé et commercialisé par la société Microsoft.

Bien qu'il ait été initialement Co développé par Sybase et Microsoft, Ashton-Tate a également été associé à sa première version, sortie en 1989.

CHAPITRE VI. REALISATION DU SYSTEME

Cette version est sortie sur les plates-formes Unix et OS/2. Depuis, Microsoft a porté ce système de base de données sous Windows et il est désormais uniquement pris en charge par ce système.

Lors de sa création, Sybase SQL Server hérite des principes du moteur Ingres développé à l'origine par l'université de Berkeley.

En 1994, le partenariat entre les deux (02) sociétés ayant été rompu, Microsoft a sorti la version 6.0 puis 6.5 seul, sur la plate-forme Windows NT.

Microsoft a continué de commercialiser le moteur de base de données sous le nom de SQL Server.

Tandis que Sybase, pour éviter toute confusion, a renommé Sybase SQL Server en Sybase Adaptive Server Enterprise.

Microsoft SQL Server fait désormais partie de la stratégie technique de Microsoft en matière de base de données. Le moteur MSDE, qui est la base de SQL Server, doit à terme remplacer le moteur Jet (celui qui gère les bases Access) dans les applications telles qu'Exchange et Active Directory.

La version 2005 de SQL Server est sortie le 3 novembre 2005 en même temps que Visual Studio 2005. La prise en charge de Windows Vista et de Windows Server 2008 n'a été ajoutée qu'à partir du Service Pack 2 (SP2). Actuellement le Service Pack 3 est disponible. L'optimiseur a été entièrement refait, tout comme le moteur relationnel.

La version 2008 de SQL Server (nom de code Katmai) est disponible depuis août 2008. Elle est actuellement au niveau de service pack 3. Elle est disponible en 9 langues, dont le français. La version 2012 de SQL Server est disponible depuis avril 2012.

Lors du développement de la version 2005, le projet était à l'époque l'un des plus grands projets informatiques du monde. Plus de 1 600 développeurs ont participé au codage du noyau et l'on estime que plus de 10 000 autres personnes y ont travaillé de près ou de loin (Interactions homme-machine, documentation, traduction...).

III. Mise en œuvre

1. Outils et Technologies

a. Tesseract

Tesseract est un logiciel de reconnaissance optique de caractères sous licence Apache.

Conçu par les ingénieurs de Hewlett Packard de 1985 à 1995, son développement est abandonné pendant les dix (10) années suivantes ; en 2005, les sources du logiciel sont libérées sous licence Apache et le logiciel est actuellement développé par Google. Initialement limité aux caractères ASCII, il supporte parfaitement les caractères UTF-8 et reconnaît maintenant quarante (40) langues.

Pour l'installation et la configuration de Tesseract se référer à l'adresse sur une machine dans un environnement Windows (ici sur une machine tournant sur Windows 8)

Source : Tesseract, Installation de Tesseract sur une machine utilisant Windows 8, Décembre 2015, <http://emop.tamu.edu/Installing-Tesseract-Windows8>.

b. Bibliothèque Twain

La bibliothèque Twain est une bibliothèque open source disponible sur le site officiel de la Twain Initiative. C'est une bibliothèque codée en C++ et qui permet de façon élémentaire d'accéder aux fonctions de scan des scanners. Pour pouvoir l'utiliser, il peut être téléchargé et compiler sous forme de dll⁷ et on peut définir ses propres méthodes pour l'utiliser. Pour plus de détail voir sur le site officiel du projet⁸ ou aller sur le lien de téléchargement de l'API⁹.

c. Web API (.Net)

⁷ **dll** : Dynamic Link Library (en français, bibliothèque de liens dynamiques) est une bibliothèque logicielle dont les fonctions sont chargées en mémoire par un programme, au besoin, lors de son exécution, par opposition aux bibliothèques logicielles statiques ou partagées dont les fonctions sont chargés en mémoire avant le début de l'exécution du programme.

⁸ Le code n'est pas disponible directement sur le site mais on a des informations sur le concept et les liens potentiels où on peut obtenir des informations : <http://www.twain.org/>

⁹ Disponible pour plusieurs langages de programmation mais ici c'est le code compatible C# : (<http://sourceforge.net/projects/twainforcsharp/?source=directory>)

CHAPITRE VI. REALISATION DU SYSTEME

Une interface de programmation (souvent désignée par le terme API pour Application Programming Interface) est un ensemble normalisé de classes, de méthodes ou de fonctions qui sert de façade par laquelle un logiciel offre des services à d'autres logiciels. Elle est offerte par une bibliothèque logicielle ou un service web, le plus souvent accompagnée d'une description qui spécifie comment des programmes consommateurs peuvent se servir des fonctionnalités du programme fournisseur.

L'objectif est de fournir une porte d'accès à une fonctionnalité en cachant les détails de la mise en œuvre. Une interface de programmation comporte typiquement des classes, des méthodes ou des fonctions, des types de données et des constantes.

Une interface de programmation est typiquement mise en œuvre par une bibliothèque logicielle qui fournit une solution à un problème informatique en faisant abstraction de son fonctionnement.

Les Web API sont conçus selon le principe des Service Rest (Cf. **Chap. IV. II.1. Service REST**).

Le support d'ASP.NET Web API dans le framework ASP.NET MVC permet de créer facilement des services Web RestFull qui peuvent être consommés à partir d'un large éventail de clients (réseaux sociaux, navigateurs, terminaux mobiles, etc.). Web API fournit les fonctionnalités suivantes :

- **un modèle de programmation HTTP moderne** : l'API Web permet d'accéder directement et manipuler les requêtes et réponses HTTP à l'aide d'un nouveau modèle objet HTTP fortement typé. Le même modèle de programmation HTTP est pratiquement disponible sur le client par l'intermédiaire de HttpClient, pouvant être utilisé pour appeler des API Web dans n'importe quelle application .NET ;
- **négociation de contenu** : le client et le serveur peuvent travailler ensemble afin de déterminer le bon format pour les données retournées à partir d'une API. Les formats pris en charge par défaut sont XML et JSON. Le développeur pourra élargir ce

soutien en ajoutant ses propres formats, ou même en remplaçant la stratégie de négociation de contenu par défaut ;

- **prise en charge complète des routes** : l'API Web prend désormais en charge l'ensemble des fonctionnalités de route qui ont toujours été une partie de la pile Web, y compris les paramètres d'itinéraire et les contraintes. En outre, la cartographie des actions possède un support complet pour les conventions, de sorte que le développeur n'aura plus besoin d'appliquer des attributs à une classe ou méthode. La configuration de Web API se fait uniquement à travers le code ;
- **composition de requêtes** : Web API permet de prendre rapidement en charge l'interrogation par l'intermédiaire des conventions URL OData. Lorsque le développeur retourne un type IQueryable de son API Web, le framework fournira un soutien pour OData, rendant facile l'implémentation de la pagination et le tri ;
- **validation des modèles** : le modèle fournit un moyen facile d'extraire des données de diverses parties d'une requête HTTP et convertit celles-ci en objets .NET, qui peuvent être utilisés par des actions Web API. Web API offre le même support qu'ASP.NET MVC pour la liaison des modèles et la même infrastructure de validation ;
- **filtres** : l'API Web permet de créer et utiliser facilement des filtres, qui permettent d'encapsuler et d'appliquer divers comportements
- **améliorations des tests** : plutôt que de fixer des détails dans les objets de contexte HTTP statiques, les actions Web API peuvent maintenant travailler avec les instances de HttpResponseMessage et HttpResponseMessage, deux (02) nouveaux objets HTTP qui permettent d'effectuer plus facilement des tests ;
- **support de l'IoC** : Web API prend en charge le modèle de service de localisation mis en œuvre par ASP.NET MVC, qui permet de résoudre les dépendances pour de nombreux dispositifs différents. Le développeur peut facilement intégrer cela avec un

conteneur IoC, ou une bibliothèque permettant l'injection de dépendances ;

- **flexibilité lors de l'hébergement** : Web API peut être déployée dans n'importe quel type d'application ASP.NET (y compris à la fois ASP.NET MVC et ASP.NET Web Forms). Web API peut également être hébergé au sein de votre plateforme, si vous ne voulez pas utiliser ASP.NET/IIS pour le faire. Cela vous donne un maximum de flexibilité dans la façon dont vous l'utilisez.

Source : À la découverte d'ASP.NET Web API, Janvier 2015,
<http://rdonfack.developpez.com/tutoriels/dotnet/decouverte-asp-net-web-api>

2. Résultats obtenus

L'intégration logicielle est une tâche qui ne se fait pas toujours sans quelques difficultés. Des difficultés liés à l'environnement de production voire de déploiement. La question du scan dans un environnement web est un sujet actuel tant et si bien que la documentation en la matière n'est pas toujours facile à obtenir. Cependant, nous nous sommes évertués à trouver des réponses à nos questions.

Aussi, une question qui revient pour toutes les personnes qui s'intéresse à la problématique des OCR reste sans doute le problème de la reconnaissance des caractères accentués qui est une question importante surtout dans nos environnements francophones. Ainsi au cours des tests sur l'OCR, le moteur de l'OCR (Tesseract **Cf.** section précédente) initialement conçu pour reconnaître des caractères non accentués (ASCII) quoique reconnaissant le format UTF-8 ne se comporte pas toujours comme il se doit même si cependant les résultats obtenus sont encourageant.

Un autre élément à prendre en compte est la qualité de l'image à reconnaître. En effet, nous avons remarqué que certains formats sont plus faciles à reconnaître que d'autres (par exemple les *tiff*, les *jpeg* de qualité moyenne i.e. moins de 400x400, les *png* donnent de meilleurs résultats). Pour

les images en noir et blanc, la conversion donne un meilleur résultat que les images en couleur. (Voir Annexes (6), (7))

3. Présentation du prototype

Il s'agit de la présentation d'une version que l'on pourrait qualifier de maquette. En effet, à la date de finalisation du rapport, nous avons à une version qui répond aux besoins des tests. Les tests déjà effectués augurent d'une avancée significative. Nous avons ci-contre une présentation basique de ce que nous avons baptisé ScanPlus. ScanPlus comme spécifié dans le cahier de charges doit pouvoir interagir avec une application tierce avec la possibilité de faire de l'OCR. Pour l'heure les fonctions de Scan sont les plus abouties. Pour ce qui est de l'OCR des tests sont en cours en vue d'avoir un meilleur rendu du texte obtenu (gestion de la casse, de la culture locale de la machine, ...). Ci-contre une présentation du prototype avec une séquence pour l'aboutissement d'un scan complet avec sauvegarde du fichier sur le serveur.



Figure 16 : Ecran 1 : Page d'accueil pour le démarrage du scan

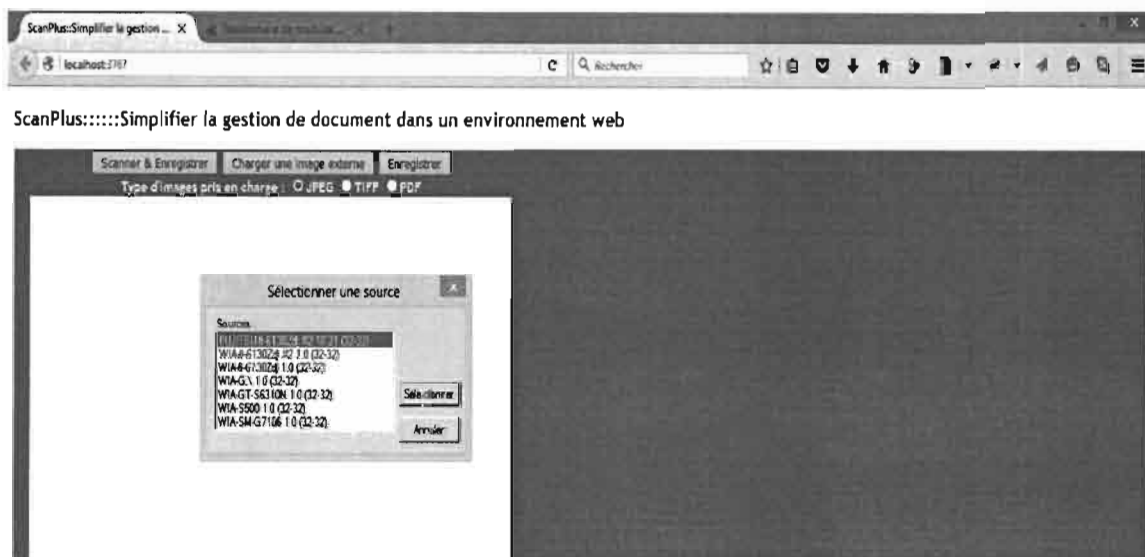


Figure 17 : Ecran 2 : Page de sélection du scanner

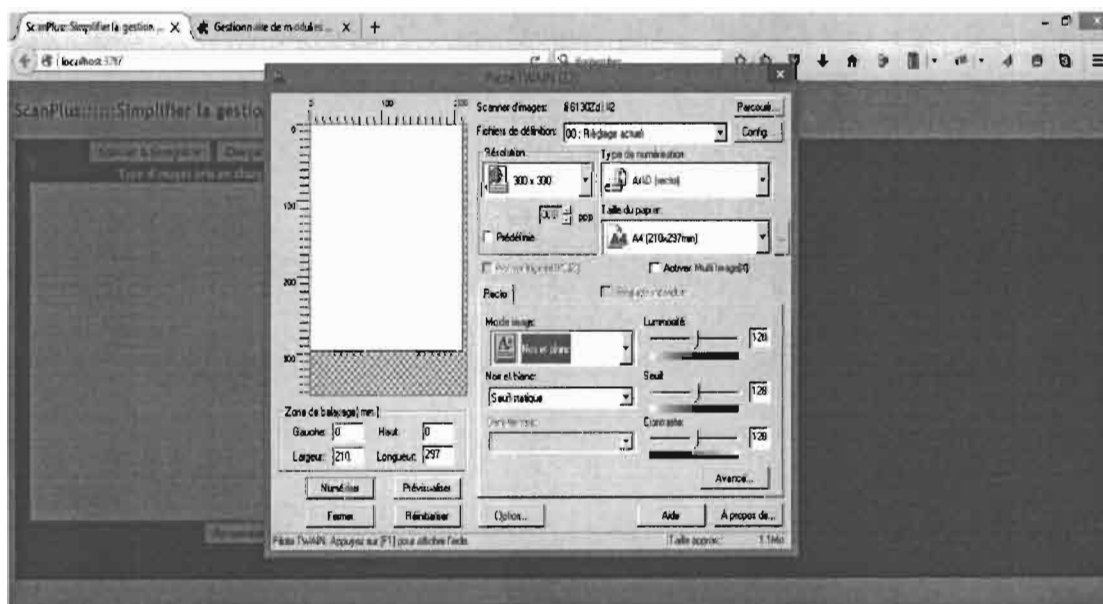


Figure 18 : Ecran 3 : Page d'affichage des options de numérisation

CHAPITRE VI. REALISATION DU SYSTEME



Figure 19 : Ecran 4 : Page d'enregistrement sur un serveur

Dans cette section nous avons présenté les résultats que nous avons pu obtenir en vue de donner une vie à notre étude. Il s'est agi d'abord de donner les concepts clés pour la réalisation puis de donner une vue globale du système à travers les prototypes.

CONCLUSION GENERALE ET PERSPECTIVES

Les systèmes d'information offrent de nouvelles perspectives pour les entreprises tant sur le plan de l'organisation du travail que la qualité du service. De cette étude, il ressort que la gestion des documents papier dans un environnement où l'ordinateur est utilisé ne se fait pas sans quelques changements, notamment sur les processus et procédures de ces entreprises. Ainsi, pouvoir gérer des documents devient une problématique qui intéresse les acteurs des SI dont l'objectif fondamental demeure de donner une nouvelle façon de travailler et ainsi d'aboutir à de meilleurs rendements. Les résultats de cette étude sont bénéfiques non seulement pour les entreprises mais aussi pour les professionnelles des TIC qui se voient doter d'un outil de prise en compte de l'une de leurs préoccupations majeures qui est la question du scan dans un environnement web.

Au terme de cette étude, il ressort que la gestion des scanners est plus que d'actualité. En effet, différentes solutions ont été proposées pour sa mise en œuvre aussi diverses les unes que les autres. Nonobstant quelques difficultés, il ressort que le thème choisi pour cette présente étude propose le double avantage de donner aux entreprises une solution de gestion des documents axée sur la diminution de la quantité de papier et de pouvoir s'intégrer sans problème dans une solution déjà existantes (utilisation d'un web service).

BIBLIOGRAPHIE

- Service Rest, P. Gambarotto (INPT/Enseeiht), Octobre 2010, <https://www.projet-plume.org/ressource/les-services-web-de-type-rest>
- Tesseract, Novembre 2015, <https://code.google.com/p/tesseract-ocr>
- Utilisation de Tesseract dans Visual Studio 2012, Charles W., Janvier 2016, <http://www.github.com/charlesw/tesseract/>
- Installation et configuration de Tesseract logiciel de reconnaissance de caractère, <http://emop.tamu.edu/Installing-Tesseract-Windows8>.
- Exemple de code pour l'utilisation du scan web, <http://www.dynamsoft.com/Downloads/WebTWAIN-Sample-Download.aspx>
- DynamSoft, Décembre 2015, <http://www.dynamsoft.com>
- Atalasoftware, Décembre 2015, <http://www.atalasoftware.com>
- TwainX, Décembre 2015, <http://twainx.sourceforge.net/index.html>
- Wikipedia, Open source, Décembre 2015, https://fr.wikipedia.org/wiki/Open_source
- Système de reconnaissance hors-ligne des mots manuscrits arabe pour multi-scripteurs, Riadh BOUSLIMI, Janvier 2016, http://www.memoireonline.com/09/09/2694/m_Systeme-de-reconnaissance-hors-ligne-des-mots-manuscrits-arabe-pour-multi-scripteurs0.html
- Bibliothèque Twain, Twain Project code for C-SHARP C#, Janvier 2016, <http://sourceforge.net/projects/twainforcsharp/?source=directory>
- Twain Initiative, Janvier 2016, <http://www.twain.org/>
- À la découverte d'ASP.NET Web API, 20 septembre 2012, <http://rdonfack.developpez.com/tutoriels/dotnet/decouverte-asp-net-web-api>

ANNEXE

ANNEXE

- (1) CAHIER DE CHARGES
- (2) DIAGRAMME DE CAS D'UTILISATION
- (3) DIAGRAMME DE SEQUENCES
- (4) DIAGRAMME D'ACTIVITES
- (5) DIAGRAMME DE CLASSES
- (6) ATTESTATION PROVISOIRE SCANNEE
- (7) ATTESTATION PROVISOIRE ANALYSE OCR

ANNEXE (1)

Thème : « *Intégration d'un module de scan dans une application Web* »

I. Présentation du thème

L'informatique est devenue aujourd'hui l'outil indispensable pour les entreprises tant du point de vue des résultats à obtenir que de la performance. Au delà de ces aspects c'est aussi un outil au service de ces entreprises pour leur faciliter leur travail quotidien.

La question de la dématérialisation est une question de l'heure. Pour les entreprises, il s'agit de mettre en place des applications qui sont à même de pouvoir résoudre l'épineuse problématique de pouvoir réduire la quantité des documents à produire et/ou à fournir lors des démarches pour les procédures dans les administrations ou les entreprises.

Il s'agit ici, de proposer une solution à même de donner la possibilité de pouvoir scanner un document en local et l'enregistrer sur un serveur distant et cela de façon transparente pour l'utilisateur avec la possibilité de pouvoir faire la reconnaissance des caractères scannés. Les scanners sont pour la plupart conçus de manière à ce que le scan se fasse en local.

II. Problématique de l'étude

1. Description du projet

Le projet de gestion d'un scanner est né de la volonté de la société SOFTNET en réponse aux nombreuses sollicitations de ses partenaires pour leur doter d'une solution simple et efficace de gestion d'un module de gestion et de prise en charges de la gestion des scans. Cette solution devra être indépendante des modules déjà fonctionnels et surtout pouvoir au besoin cité. Les entreprises sont confrontées à un sérieux problème à ce niveau.

Au niveau par exemple des institutions financières, comment faire pour scanner et un chèque et l'injecter dans le circuit normal sans passer par l'étape de saisie du montant initial marqué par un stylo sur le chèque ?

Pour les administrations, comment scanner un document, l'enregistrer sur

un serveur distant et utiliser les informations y afférentes ?

2. Objectifs

☒ Généraux

- ✓ Simplifier la procédure de scan d'un document en diminuant le nombre d'étapes lors d'un processus de scan.
- ✓ Faciliter l'interfaçage avec une application tierce.

☒ Spécifiques

- ✓ Augmentation du temps de traitement d'un processus de gestion automatisé basé sur la dématérialisation.
- ✓ Diminution de la quantité de papier
- ✓ Permettre de faire la reconnaissance des caractères saisis ;

3. Résultats attendus

- ☒ Gestion des scans à travers une interface web ;
- ☒ Gestion de l'enregistrement d'un document sur un serveur ;
- ☒ Utilisation de la fonction OCR (reconnaissance des caractères) ;
- ☒ Sauvegarde des données sur une Base de Données.

III. Outils et langages

- ☒ EDI : Visual studio 2012
- ☒ Langage de contrôle du scanner : C/C++
- ☒ Langage de programmation : ASP.Net MVC 4
- ☒ DAO : Entity framework
- ☒ BD : SQL Server
- ☒ Scanner utilisant le projet TWAIN

IV. BESOINS

Les besoins se définissent comme l'ensemble des ressources humaines, financières, matérielles et logicielles nécessaire pour la tenue du projet. Ces ressources doivent être définies dès le début du projet pour prendre en

compte l'ensemble des besoins afin de pouvoir conduire jusqu'à terme le projet.

Pour ce projet, les besoins peuvent se résumer au travers de ce tableau :

Intitulé	Nature	Nombre
Ressources humaines	Ingénieurs	01
Ressources matérielles	Ordinateurs	01
	Scanner	01
Ressources logicielles	Base de données	1 licence
	EDI*	1 licence

NB: Pour les besoins financiers, se reporter à la section suivante.

V. BUDGET PREVISIONNEL

Le cout de réalisation se déduit des couts en matériel, en logiciel et bien sur des ressources humaines employées pour la réalisation du projet. Ce cout est évidemment prévisionnel et donc peut se retrouver modifié au cours du projet soit en hausse ou à la baisse avec une marge de tolérance de plus ou moins 10%.

Toute modification doit faire l'assentiment du groupe de projet

Désignation	Quantité	Prix Unitaire (F CFA HT)	Total (F CFA HT)
Scanner	1	788 700 (1195 €)	788 700 (1195 €)
Cout du développement	1	200 000	200 000
Formation des utilisateurs	3	15 000	45 000
COUT TOTAL (HT)			1 033 700

ANNEXE (2)

Annexe

Diagramme des cas d'utilisation

DIAGRAMME DE CAS D'UTILISATION

Objectif : Le diagramme de cas d'utilisation représente les cas d'utilisation identifiés et l'acteur associé à chacun.

Formalisme



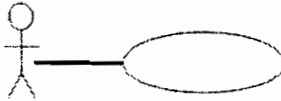
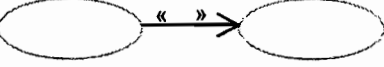
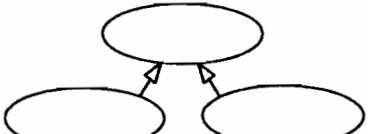
```

    graph TD
      A1[Acteur 1] --- CU1((nom Cas d'utilisation))
      CU2((nom Cas d'utilisation)) -.->|« Étend »| CU1
      CU3((nom Cas d'utilisation)) -.->|« Inclus »| CU1
      CU4((nom Cas d'utilisation)) --> CU3
      CU5((nom Cas d'utilisation)) --> CU3
      A2[Acteur 2] --- CU5
      style A2 fill:none,stroke:none
  
```

Description d'un cas

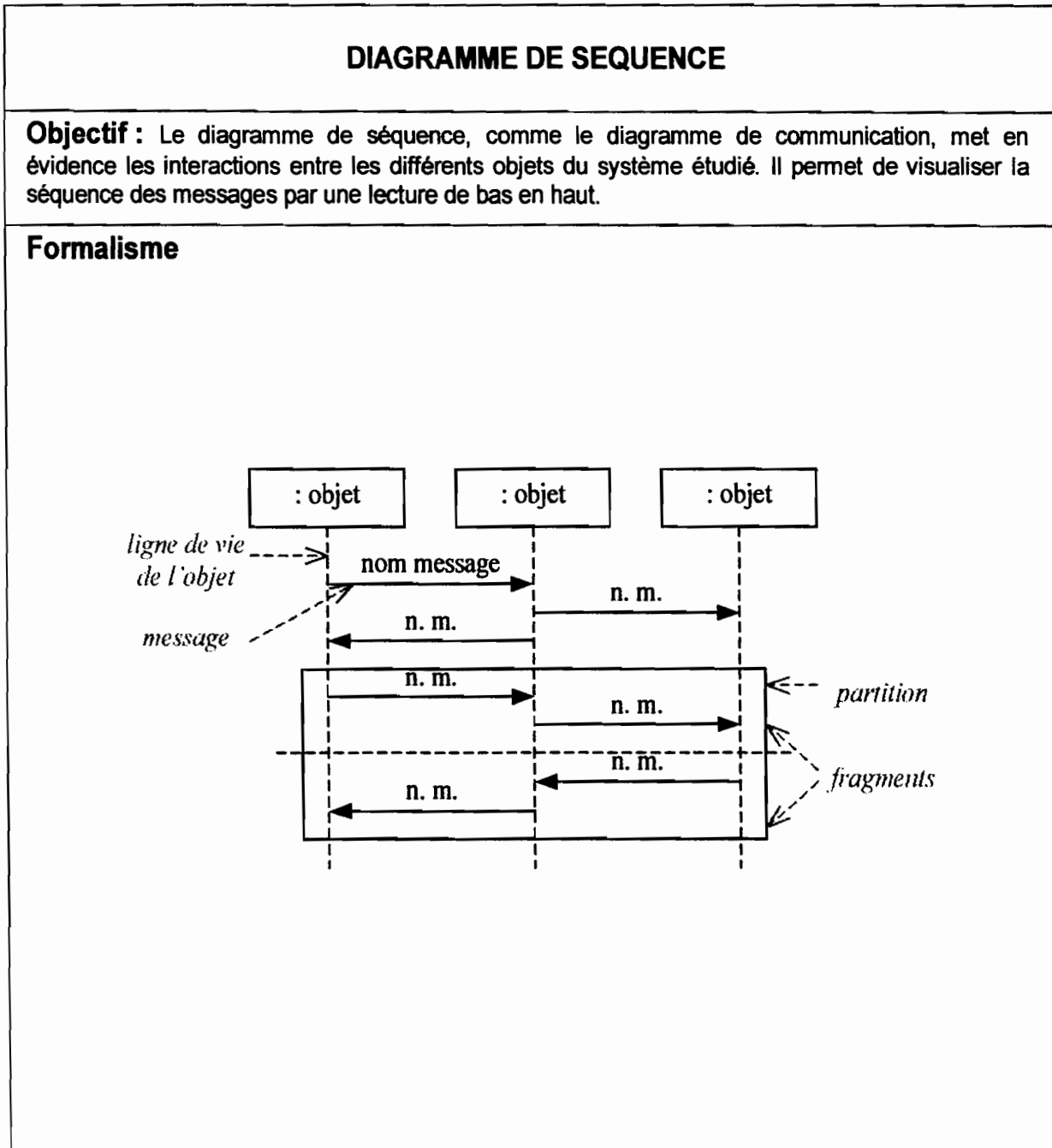
- Liste des actions
- Liste des règles
- Volumétries
- Fréquences
- Diagrammes

Principaux concepts	
Cas	Description du système étudié privilégiant le point de vue de l'utilisateur.
Acteur	Classe permettant de préciser l'acteur qui donne son point de vue.
Action	Traitement élémentaire. Un cas d'utilisation est composé d'une séquence d'actions.
« inclus »	Le premier cas inclus dans sa description les actions du deuxième cas.
« étend »	Le premier cas est une extension du deuxième cas.

Notation	
	Cas d'utilisation
	Acteur
	Association
	Contrainte
	Généralisation Spécialisation

ANNEXE (3)

Diagramme de séquence



Principaux concepts		Notation	
Objet	Emetteur ou récepteur d'un message.		Objet
Message	Information envoyée à un objet et provoquant en réponse le déclenchement d'actions associées à cet objet.		Message synchrone
			Message asynchrone
			Message retour
Ligne de vie	Ligne verticale, parallèle à l'axe du temps, indiquant le rôle de l'objet associé dans une interaction.		Partition
Partition	Unité organisationnelle. Elle est aussi appelée couloir.		Fragment
Fragment	Élément structurel d'une interaction.		

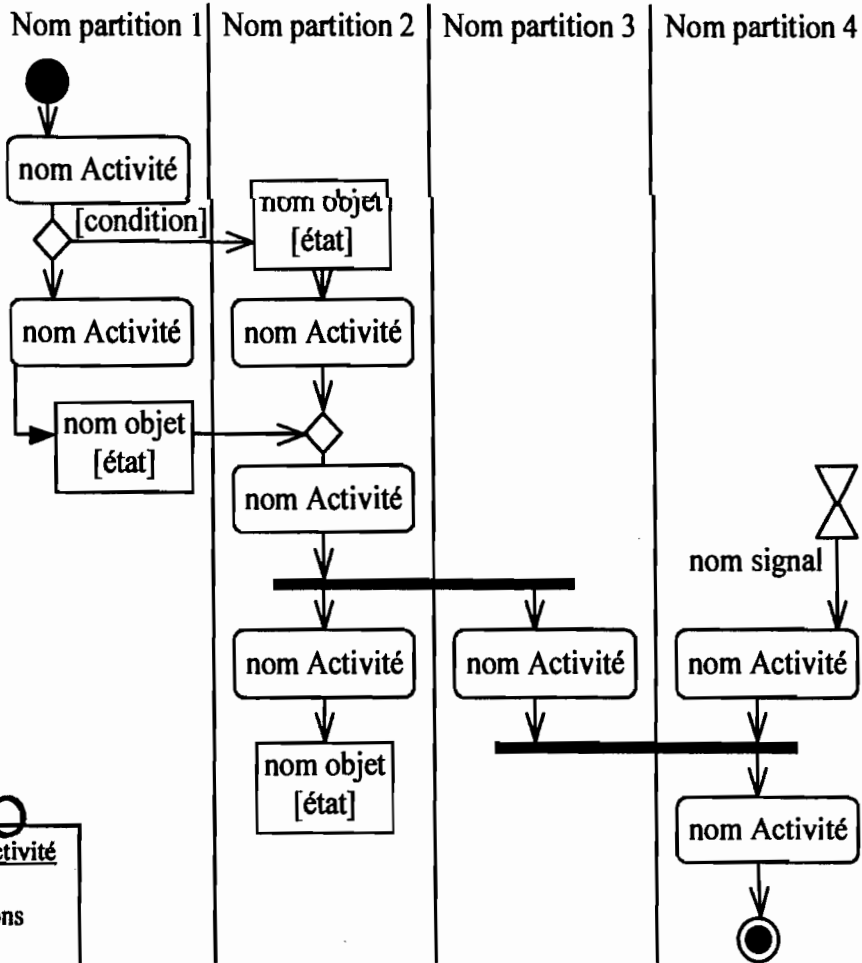
ANNEXE (4)

Diagramme d'activités

DIAGRAMME D'ACTIVITES

Objectif : Le diagramme d'activités permet de décrire un processus comme un ensemble organisé d'activités.

Formalisme



Description d'une activité

- Liste des actions

Principaux concepts	
Activité	Spécification comportementale qui représentent les étapes séquentielles et simultanées d'un processus.
Action	Traitement élémentaire. Une activité est un ensemble d'actions.
Partition	Couloir permettant de d'associer les activités à des acteurs.
Flot de contrôle	Représentation du déroulement d'un ensemble d'activités.
Flot d'objet	Représentation de la part prise par chaque objet dans l'exécution d'un travail.

Notation	
	Activité
	Flot
	Branchement ou fusion
	Synchronisation
	Partition
	Signal temporel
	Signal
	Nœud initial et final

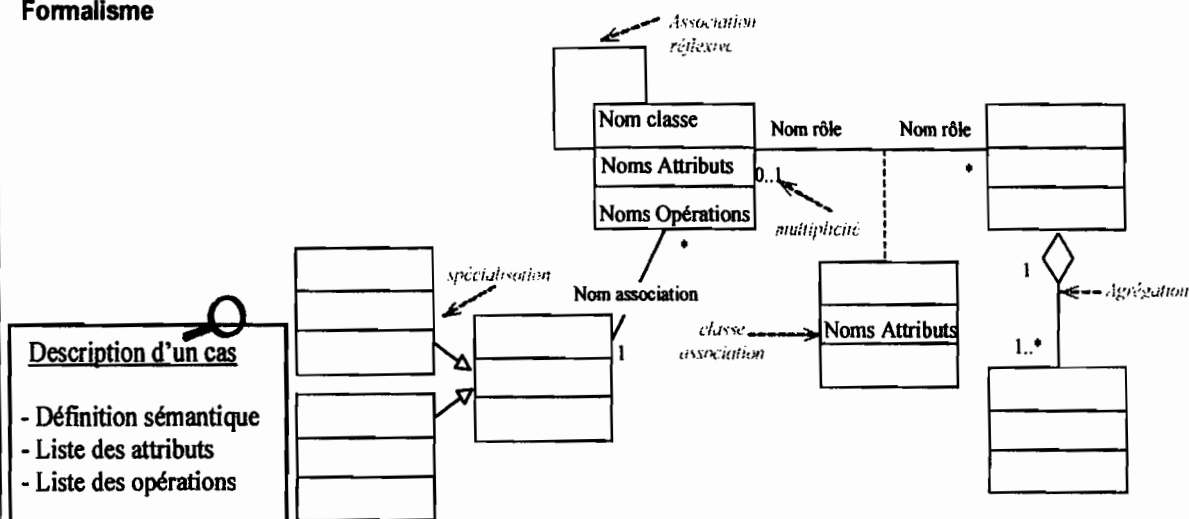
ANNEXE (5)

Diagramme de classes

DIAGRAMME DE CLASSES

Objectif : Permet de représenter l'ensemble des informations formalisées ayant fait l'objet d'une définition sur le fond et sur la forme, qui sont gérées dans le domaine.

Formalisme



Principaux concepts	
Classe	Ensemble d'objets sur lesquels on peut reconnaître des similitudes.
Attribut	Information élémentaire composant une classe. Un attribut peut permettre d'identifier la classe.
Opération	Fonctionnalité assurée par la classe.
Association	Lien entre classes.
Association réflexive	Association mettant en relation une classe avec elle-même.
Classe association	Association porteuse d'attributs.
Multiplicité	Nombre d'instances impliquées dans l'association.
Agrégation	Type d'association mettant en évidence une classe agrégée et une classe agrégat. Chaque objet de la classe agrégée est associé à un ou plusieurs objets de la classe agrégat.
Généralisation spécialisation	Permet d'identifier parmi les objets d'une classe (générique) des sous-ensembles d'objets (des classes spécialisées) ayant des définitions spécifiques.
Polymorphisme	Possibilité pour un même message de déclencher des traitements différents, suivant les objets auxquels il est adressé.

Notation	
	Classe
	Association
	Classe association
	Généralisation spécialisation
	Contrainte entre association
	Contrainte de spécialisation
	Décomposition
	Agrégation

ANNEXE (6)

MINISTRE DES ENSEIGNEMENTS SECONDAIRE,
SUPERIEUR ET DE LA RECHERCHE SCIENTIFIQUE

SECRETARIAT GENERAL



UNIVERSITE POLYTECHNIQUE DE BOBO-DIOULASSO

ECOLE SUPERIEURE D'INFORMATIQUE
01 BP 1091 BOBO-DIOULASSO 01
Tél. : 20 97 27 64 / Fax : 20 97 05 57

BURKINA FASO
Unité - Progrès - Justice

Année Académique 2009-2010

ATTESTATION PROVISOIRE

Je soussigné, Président du jury de délibération de troisième et dernière année du cycle des ingénieurs de travaux informatiques, option **Analyse et Programmation** atteste que :

Nom et prénom (s) : **TRAORE Lamine** Né (e) le **06/04/1986** à **Bobo-Dioulasso** N°matricule **2781** a pris part aux examens de la session de juin 2010 et a obtenu la moyenne générale de **14,26/20**

Décision du jury : **Admis** Mention : **Abien**

Fait à Bobo-Dioulasso, le 23 décembre 2010

Pour le Président du jury
P/D le Directeur Adjoint de l'ESI

Dr B. Michel Jonas SOME

ANNEXE (7)

MINISTRE DES ENSEIGNEMENTS SECONDAIRES,
SUPERIEUR ET DE LA RECHERCHE SCIENTIFIQUE

BURKINA FASO
Droits - Droppes - Postes

SECRETARIAT GENERAL

Année Académique 2009-2010

UNIVERSITE POLYTECHNIQUE DE BOBO-DIOULASSO

ECOLE SUPERIEURE D'INFORMATIQUE

DEPARTEMENT DE BOBO-DIOULASSO

TEL : 20 20 27 847 / FAX 20 20 06 670

ATTESTATION PROVISOIRE

Je soussigné, Président du jury de dissertation de troisième et deuxième année de cycle des ingénieurs de
travaux informatiques, option Analyse et Programmation atteste que :

Nom et prénom (s) : TRAORE Lamine Né (e) le 06/04/1986 à Bobo-Dioulasso N° matricule 2781 a pays (pays)

aux examens de la session de juin 2010 et a obtenu la moyenne générale de 04,26/20

Décision de jury : Admis Mastère Fabien

Fait à Bobo-Dioulasso, le 23 décembre 2010

LA PRESIDENTE
Pour le Président du jury
P/D le Directeur Adjoint de l'ESI
B. TRAORE
Dr B. TRAORE
BOBO-DIOULASSO