

REPUBLIQUE DU SENEGAL



UN PEUPLE – UN BUT – UNE FOI

**MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA
RECHERCHE**

DIRECTION DE L'ENSEIGNEMENT SUPERIEUR

INSTITUT SUPERIEUR DE TECHNOLOGIE INDUSTRIELLE

I.S.T.I



Sicap Sacré Cœur II – BP : 10155 – Tel. /Fax : 33 824 80 32 Dakar-Sénégal

Site web: www.ipg-isti.com

Email: information@ipg-isti.com

**MEMOIRE DE FIN DE CYCLE POUR L'OBTENTION DU DÎPLOME
DE TECHNICIEN SUPERIEUR**

OPTION : ANALYSTE PROGRAMMEUR

THEME :

**Etude et mise en place d'un système de
gestion des dossiers et Informations des
patients dans les hôpitaux du Sénégal.**

Présenté par :
Gabriel Mayid Dioh

Encadré par :
M. NGOR SECK

Année Académique : 2017-2018

Etude et mise en place d'un
système de gestion des
dossiers et Informations
des patients dans les
hôpitaux du Sénégal.

DEDICACES

Je dédie ce mémoire :

À mes parents Louise et Raymond Dioh qui se sont donnés corps et âmes pour que je réussisse mes études et qui font tout leur possible pour que je ne manque de rien.

À mes sœurs qui m'encouragent et m'apportent conseils et soutien.

À toutes personnes ayant contribué de près ou de loin à mon Développement ou ma réussite.

Je dédie spécialement ce mémoire à Mon défunt Grand Père, Frère Aloyse Dioh qui nous a quitté, que la terre lui soit légère.

REMERCIEMENTS

Nous tenons tout d'abord à remercier **Dieu le tout-puissant et miséricordieux**, qui nous a donné la force et la patience d'accomplir ce Modeste travail.

En second lieu, nous tenons à remercier notre encadreur **M. NGOR SECK** pour ses précieux conseils, son aide et sa disponibilité durant toute la période du travail.

Au **Major Aby Sané**, Responsable du Centre de Santé HLM5, à **Madame Fatou Touré Diallo** de la division Accueil et orientation de l'hôpital FANN et **M. Makhete Niang**, chef du service informatique à l'hôpital FANN, pour leurs aides et informations fournies.

À ma famille, surtout à mon père **Dr Raymond Dioh**, qui m'a permis de réaliser ce travail.

Nos remerciements s'étendent également à tous mes enseignants durant les années d'études.

Mes camarades de promotion à qui je renouvelle toute mon amitié et ma reconnaissance.

Nos vifs remerciements vont également aux membres du jury pour l'intérêt qu'ils ont porté à notre recherche en acceptant d'examiner notre travail et de l'enrichir par leurs propositions.

Enfin, nous tenons également à remercier toutes les personnes qui ont participé de près ou de loin à la réalisation de ce travail, **mais surtout au corps professoral de L'I.S.T.I** pour leur parfait professionnalisme car sans eux tout ceci ne serait réalisable.

TABLE DES MATIERES

Dédicaces.....	III
Remerciements.....	IV
Table des Matières.....	V
Liste des tableaux.....	X
Liste des figures.....	XI
Liste des sigles et Abréviations.....	XII
Avant-Propos.....	XV
Introduction.....	1
<u>CHAPITRE 1</u> : Cadre théorique et étude préalable.....	2
1.1-Présentation du sujet : Le dossier Patient.....	3
1.1.1-Définition.....	3
1.1.2-Contenu du dossier Patient.....	4
1.1.3-Procédure de Création du dossier patient.....	4
1.1.4-Qu'est-ce que la dématérialisation et quels sont les avantages de la dématérialisation des données de santé ?.....	4
1.2-La méthode d'analyse MERISE.....	5
1.2.1-Définition.....	5
1.2.2-Organisation de Merise.....	6
1.3-Problématiques et Solutions.....	8
1.3.1-Comment assurer la sécurité de l'accès de notre système ?	8

1.3.2-Comment partager des informations tout en essayant de respecter le secret médical (cas du DMP) ?.....	9
1.4-Le Modèle Conceptuel de la Communication (MCC).....	10
1.4.1-Formalisme du MCC.....	10
1.4.1.1-Définition.....	10
1.4.1.2-L'organisme.....	10
1.4.1.3-Acteur.....	11
1.4.1.3.1-Acteurs interne.....	11
1.4.1.3.2-Acteur externe.....	12
1.4.1.4-Flux d'information.....	12
1.4.2-Elaboration du MCC.....	13
<u>CHAPITRE 2 : Etude Conceptuelle</u>	14
2.1-Introduction.....	15
2.2-Le Modèle Conceptuel des Données (MCD).....	15
2.2.1-Formalisme du MCD.....	15
2.2.1.1-Définition.....	15
2.2.1.2-Entités et classe d'entité.....	15
2.2.1.3-Propriété.....	16
2.2.1.4-Relation.....	17
2.2.1.4.1-Définition de cardinalité.....	18
2.2.1.5-Contraintes d'Intégrité fonctionnelle (CIF).....	19
2.2.1.6-Le Dictionnaire des données.....	19

2.2.2-Elaboration du MCD.....	22
2.2.2.1-Le Dictionnaire de Donnée.....	23
2.3-Le Modèle Conceptuel des Traitements (MCT).....	24
2.3.1-Formalisme du MCT.....	24
2.3.1.1-Définition.....	24
2.3.1.2-Evénement.....	25
2.3.1.3-Opération.....	26
2.3.1.4-Synchronisation.....	26
2.3.2-Elaboration du MCT.....	27
CHAPITRE 3 : Etude détaillée et Implémentation.....	28
3.1-Etude détaillée.....	29
3.1.1-Definition.....	29
3.1.2-Le Modèle Organisationnel des Traitements (MOT).....	29
3.1.2.1-Formalisme du MOT.....	29
3.1.2.1.1-Définition.....	29
3.1.2.2-Elaboration du MOT.....	31
3.2-Implémentation.....	39
3.2.1-Introduction	39
3.2.2-Le Système de gestion de base de données (SGBD).....	39
3.2.2.1-Definition.....	39
3.2.2.2-But visé.....	40
3.2.2.3-Construction et fonctionnement.....	41

3.2.3-Les langages et choix	43
3.2.3.1-Le SQL.....	43
3.2.3.2-Le Html.....	44
3.2.3.3-Le CSS.....	45
3.2.3.4-Le PHP.....	46
3.2.3.5-Le JavaScript.....	47
3.2.4-Choix de l'outils de conception du système *.....	48
3.2.4.1-Certaines failles web et comment s'en protéger.....	49
3.2.4.1.1- La faille XSS.....	49
3.2.4.1.2- La faille Include.....	50
3.2.4.1.3- L'injection SQL.....	51
3.2.4.1.4- L'attaque par force brute	51
3.2.4.2-Comment assurer la confidentialité des informations du système ?	53
3.2.4.2.1-Le hashage.....	53
3.2.4.2.2-Le cryptage.....	55
3.2.4.3-Comment est-t-il possible de partager les données d'un site à l'autre et assurer la sécurité du partage de ses informations ?.....	55
3.2.5-Présentation de l'application.....	56
3.2.5.1-La page de Connection.....	56
3.2.5.2-La page d'accueil.....	57
3.2.5.3-Patient.....	57

3.2.5.3.1-La page d'ajout des patients.....	59
3.2.5.3.2-La page Actualisation du dossier.....	60
3.2.5.3.3-Consultation des informations patient.....	61
3.2.5.3.4- Modification des informations patient.....	62
3.2.5.3.5-Archivage du patient.....	63
3.2.5.4-Recherche hors hôpital.....	64
3.2.5.5-Trace.....	65
Conclusion.....	66
Références.....	67
Résumé en français +mots clés.....	69
Résumé en anglais +mots clés.....	70
Annexes.....	71

* : veut dire que des références techniques liées à la programmation, au réseau seront évoquées, vous pouvez ne pas comprendre si vous n'avez pas certaines notions.

LISTE DES TABLEAUX

<u>Tableau 1.1:</u> Organisation de Merise.....	7
<u>Tableau 2.1 :</u> Dictionnaire de donné d'un système de gestion des emprunts d'une bibliothèque.....	21
<u>Tableau 2.2 :</u> Dictionnaire de donnée de gestion du dossier patient.....	23
<u>Tableau 3.1 :</u> MOT Création du dossier patient.....	31

LISTE DES FIGURES

<u>Figure 1.1 :Formalisme Organisme d'un Mcc.....</u>	11
<u>Figure 1.2 : Formalisme Acteur interne du MCC.....</u>	12
<u>Figure 1.3 : Formalisme Acteur externe du MCC.....</u>	12
<u>Figure 1.4 : Diagramme acteur Flux.....</u>	13
<u>Figure 1.5: Diagramme acteur flux gestion des patients.....</u>	13
<u>Figure 2.1 : Formalisme d'une Entité MCD.....</u>	16
<u>Figure 2.2 : Exemple d'une entité client avec des propriétés.....</u>	17
<u>Figure 2.3 : Formalisme d'une Relation MCD.....</u>	17
<u>Figure 2.4 : Exemple d'une relation client/facture.....</u>	18
<u>Figure 2.5 : MCD de gestion du dossier patient.....</u>	22
<u>Figure 2.6 : Exemple MCT de facturation.....</u>	25
<u>Figure 2.7 : Formalisme d'un événement.....</u>	26
<u>Figure 2.8 : Formalisme d'une Opération.....</u>	26
<u>Figure 2.9 : Formalisme d'une Synchronisation.....</u>	26
<u>Figure 2.10 : MCT création du dossier patient.....</u>	27
<u>Figure 3.1 :Formalisme d'un MCT.....</u>	30
<u>Figure 3.2.: Exemple de Code SQL.....</u>	44
<u>Figure 3.3 : Exemple de Code HTML.....</u>	45
<u>Figure 3.4 : Exemple de Code CSS.....</u>	45
<u>Figure 3.5 : Exemple de Code PHP.....</u>	47
<u>Figure 3.6 : Exemple de Code JavaScript.....</u>	48
<u>Figure 3.7 : Captcha de Vérification.....</u>	52

LISTE DES SIGLES ET ABREVIATIONS

ANSI : American National Standards Institute

ASP : Active Server Pages

CIF : Contraintes d'intégrité fonctionnelle

CRLF: Carriage Return Line Feed

CRUD: Create, Read, Update, Delete

CSRF: Cross-Site Request Forgery

CSS: Cascading Style Sheets

DBMS: Database management system

DCL: Data Control Language

DDL: Data Definition Language

DML: Data Manipulation Language

ECMA: European Computer Manufacturers Association

FTP: File Transfer Protocol

HTML: Hypertext Markup Language

HTTP: Hypertext Transfer Protocol

HTTP: Hypertext Transfer Protocol Secure

IBM: International Business Machines

IP: Internet Protocol

JSP : JavaServer Pages

LCD : Langage de Contrôle de Données

LDD : langage de Définition de Données

LMD : Langage de Manipulation de Données

MCC : Modèle Conceptuel de Communication

MCD : Modèle Conceptuel des Données

MCT : Modèle Conceptuel des Traitements

MD5 : Message Digest 5

MERISE : MEthode pour Rassembler les Idées Sans Effort

MLC : Modèle Logique de Communication

MLD : Modèle Logique des Données

MLT : Modèle Logique des Traitements

MOC : Modèle Organisationnel de Communication

MOD : Modèle Organisationnel des Données

MOT : Modèle Organisationnel des Traitements

MPC : Modèle Physique de Communication

MPD : Modèle Physique des Données

MPT : Modèle Physique des Traitements

PHP/FI : Personal Home Page Tools/Form Interpreter

PHP: HyperText Preprocessor

QBE: Query By Example

RSE : Responsabilité Sociale d'Entreprise

SGBD : Le Système de gestion de base de données

SHA : Secure Hash Algorithm

SIH : système d'information hospitalier

SQL: Structured Query Language

SSL: Secure Sockets Layer

TSL: Transport Layer Security

URL: Uniform Resource Locator

VPN: Virtual Private Network

WEB: Website

WWW: World Wide Web

XML: Extensible Markup Language

XSS : Cross-Site Scripting

AVANT-PROPOS

Créer en 1994, l’Institut Supérieur de Technologie Industrielle (ISTI) vise une formation technique équilibrée en synergie avec le milieu industriel.

L’institut Supérieur de Technologie Industrielle offre des formations dans différentes filières :

- Informatique de Gestion
- Électromécanique
- Informatique et Electronique Industrielle
- Électricité Industrielle
- Génie Civil

Le présent mémoire rentre dans le cadre de l’obtention du Diplôme de Technicien Supérieur (DTS), option Informatique de Gestion.

L’idée de cette étude m’est venu en voulant mettre mes connaissances acquises au service de cet art qu’est la médecine, qui est pratiquée par mon Père. C’est en ce sens que je me suis posé ces questions : comment puis-je aider ? Que puis-je mettre en œuvre pour faciliter la vie des médecins ? Comment être innovant, tout en étant utile ?

C’est en ce sens que mes recherches ont été portées après une longue réflexion sur un système de gestion des patients mais aussi de partage d’informations entre différentes structures. Mais où devrais-je implanter cette structure pour qu’elle apporte une aide maximale ? Évidement l’hôpital avec les dizaines de patients que celui-ci accueille chaque jour c’est ce dernier qui en a le plus besoin, même si toutes les autres structures comme les cabinets, les postes de santé... pourraient aussi en avoir besoin, et d’ailleurs ceci permettrait de pouvoir avoir des informations beaucoup plus précises sur un patient, mais bon peut être plus tard, commençons d’abord par voir plus petit pour ensuite faire encore plus grand.

Cette étude se veut être une contribution devant permettre de mettre en relief les différents obstacles pouvant être rencontrés lors de l’élaboration du système et ainsi trouver des solutions pour lever ces obstacles

INTRODUCTION

L'informatique prend une place de plus en plus importante au sein du monde médical. On considère généralement que l'application de l'informatique au domaine de la santé est restreinte ou cloisonnée à un ensemble de techniques et d'outils mais n'oublions pas que l'apport de l'informatique permet aux médecins de simplifier de nombreuses tâches (comptabilité, gestion des données...), d'accéder facilement au dossier médical de leurs patients, de sécuriser certaines informations médicales

Le Système d'Information Hospitalier (SIH) quant à elle, est un système d'information qui est capable, selon des règles et modes opératoires prédéfinis, d'acquérir des données, de les évaluer, de les traiter par des outils informatiques ou organisationnels, de distribuer des informations contenant une forte valeur ajoutée à tous les partenaires internes ou externes de l'établissement. Le SIH doit se reposer sur une gestion commune du Dossier Patient Informatisé unique, au cœur du système, autour duquel gravitent toutes les applications, qui, outre la prise en charge médicale du patient, contribuent à alimenter les outils d'aide à la décision. Ce dossier correspondra à la somme des informations d'un patient donné, produites par les acteurs de soins. Lors de l'étude et du recueil des informations des questions ont été posées :

Qu'est-ce que la dématérialisation et quels sont les avantages de la dématérialisation des données de santé ? Comment assurer la sécurité de l'accès au système (dans notre cas un site Web) ? Comment assurer la confidentialité des informations du système ? Comment est-t-il possible de partager les données d'un site à l'autre et assurer la sécurité du partage de ses informations ? Comment partager des informations tout en essayant de respecter le secret médical (cas du DMP) ?

Dans ce sens le recueil nous a permis de mieux comprendre le SIH face aux problèmes qu'il rencontre tels que la perte de données, une gestion archaïque des dossiers patients sur du support papier, des problèmes d'espace ou de stockage des dossiers, une désorganisation des dossiers patients, une mauvaise prise en charge d'un patient, la mise en place de notre système permettra d'apporter la solution aux problèmes.

Dans l'élaboration du mémoire, nous commencerons d'abord par le Cadre théorique et l'étude préalable puis par l'étude conceptuelle pour terminer par l'étude détaillée et l'implémentation.

CHAPITRE 1 :

Cadre théorique et étude
préalable.

1.1-Présentation du sujet : Le dossier Patient

1.1.1-Définition

Le dossier du patient est le lieu de recueil et de conservation des informations administratives, médicales et paramédicales, formalisées et actualisées, enregistrées pour tout patient accueilli, à quelque titre que ce soit.

Le dossier du patient assure la traçabilité de toutes les actions effectuées. Il est un outil de communication, de coordination et d'information entre les acteurs de soins et avec les patients. Il permet de suivre et de comprendre le parcours hospitalier du patient. Il est un élément primordial de la qualité des soins en permettant leur continuité dans le cadre d'une prise en charge pluriprofessionnelle et pluridisciplinaire. Le rôle et la responsabilité de chacun des différents acteurs pour sa tenue doivent être définis et connus.

L'informatisation du Dossier Patient doit ainsi permettre à toute direction de répondre à ses objectifs stratégiques :

*Finaliser la démarche d'informatisation engagée

*Adapter l'outil informatique à l'organisation de la structure en fonction des besoins exprimés et des spécificités des modes de prises en charge.

*Faciliter la gestion du dossier patient afin que les professionnels de santé puissent se consacrer pleinement à leur métier

*Favoriser à tous les niveaux du management une gestion optimisée de l'activité et des ressources (notion de performance du dossier informatisé du patient)

*Répondre aux nombreuses contraintes règlementaires actuelles et à venir auxquelles doivent répondre les établissements sanitaires

*Optimiser l'urbanisation de son Système d'Information Hospitalier.

Dans la dématérialisation des données de santé la confidentialité et la sécurité sont très importantes pour le respect de la vie privée des patients.

1.1.2-Contenu du dossier patient

Le dossier patient complet peut contenir plusieurs éléments (Voir annexe page 71).

Mais pour éviter une lourdeur du dossier, plusieurs contenus ont été retirés pour ne contenir que l'essentiel c'est en ce sens que l'on retrouve un dossier patient avec les informations nécessaires du patient, ou un dossier patient résumé (Voir annexe page 72).

1.1.3-Procédure de Cr éation du dossier patient

L'étude que nous avons réalisée et les recherches nous a montré que tous les hôpitaux du Sénégal non pas le même niveau de développement certes et sont différents, mais ils disposent de point commun dans l'élaboration de la création du dossier patient. Comment cela se passe-t-il ?

Le patient se présente à l'hôpital pour une consultation, lorsque ce dernier se présente à l'accueil il manifeste une demande de consultation s'il dispose d'un ticket de consultation, il va être dirigé vers un médecin sinon dans le contraire il va être dirigé à la caisse pour l'achat du ticket c'est après cela que l'accueil pourra le diriger vers un médecin pour consultation ; Après consultation le médecin pourra oui ou non, établir si ce dernier a besoin d'être hospitalisé, si le médecin établi une hospitalisation le patient rempli une fiche d'hospitalisation et son dossier patient est créé. Ce dossier contiendra une partie administrative du patient c'est à dire son nom, prénom, sa date de naissance, date d'entrée...Ce dossier contiendra aussi le motif d'hospitalisation avec la personne l'ayant hospitalisé, les différents examen cliniques et paracliniques faites, l'histoire de la maladie, les hypothèses de diagnostics et Résumé syndromique, les résultats d'analyse et diagnostiques retenus, puis les traitements effectués, et une description de l'évolution de la maladie.

1.1.4-Qu'est-ce que la dématérialisation et quels sont les avantages de la dématérialisation des données de santé ?

La dématérialisation est une chaîne de valeurs technologique, économique, et surtout organisationnelle, dont chaque maillon apporte une valeur à optimiser au travers d'une démarche projet, elle consiste à transformer des documents physiques en fichiers

numériques ou à créer ces documents directement sous forme numérique pour les intégrer à un processus. En effet la dématérialisation contribue de fait à une augmentation exponentielle de la production et de la circulation de l'information.

Les avantages en sont multiples du point de vue :

Sécurité : les données électroniques sont faciles à protéger et à dupliquer et moins exposées aux dégradations physiques et aux catastrophes naturelles. La disparition des manipulations physiques évite les pertes de documents.

Confidentialité : La gestion des accès est facilement paramétrable et permet de s'assurer que seul le personnel médical habilité consulte l'information. Le transit physique du dossier disparaît et avec lui les risques de coups d'œil intempestifs.

Pérennité : si la numérisation est pratiquée par un professionnel, les formats des données sont standards et publics et garantissent une lisibilité de l'information pour une très longue période.

Ubiquité : À l'inverse d'un document papier, un dossier numérisé est accessible au même moment par de nombreuses personnes, quelle que soit leur localisation.

Disponibilité : l'information est disponible immédiatement et en tout lieu.

Réduction des coûts : l'information numérique est accessible à très bas coût sans avoir à gérer les mouvements physiques de sortie d'archives.

1.2-La méthode d'analyse MERISE

1.2.1-Introduction

On appelle analyse informatique la discipline qui explique comment réaliser un cahier des charges qui décrive exactement et précisément ce que l'on attend.

Ce dernier permettra une meilleure réalisation du système et l'analyse des besoins pour sa réalisation.

Plusieurs méthodes d'analyse son utiliser pour l'étude d'un système, mais notre choix se portera sur Merise de par notre connaissance de ce dernier, de par son adaptation à notre système, mais aussi du fait de sa popularité d'où une meilleure documentation

mais aussi une facilité d'aide en cas de problème. Merise permettra une meilleure conception et réalisation de notre système. MERISE est une méthode de conception, de développement et de réalisation de projets informatiques. Le but de cette méthode est d'arriver à concevoir un système d'information. La méthode MERISE est basée sur la séparation des données et des traitements à effectuer en plusieurs modèles conceptuels et physiques.

La séparation des données et des traitements assure une longévité au modèle. En effet, l'agencement des données n'a pas à être souvent remanié, tandis que les traitements le sont plus fréquemment.

La méthode MERISE (MEthode pour Rassembler les Idées Sans Effort) créé entre 1978-1979 en France par Peter Chen et Hubert Tardieu à Aix-en-Provence (France), et fait suite à une consultation nationale lancée en 1977 par le ministère de l'Industrie français dans le but de choisir des sociétés de conseil en informatique afin de définir une méthode de conception de systèmes d'information.

1.2.2-Organisation de Merise

Merise est organisé en :

- 3 Domaines :

*Domaine de Communication

*Domaine des Données

*Domaine des traitements

- 4 niveaux :

*Niveau Conceptuel

*Niveau Organisationnel

*Niveau Logique

*Niveau Physique

Le tableau ci-dessous nous montre cette organisation de façon plus détaillé :

Tableau 1.1: Organisation de Merise

Domaine de Communication	Domaine des Données	Domaine des traitements
1. Modèle Conceptuel de Communication (MCC) Intervenant, Domaine, Message Définition du Pourquoi	3. Modèle Conceptuel des Données (MCD) Individu, Association, Propriété Définition des limites, connaissance du domaine et/ou du vocabulaire du projet Invariant par rapport à la géolocalisation et par rapport au logiciel qui sera utilisé	2. Modèle Conceptuel des Traitements (MCT) Opération, Règle de gestion, Evénements, Résultats, Synchronisation des opérations Définition du Quoi, Que faire Invariant par rapport à la géolocalisation et par rapport à l'organisation, la répartition des traitements entre l'homme et les ordinateurs.
6. Modèle Organisationnel de Communication (MOC)	5. Modèle Organisationnel des Données (MOD) Extrait du MCD pour chaque poste de travail	4. Modèle Organisationnel des Traitements (MOT) Phase, Unité Géographique de Traitement (UGF), Procédure, Poste de travail, Tâche Définition du Qui fait Quoi sur Quel Poste de travail, ...
9. Modèle Logique de Communication (MLC)	7. Modèle Logique des Données (MLD) Entité, Relation, Champ, Clé primaire, Clé étrangère, Clé candidate, Index	8. Modèle Logique des Traitements (MLT) Transaction, Grille d'écran, Maquette d'état, Module de traitement,
	Dérivation des MOD en vue de respecter les règles logiques d'une base de données Dépendant de la géolocalisation Invariant par rapport au logiciel qui sera utilisé	Avec quelle allure générale Dépendant de la géolocalisation Invariant par rapport aux machines utilisées
11. Modèle Physique de Communication (MPC)	9. Modèle Physique des Données (MPD) Fichier, table, Index Dérivation du MLD, Sécurité Implantation, Dimensionnement, Dépendant du logiciel utilisé	10. Modèle Physique des Traitements (MPT) Transaction, Grille, Maquette précisément Comment et avec quels outils ou logiciels Dépendant des matériels et des logiciels utilisés

1.3-Problématiques et Solutions

1.3.1-Comment assurer la sécurité de l'accès de notre système ?

La protection des données personnelles occupe une place centrale dans la Responsabilité Sociale d'Entreprise (RSE). La majorité des applications Web souffrent de vulnérabilités, un site peut être plus ou moins sûr, et la sécurité est souvent une affaire de compromis. Elle dépend de nombreux facteurs, tels que les connaissances du développeur bien entendu, mais aussi la conception et la complexité du site, le coût et les délais de fabrication et même parfois aussi étonnant que cela puisse paraître de la responsabilisation de ses utilisateurs (par exemple refus de mise à jour du navigateur Web utilisé, utilisation d'un mot de passe faible ou facile à trouver).

Ces dernières années, les menaces ont changé et ont été automatisées.

L'attaque classique consistant à modifier l'aspect d'un site existe toujours, mais la principale attaque est insidieuse, elle cherche à bloquer le fonctionnement du site, ou à l'utiliser pour s'y infiltrer et commettre des actions illégitimes de manière la plus discrète possible.

Le deuxième changement est lié aux outils utilisés par les pirates : grâce à des scripts et robots, ils vont pouvoir scanner sur le web un très grand nombre de sites pour y détecter une faille et s'y infiltrer. Quelle que soit votre activité, vous pouvez être visé sans que ce soit votre activité en tant que telle qui intéresse le pirate.

C'est l'utilisation de ces outils très puissants qui augmente la menace et la fréquence des attaques.

C'est en ce sens que nous devons, nous développeur essayer au maximum de protéger notre système, et ce dernier dépend de plusieurs facteurs comme le choix de l'outil de conception du système web.

Lors de la conception du système plus précisément à la partie implémentation de la page 49 à 56 nous parlons de la sécurité de façon plus détaillée et nous montrons comment assurer cette sécurité de manière plus technique.

1.3.2-Comment partager des informations tout en essayant de respecter le secret médical (cas du DMP) ?

Le DMP est un carnet de santé numérique.

Le dossier médical partagé doit permettre aux professionnels de santé d'avoir accès aux informations de santé d'un patient et aux soins qui lui ont été prescrits.

En France par exemple un décret officiel fixe la liste des informations susceptibles d'y figurer mais seulement à titre indicatif.

Ainsi, votre DMP pourra contenir :

=>des prescriptions

=> des comptes-rendus opératoires et radiologiques

=> des résultats d'analyses

=> des comptes-rendus d'hospitalisation

=> le consentement à l'anesthésie

=> des allergies éventuelles

=> les coordonnées des proches à prévenir en cas d'urgence

Ce dispositif compromet-il le secret de médical, la réponse est non car même avec la généralisation de ce dossier médical partagé, le secret médical reste un principe de base de notre système de soins, l'accès aux données qu'il contient sera restreint selon l'utilité pour le professionnel de santé, dans notre cas entre les différents médecins d'un service hôpital se trouvant dans notre système.

Le DMP est un outil collaboratif entre professionnels de santé. Il collecte les informations utiles de santé d'un patient pour favoriser la continuité et la coordination des soins, mais aussi en cas d'urgence, mais aussi il sert de support d'échange entre les professionnels de santé.

Chaque DMP constitue, en soi, une microbase de données contenant, lorsqu'il est alimenté, divers éléments d'information concernant la santé du patient.

Lors de la mise en place de notre site, pour mieux respecter la vie privée du patient une clause sera signifiée pour demander au patient un accord pour partager certaines de ces informations comme ceux citer précédemment entre les différents hôpitaux

futurs où ils seront pris en charge, dans cette clause nous montrerons les avantages de ce système au patient pour sa meilleure prise en charge.

Dans notre site il y'aura une option ‘**Partager**’ dans la création du dossier patient qui permettra si elle est cochée de pouvoir partager les informations (DMP) de ce dernier entre les hôpitaux du système. Cette option pourra être désactivée à tout moment.

Une option ‘**Personne sensible**’ est aussi ajoutée lors de la création du dossier et après la création cette option si elle est cochée, même si le partage est autorisé les informations que l'on pourra voir seront le nom du médecin qui le traite, l'hôpital et le numéro. Cette option pourra être désactivée avec une autorisation spéciale.

Un médecin d'un service ne pourra qu'ajouter des patients de son service. Il pourra aussi qu'accéder au patient de son service.

1.4-Le Modèle conceptuel de la communication (MCC)

1.4.1-Formalisme du MCC

1.4.1.1-Définition

Le modèle conceptuel des communications appelé aussi modèle des acteurs ou modèle de flux représente au niveau conceptuel, les échanges ou le flux d'informations entre les acteurs. Le modèle conceptuel des communications est utilisé dans l'étude de l'existant (la modélisation des habitudes de travail dans l'organisation concernée). Les étapes pour construire un modèle conceptuel de communication sont :

*Délimiter le domaine étudié.

*Réduire la complexité par subdivision en sous problème.

*Identification des acteurs internes et externes.

*Modélisation des échanges de l'information entre les différents acteurs.

1.4.1.2-L'organisme

Un organisme est un nom générique désignant l'entreprise, la structure, le service public, le service, etc. qui fait l'objet d'une étude, c'est le domaine étudié.

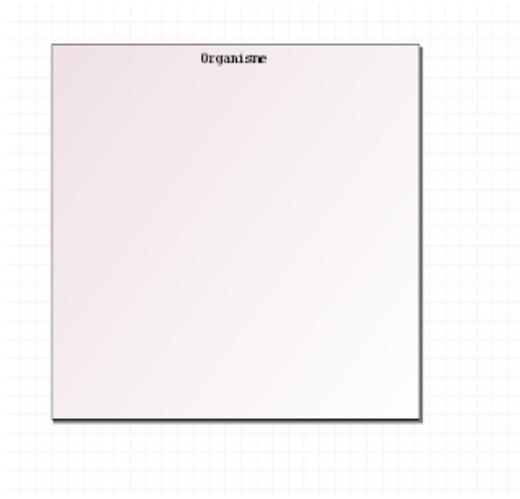


Figure 1.1 :Formalisme Organisme d'un Mcc

1.4.1.3-Acteur

L'acteur présente une unité active (fait quelque chose) intervenant dans le fonctionnement d'un système, c'est un rôle plutôt qu'une personne physique. Il peut :

- Stimuler (activer -exciter) par des flux d'information.
- Transformer et émettre des flux d'information.

Nous avons deux types d'acteurs :

-L'acteur interne

-L'acteur externe

1.4.1.3.1-Acteurs interne

Un acteur interne est un acteur faisant partie du système d'information étudié, si le domaine est complexe on peut considérer un acteur interne comme un sous-domaine et détailler ce sous-domaine dans un nouveau MCC.

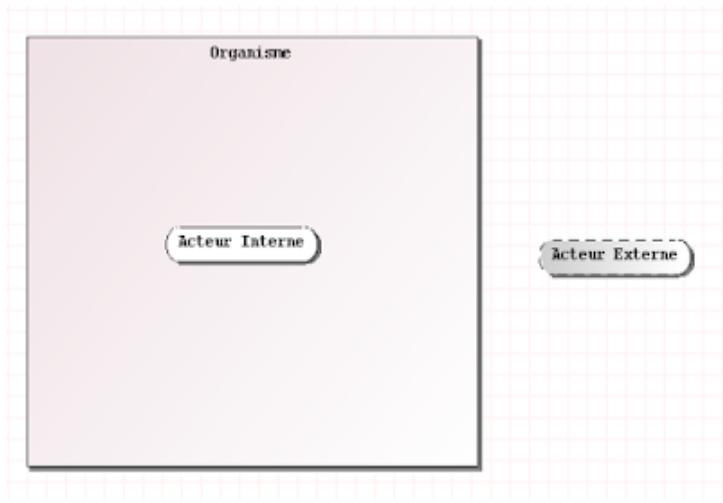


Figure 1.2 : Formalisme Acteur interne du MCC

1.4.1.3.2-Acteur externe

C'est un élément externe avec lequel le système échange des flux d'information.

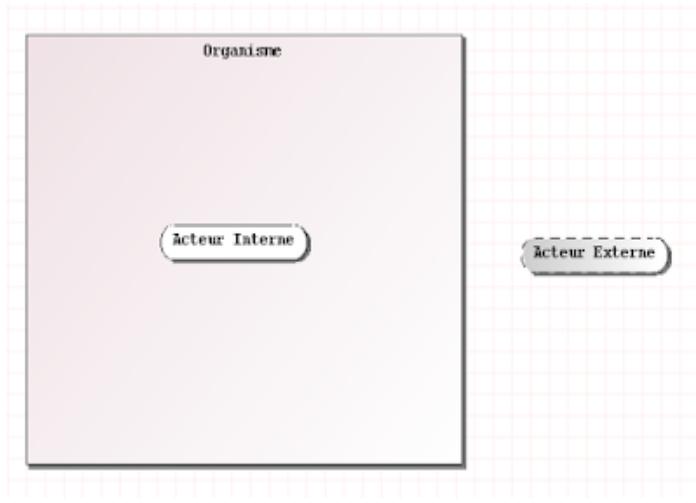


Figure 1.3 : Formalisme Acteur externe du MCC

1.4.1.4-Flux d'information

C'est un échange d'information entre deux acteurs (documents, appels téléphoniques, données informatiques).

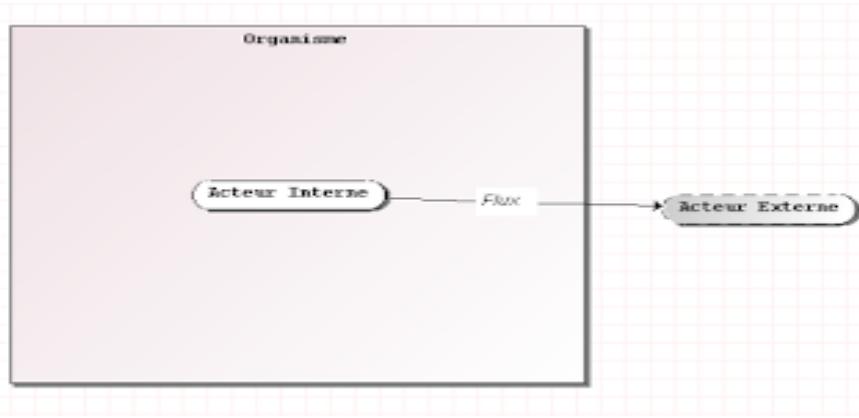


Figure 1.4 : Diagramme acteur Flux

1.4.2-Elaboration du MCC

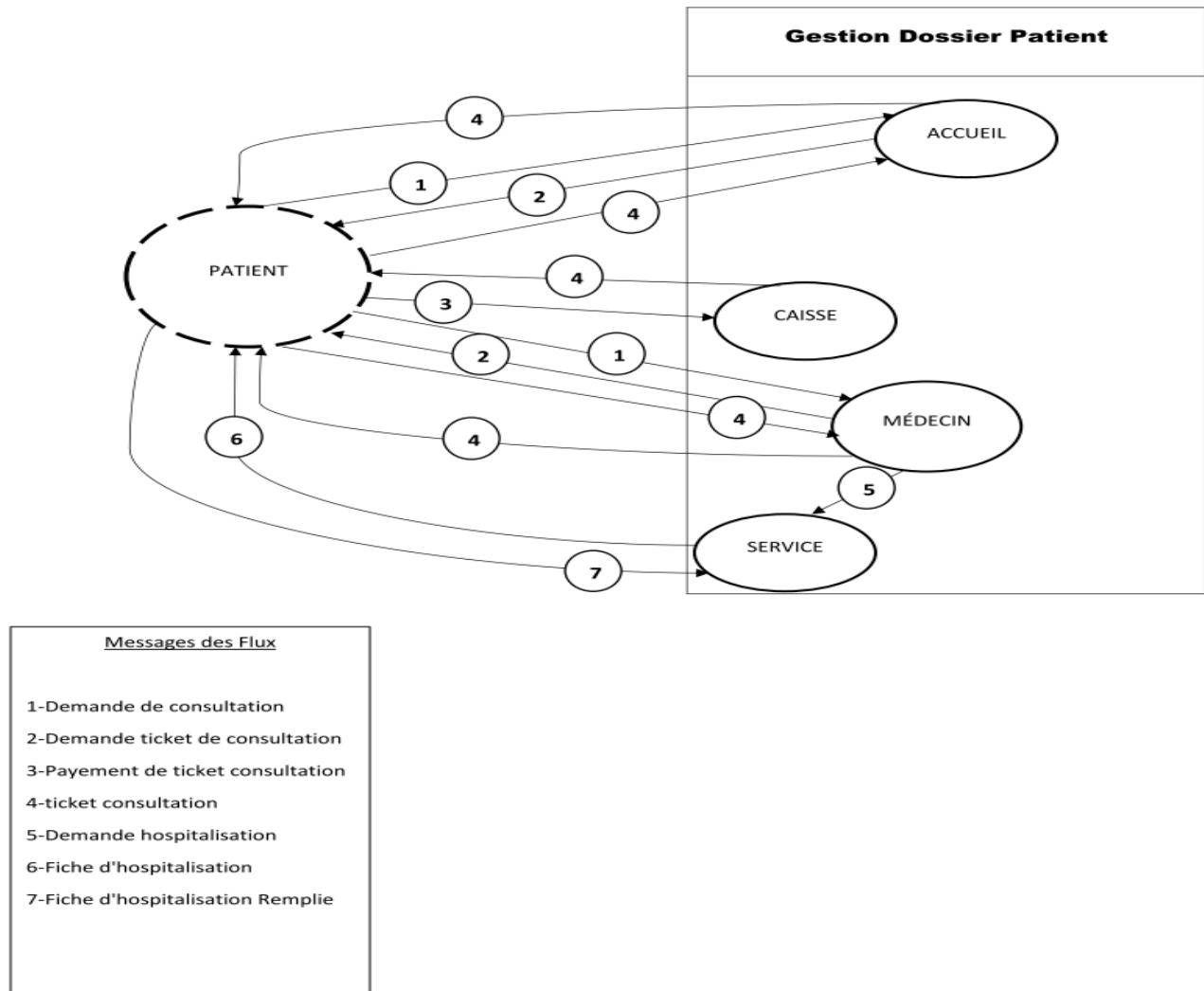


Figure 1.5: Diagramme acteur flux gestion des patients

CHAPITRE 2 :

Etude Conceptuelle.

2.1-Introduction

L'étude conceptuelle consiste à concevoir un système en faisant abstraction de toutes les contraintes techniques ou organisationnelles et cela tant au niveau des données que des traitements. Cette étude conceptuelle a pour but de répondre à la question Quoi ? (Le quoi faire, avec quelles données).

Ainsi, cette étude sera sanctionnée par deux modèles à savoir :

- Le **Modèle Conceptuel des Données (MCD)**.
- Le **Modèle Conceptuel des Traitements (MCT)**.

2.2-Le Modèle Conceptuel des Données (MCD)

2.2.1-Formalisme du MCD

2.2.1.1-Définition

Le modèle conceptuel des données (**MCD**) a pour but de d'écrire de façon formelle les données qui seront utilisées par le système d'information. Il s'agit donc d'une représentation des données, facilement compréhensible, permettant de décrire le système d'information à l'aide d'entités.

Le formalisme utilisé est connu sous le nom de "Schéma Entité-Relation". Il se base autour de 3 concepts principaux, les entités, les relations et les propriétés.

2.2.1.2-Entités et classe d'entité

Une entité est la représentation d'un élément matériel ou immatériel ayant un rôle dans le système que l'on désire décrire.

On appelle **classe d'entité** un ensemble composé d'entités de même type, c'est-à-dire dont la définition est la même. Le classement des entités au sein d'une classe s'appelle *classification* (ou *abstraction*). Une entité est une *instanciation* de la classe. Chaque entité est composée de propriétés, données élémentaires permettant de la décrire.

Prenons par exemple une *Ford Fiesta*, une *Renault Laguna* et une *Peugeot 306*. Il s'agit de 3 entités faisant partie d'une classe d'entité que l'on pourrait appeler *voiture*. La *Ford Fiesta* est donc une instantiation de la classe *voiture*. Chaque entité peut posséder les propriétés *couleur*, *année* et *modèle*.

Les classes d'entités sont représentées par un rectangle. Ce rectangle est séparé en deux champs :

- Le champ du haut contient le libellé. Ce libellé est généralement une abréviation pour une raison de simplification de l'écriture. Il s'agit par contre de vérifier qu'à chaque classe d'entité correspond un et un seul libellé, et réciproquement
- Le champ du bas contient la liste des propriétés de la classe d'entité



Figure 2.1 : Formalisme d'une Entité MCD

2.2.1.3-Propriété

Une propriété est une donnée élémentaire d'une entité.

Exemple : Nom du client : DIOH, NDIAYE

Ville : DAKAR, THIES

Une propriété n'est rattachée qu'à une et une seule entité.

Pour faire référence à chaque occurrence d'une entité, l'entité doit être munie d'un identifiant. L'identifiant est composé d'une ou de plusieurs propriétés de l'entité. A chaque valeur de l'identifiant doit correspondre une et une seule occurrence de l'entité.

Exemple : Il est nécessaire d'ajouter un code client dans l'entité « CLIENT » car les propriétés existantes ne sont pas nécessairement uniques (ex Nom du client). Le code client permet donc l'unicité du client.

Là où les propriétés qui constituent l'identifiant d'une entité sont soulignées.

CLIENT
<u>Code client</u>
Nom
Prénom
Adresse
Code postal
Ville

Figure 2.2 : Exemple d'une entité client avec des propriétés

2.2.1.4-Relation

Une relation décrit un lien entre deux ou plusieurs entités. Chaque relation possède un nom, qui est généralement constitué par un verbe à l'infinitif.

Chaque relation a obligatoirement un identifiant, qui est composé par les identifiants des entités auxquelles elle est liée. L'occurrence d'une relation est représentée par les occurrences des entités liées à la relation.

Exemple

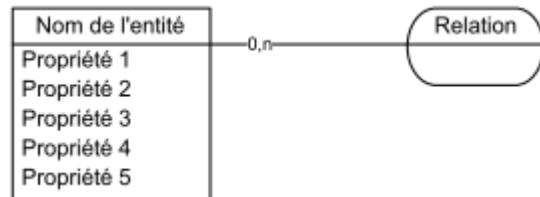


Figure 2.3 : Formalisme d'une Relation MCD

Une relation est liée à chacune de ses entités par un lien sur lequel on indique les Cardinalités. Les cardinalités représentent la participation de l'entité concernée à la relation.

Le premier nombre indique la cardinalité minimale, le deuxième la cardinalité maximale.

Exemple de relation entre un client et une commande

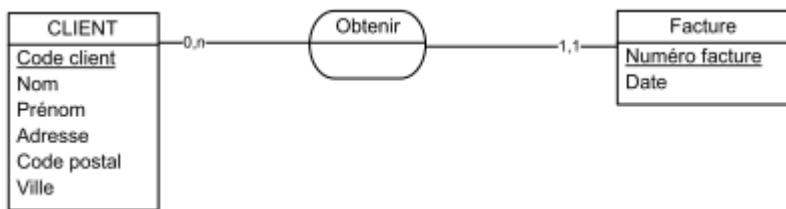


Figure 2.4 : Exemple d'une relation client/facture

Entité « CLIENT » et relation « OBTENIR »

- * Cardinalité minimale = 0 : chaque client obtient aucune facture.
- * Cardinalité maximale = n : chaque client peut obtenir plusieurs (n) factures.

Entité «FACTURE» et relation «OBTENIR»

- * Cardinalité minimale = 1 : chaque facture est obtenue par au moins un client.
- * Cardinalité maximale =1 : chaque facture est obtenue au maximum par un seul Client.

2.2.1.4.1-Définition de cardinalité

La cardinalité minimale exprime le nombre de fois minimum où une occurrence d'une entité participe à une relation. Cette cardinalité est généralement 0 ou 1.

- * Cardinalité minimale = 0 : Certaines occurrences de l'entité ne participent pas à la relation
- * Cardinalité minimale = 1 : Chaque occurrence de l'entité participe au moins une fois à la relation

La cardinalité maximale exprime le nombre de fois maximum où une occurrence d'une entité participe à une relation. Cette cardinalité vaut souvent 1 ou n, avec n indiquant une valeur >1 mais pas connue.

- *Cardinalité maximale = 1 : Chaque occurrence de l'entité participe au plus une seule fois à la relation.
- * Cardinalité maximale = n : Chaque occurrence de l'entité peut participer plusieurs fois à la relation.

2.2.1.5-Contraintes d'intégrité fonctionnelle (CIF)

Une CIF indique que l'une des entités est totalement déterminée par la connaissance de l'autre.

Exemple : connaissant une facture bien précise, on connaît avec certitude le client Correspondant.

2.2.1.6-Le Dictionnaire des données

Le dictionnaire des données est un document qui regroupe toutes les données que vous aurez à conserver dans votre base (et qui figureront donc dans le MCD). Pour chaque donnée, il indique :

- Le **code mnémonique** : il s'agit d'un libellé désignant une donnée (par exemple « *titre_l* » pour le titre d'un livre)
- La **désignation** : il s'agit d'une mention décrivant ce à quoi la donnée correspond (par exemple « *titre du livre* »)
- Le **type de donnée** :
 - **A ou Alphabétique** : lorsque la donnée est uniquement composée de caractères alphabétiques (de 'A' à 'Z' et de 'a' à 'z')
 - **N ou Numérique** : lorsque la donnée est composée uniquement de nombres (entiers ou réels)
 - **AN ou Alphanumérique** : lorsque la donnée peut être composée à la fois de caractères alphabétiques et numériques
 - **Date** : lorsque la donnée est une date (au format AAAA-MM-JJ)
 - **Booléen** : Vrai ou Faux
- La **taille** : elle s'exprime en nombre de caractères ou de chiffres. Dans le cas d'une date au format AAAA-JJ-MM, on compte également le nombre de caractères, soit 10 caractères. Pour ce qui est du type booléen, nul besoin de préciser la taille (ceci dépend de l'implémentation du SGBDR).
- Et parfois des **remarques** ou **observations** complémentaires (par exemple si une donnée est strictement supérieure à 0, etc.).

Remarques :

- Les données qui figurent dans le MCD (et donc dans le dictionnaire des données) doivent être, dans la plupart des cas, **élémentaires** :
- Elles ne doivent pas être **calculées** : les données calculées doivent être obtenues, par le calcul, à partir de données élémentaires qui, elles, sont conservées en base. Cependant, il existe quelques cas où il s'avère pertinent de conserver, pour des raisons d'optimisation, une donnée calculée, le montant d'une commande par exemple. On ne conservera cependant pas les données calculées intermédiaires sauf en cas d'obligation légale (c'est le cas pour un montant HT par exemple, où les composantes peuvent d'ailleurs avoir un prix variable dans le temps). En effet, cela évite de refaire les calculs plusieurs fois pour un résultat qui restera fixe.
- Elles ne doivent pas être **composées** : les données composées doivent être obtenues par la concaténation de données élémentaires conservées en base. Par exemple une adresse est obtenue à partir d'une rue, d'une ville et d'un code postal : ce sont ces trois dernières données qui sont conservées et donc qui figureront dans le MCD (et dans le dictionnaire des données).
- Lorsque l'on n'effectue jamais de calcul sur une donnée numérique, celle-ci doit être de type AN (c'est le cas par exemple pour un numéro de téléphone).

Exemple de Dictionnaire de donné d'un système de gestion des emprunts d'une bibliothèque

Tableau 2.1 : Dictionnaire de données d'un système de gestion des emprunts d'une bibliothèque

Code mnémonique	Désignation	Type	Taille	Remarque
id_i	Identifiant numérique d'un inscrit	N		
nom_i	Nom d'un inscrit	A	30	
prenom_i	Prénom d'un inscrit	A	30	
rue_i	Rue où habite un inscrit	AN	50	
ville_i	Ville où habite un inscrit	A	50	
cp_i	Code postal d'un inscrit	AN	5	
tel_i	Numéro de téléphone fixe d'un inscrit	AN	15	
tel_port_i	Numéro de téléphone portable d'un inscrit	AN	15	
email_i	Adresse e-mail d'un inscrit	AN	100	
date_naissance_i	Date de naissance d'un inscrit	Date	10	Au format AAAA-JJ-MM
id_l	Identifiant numérique d'un livre	N		
titre_l	Titre d'un livre	AN	50	
annee_l	Année de parution d'un livre	N	4	
resume_l	Résumé d'un livre	AN	1000	
ref_e	Code de référence d'un exemplaire d'un livre	AN	15	Cette référence servira également d'identifiant dans ce système
id_t	Identifiant numérique d'un type de livre	N		
libelle_t	Libellé d'un type de livre	AN	30	
id_ed	Identifiant numérique d'une édition de livre	N	6	
nom_ed	Nom d'une édition de livre	AN	30	
id_a	Identifiant numérique d'un auteur	N		
nom_a	Nom d'un auteur	A	30	
prenom_a	Prénom d'un auteur	A	30	
date_naissance_a	Date de naissance d'un auteur	Date		Au format AAAA-JJ-MM
id_p	Identifiant numérique d'un pays	N		
nom_p	Nom d'un pays	A	50	
id_em	Identifiant numérique d'un emprunt	N		
date_em	Date de l'emprunt	Date		Au format AAAA-11-MM

2.2.2-Elaboration du MCD

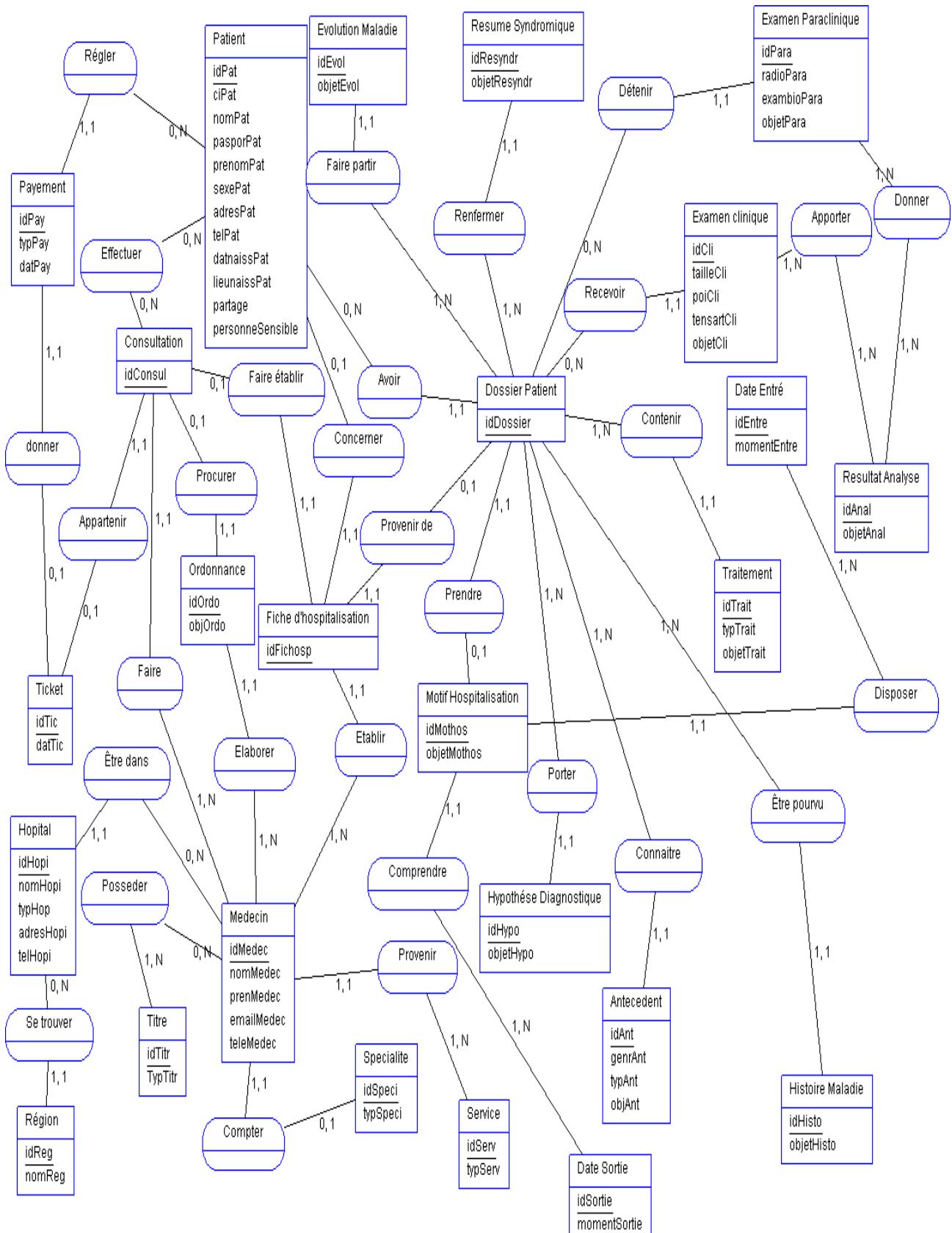


Figure 2.5 : MCD de gestion du dossier patient

2.2.2.1-Le Dictionnaire de Donnée

Tableau 2.2 : Dictionnaire de donnée de gestion du dossier patient

Code Mnémonique	Désignation	Type	Taille	Remarque
IdPat	Identifiant patient	N		
ciPat	Carte d'identité patient	N		
nomPat	Nom patient	A	50	
pasporPat	Passeport patient	A	150	
prenomPat	Prénom patient	A	150	
sexePat	Sexe patient	A	1	
adresPat	Adresse patient	AN	1000	
telPat	Téléphone patient	N		
datnaissPat	Date naissance patient	AN	10	
lieunaissPat	Lieu naissance patient	A	25	
idMedec	Identifiant médecin	N		
nomMedec	Nom médecin	D		Au format AAAA-MM-JJ
prenMedec	Prénom médecin	A	150	
emailMedec	Email médecin	AN	25	
teleMedec	Téléphone patient	N		
idConsul	Identifiant consultation	N		
idPay	Identifiant paiement	N		
typPay	Type paiement	A	24	
datPay	Date paiement	D		Au format AAAA-MM-JJ
idTic	Identifiant ticket	N		
datTic	Date ticket	D		Au format AAAA-MM-JJ
idHopi	Identifiant hôpital	N		
nomHopi	Nom hôpital	A	150	
typHopi	Type hôpital	A	30	
adresHopi	Adresse hôpital	AN	1000	
telHopi	Telephone hôpital	N		
idReg	Identifiant région	N		
nomReg	Nom région	A	150	
idTitr	Identifiant titre	N		
typTitr	Type titre	A	30	
idOrdo	Identifiant ordonnance	N		
objOrdo	Objet ordonnance	AN	10000	
idFichosp	Identifiant fiche hospitalisation	N		
idSpeci	Identifiant spécialité	N		
typSpeci	Type spécialité	A	20	
idEvol	Identifiant évolution maladie	N		
objetEvol	Objet évolution maladie	AN	1000	
idMothos	Identifiant motif d'hospitalisation	N		
objetMothos	Objet motif d'hospitalisation	AN	10000	

objetMothos	Objet motif d'hospitalisation	AN	10000	
idResyndr	Identifiant résumé syndromique	N		
objetResyndr	Objet résumé syndromique	AN	10000	
idDossier	Identifiant dossier patient	N		
idHypo	Identifiant hypothèse diagnostique	N		
objetHypo	Objet hypothèse diagnostique	AN	10000	
idServ	Identifiant service	N		
typServ	Type service	A	25	
idAnt	Identifiant antécédent	N		
genrAnt	Genre antécédent	A	25	
typAnt	Type antécédent	A	25	
objAnt	Objet antécédent	AN	10000	
idSortie	Identifiant sortie	N		
momentSortie	Moment sortie	D		Au format AAAA-MM-JJ
idHisto	Identifiant histoire maladie	N		
objetHisto	Objet histoire maladie	AN	10000	
idTrait	Identifiant traitement	N		
typTrait	Type traitement	A		
objetTrait	Objet traitement	AN	10000	
idAnal	Identifiant résultat analyse	N		
objetAnal	Objet résultat analyse	AN	10000	
idEntre	Identifiant entré	N		
momentEntre	Moment entré	D		Au format AAAA-MM-JJ
idCli	Identifiant examen clinique	N		
tailleCli	Taille examen clinique	AN	8	
poiCli	Poids examen clinique	N		
tensartCli	Tension artérielle examen clinique	AN	10	
objetCli	Objet examen clinique	AN	10000	
idPara	Identifiant paraclinique	N		
radioPara	Radio paraclinique			Ceci est l'ajout d'un fichier
exambioPara	Examen biologique paraclinique			Ceci est l'ajout d'un fichier
objetPara	Objet paraclinique	AN	10000	
partage	Partage information patient	booléen		
personneSensible	Donnée personne sensible	booléen		

2.3-Le Modèle Conceptuel des Traitements (MCT)

2.3.1-Formalisme du MCT

2.3.1.1-Définition

Le Modèle conceptuel de traitements permet de représenter l'enchaînement des Opérations du système d'information et des conditions de déclenchement de son activité par des événements extérieurs ou non au système.

Exemple de MCT de facturation

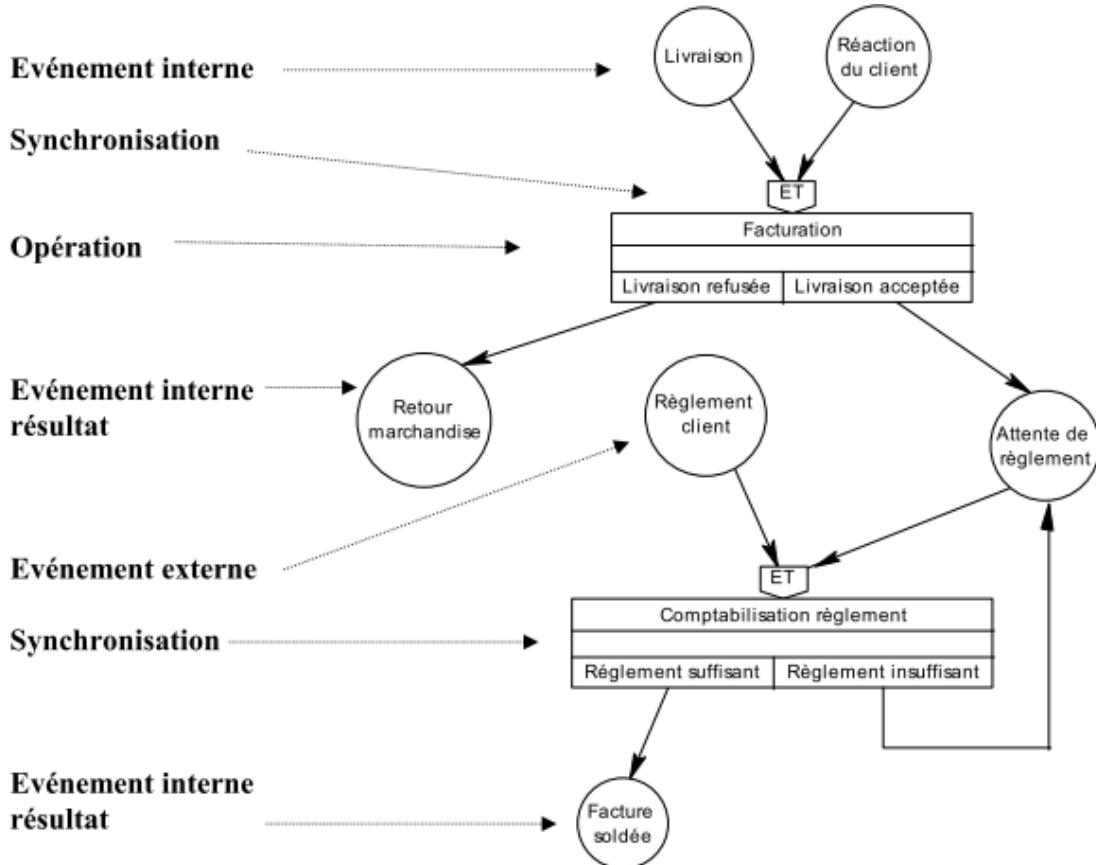


Figure 2.6 : Exemple MCT de facturation

Etude du processus de facturation

- Livraison des articles que le client a commandés.
- A la livraison, le client peut refuser la marchandise. Dans ce cas il y a retour de marchandise.
- Si le client accepte la livraison, la comptabilité émet une facture qui ne sera soldée qu'après règlement complet de celle-ci.
- Si le client n'a pas réglé complètement la facture il y aura une attente de règlement jusqu'au paiement complet de celle-ci.

2.3.1.2-Evénement

Un Evénement représente le fait que quelque chose s'est produit dans ou hors du système d'information. Il existe deux types d'événements :

- Externe : il provient de l'univers extérieur au système d'information. Il doit provoquer une nouvelle réaction du système d'information sous la forme d'une

opération.

- Interne : il provient de l'univers interne au système d'information.

Il peut provoquer une nouvelle réaction du système d'information sous la forme d'une opération, soit constituer un résultat pour l'univers extérieur.

Représentation graphique



Figure 2.7 : Formalisme d'un événement

2.3.1.3-Opération

Une opération est un « bloc de traitements », une succession de tâches qui doit s'exécuter sans interruption et dont l'activité aboutit à l'émission d'événements internes (poursuite de l'activité) ou externes (signaux pour extérieur du système d'information).

Représentation graphique



Figure 2.8 : Formalisme d'une Opération

2.3.1.4-Synchronisation

La Synchronisation d'une opération marque le rendez-vous des événements qui doivent arriver pour déclencher l'opération. Une synchronisation est de type logique (ET, OU).

Représentation graphique

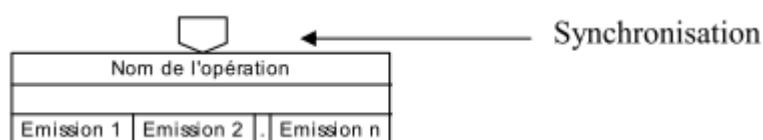


Figure 2.9 : Formalisme d'une Synchronisation

2.3.2-Elaboration du MCT

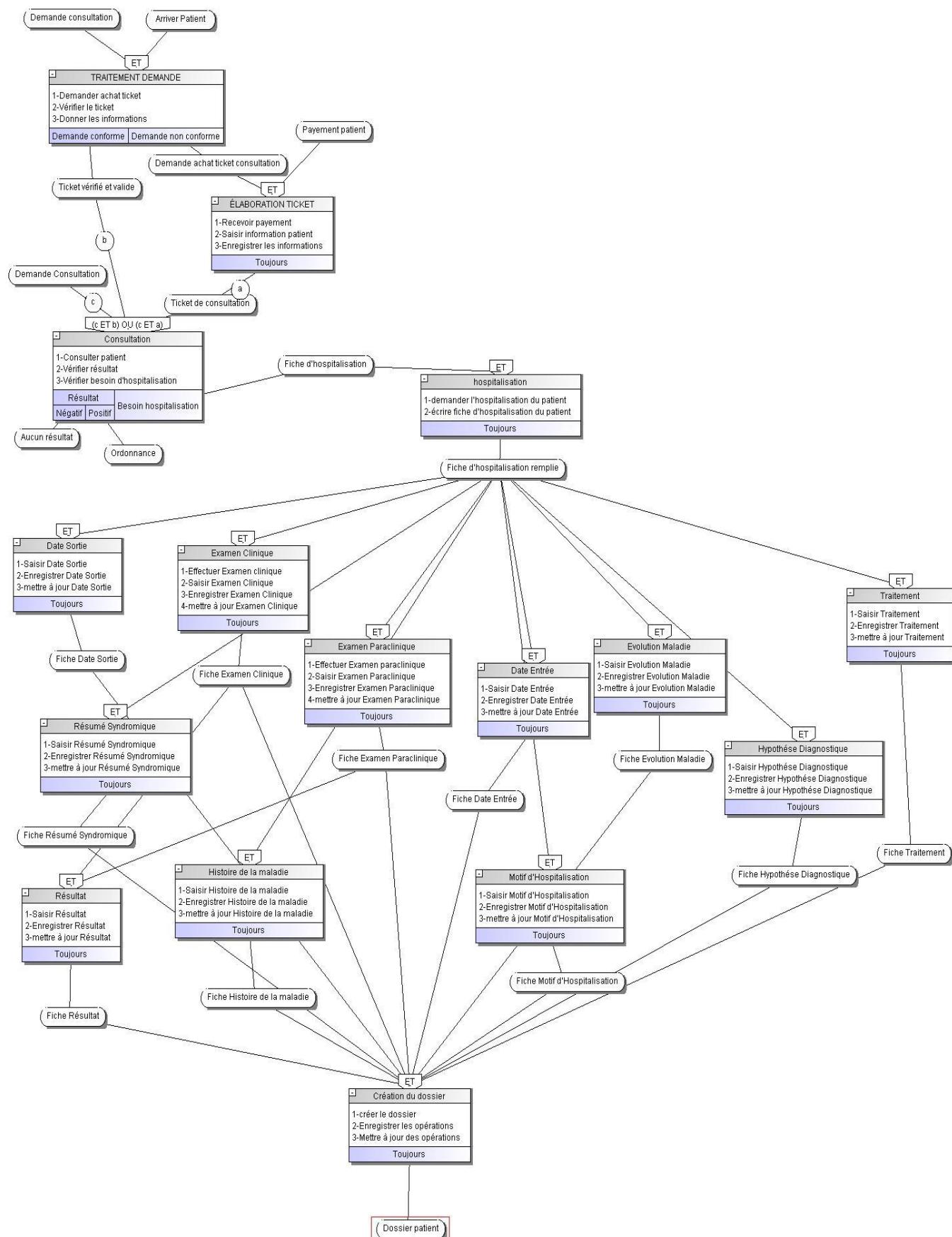


Figure 2.10 : MCT création du dossier patient

CHAPITRE 3 :

Etude détaillée

et

Implémentation

3.1-Etude détaillée

3.1.1-Definition

L'étude détaillée approfondit la variante décrite dans l'étude. Elle présente comment le système remplit les exigences. Des études détaillées peuvent être rédigées pour divers domaines thématiques. Des variantes de solutions peuvent être élaborées et évaluées dans l'étude détaillée. Dans les projets informatiques, les résultats des études détaillées sont récapitulés dans l'architecture du système. Ils constituent une annexe à celle-ci.

3.1.2-Le Modèle Organisationnel des Traitements (MOT)

3.1.2.1-Formalisme du MOT

3.1.2.1.1-Définition

Le modèle organisationnel des traitements (MOT) décrit l'organisation appliquée aux traitements en donnant les informations sur :

- la chronologie des procédures fonctionnelles de traitement ;
- la répartition organisationnelle des procédures fonctionnelles ;
- la circulation des informations entre les postes de travail ;
- les ressources humaines et techniques utilisées par la procédure fonctionnelle modélisée.

Le MOT s'appuie sur le formalisme du MCT. Il prend en compte les aspects organisationnels des traitements, c'est à dire :

- **Les postes de travail ou l'acteur** (qui exécute le traitement ? ex : secrétariat, expert, chef de service ...)
- **Le temps** (Quand une opération est-elle déclenchée ?)
- **La nature du traitement** (comment le traitement est-il réalisé ?) :

- **Manuel** : le traitement se réalise sans recours à l'ordinateur : ex : rédaction d'une fiche client,
 - **Automatique** : le traitement se fait sans intervention humaine (ex : transfert automatique des bons de livraison en facture, sauvegarde de la base de données)
 - **Conversationnel** : le traitement repose sur un dialogue homme-machine interactif (saisie dans un formulaire avec affichage automatique des éléments calculés), On parle aussi de tâche interactive ou en temps réel
- **La périodicité des traitements** : journalier, hebdomadaire, mensuel, trimestriel, semestriel, annuel
- Et éventuellement **la durée**

Son formalisme se présente comme suit :

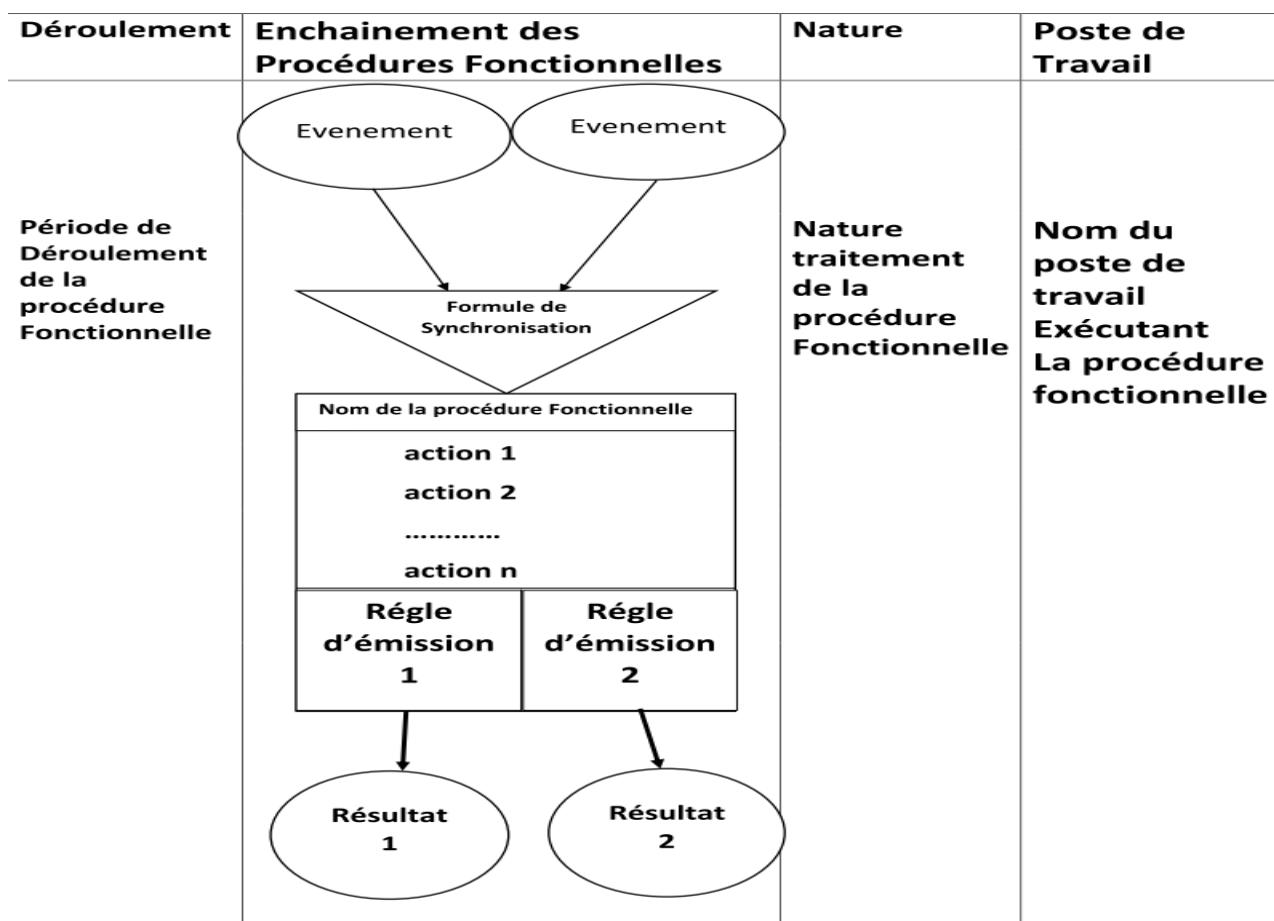
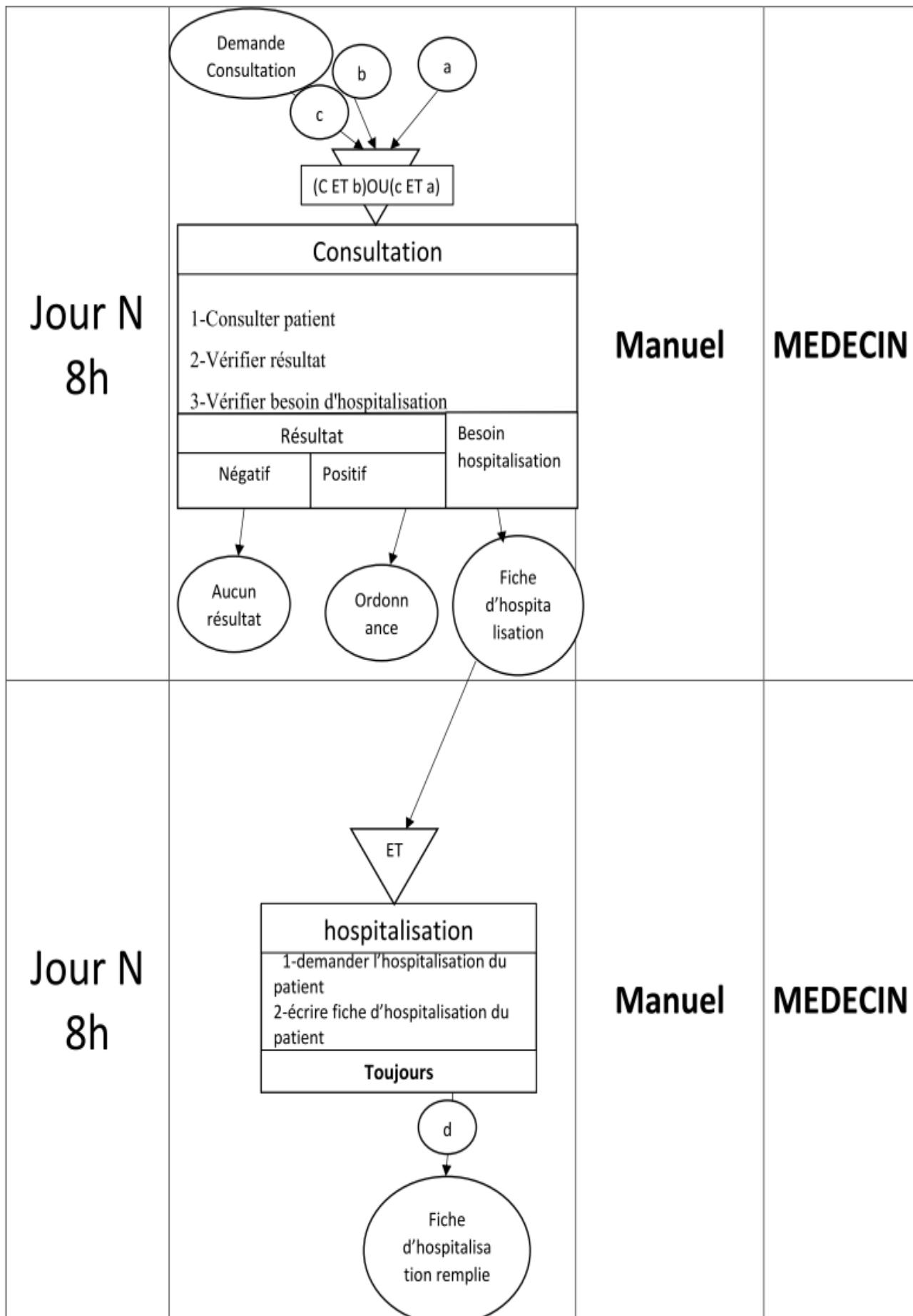


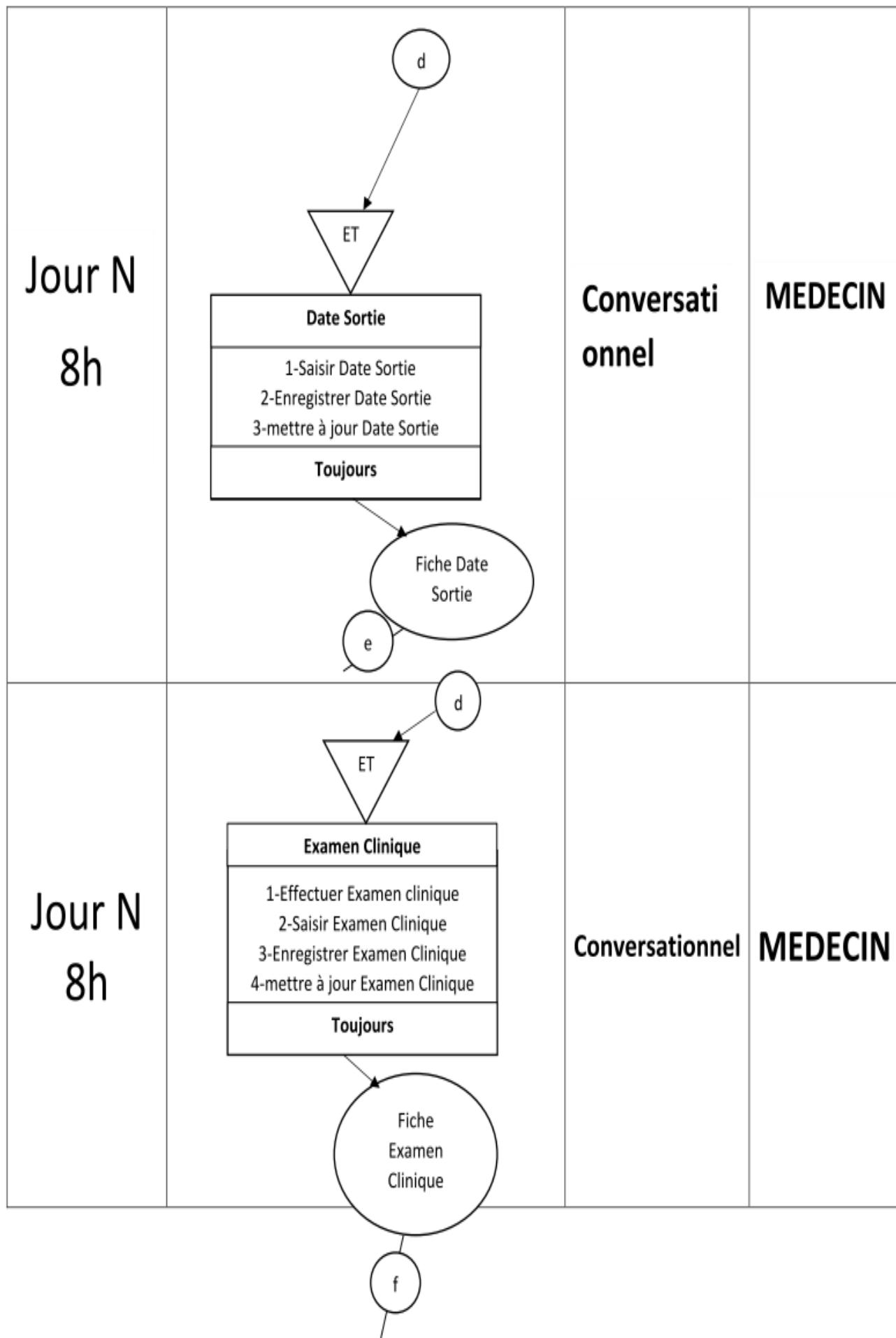
Figure 3.1 :Formalisme d'un MCT

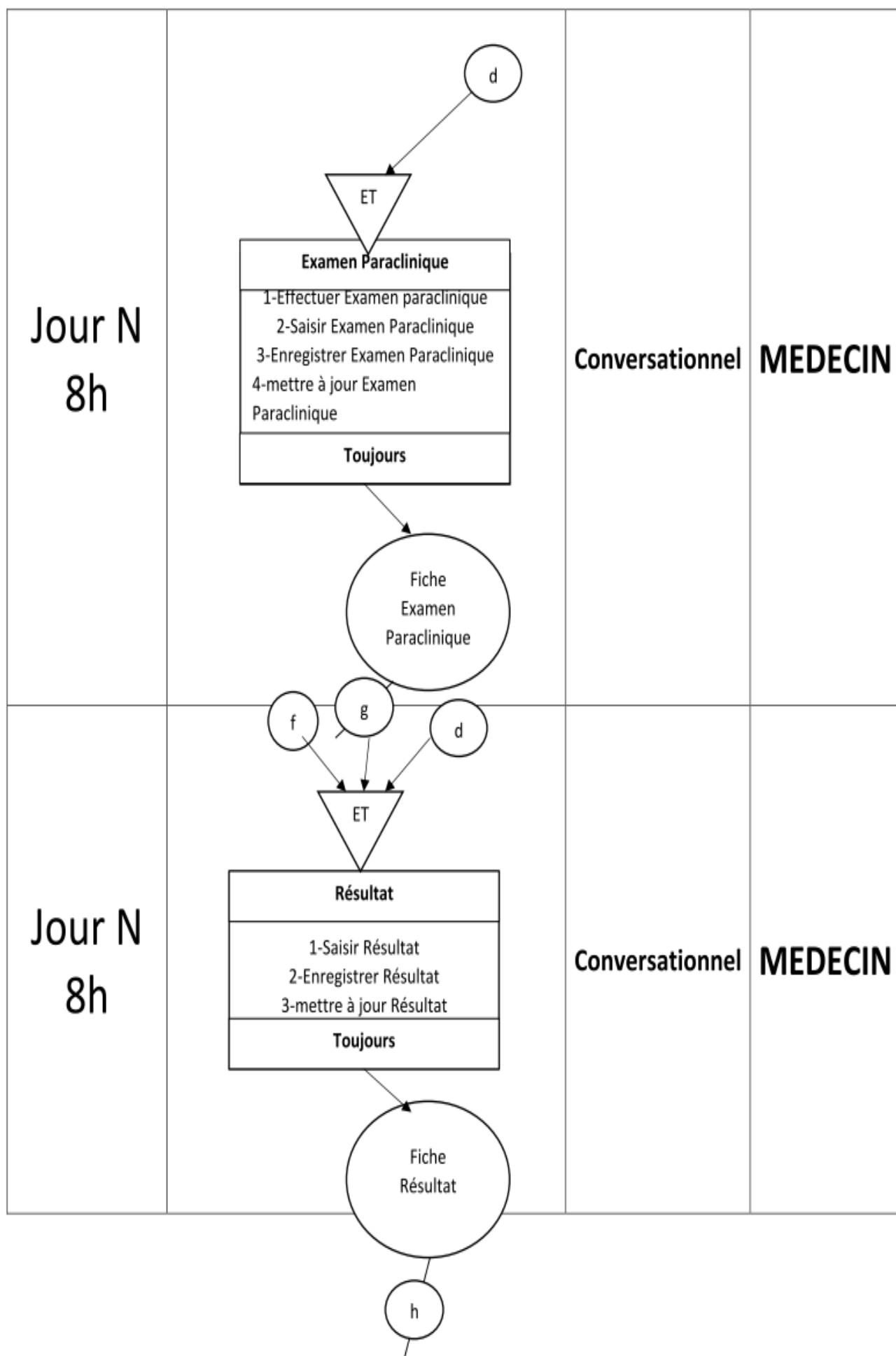
3.1.2.2-Elaboration du MOT

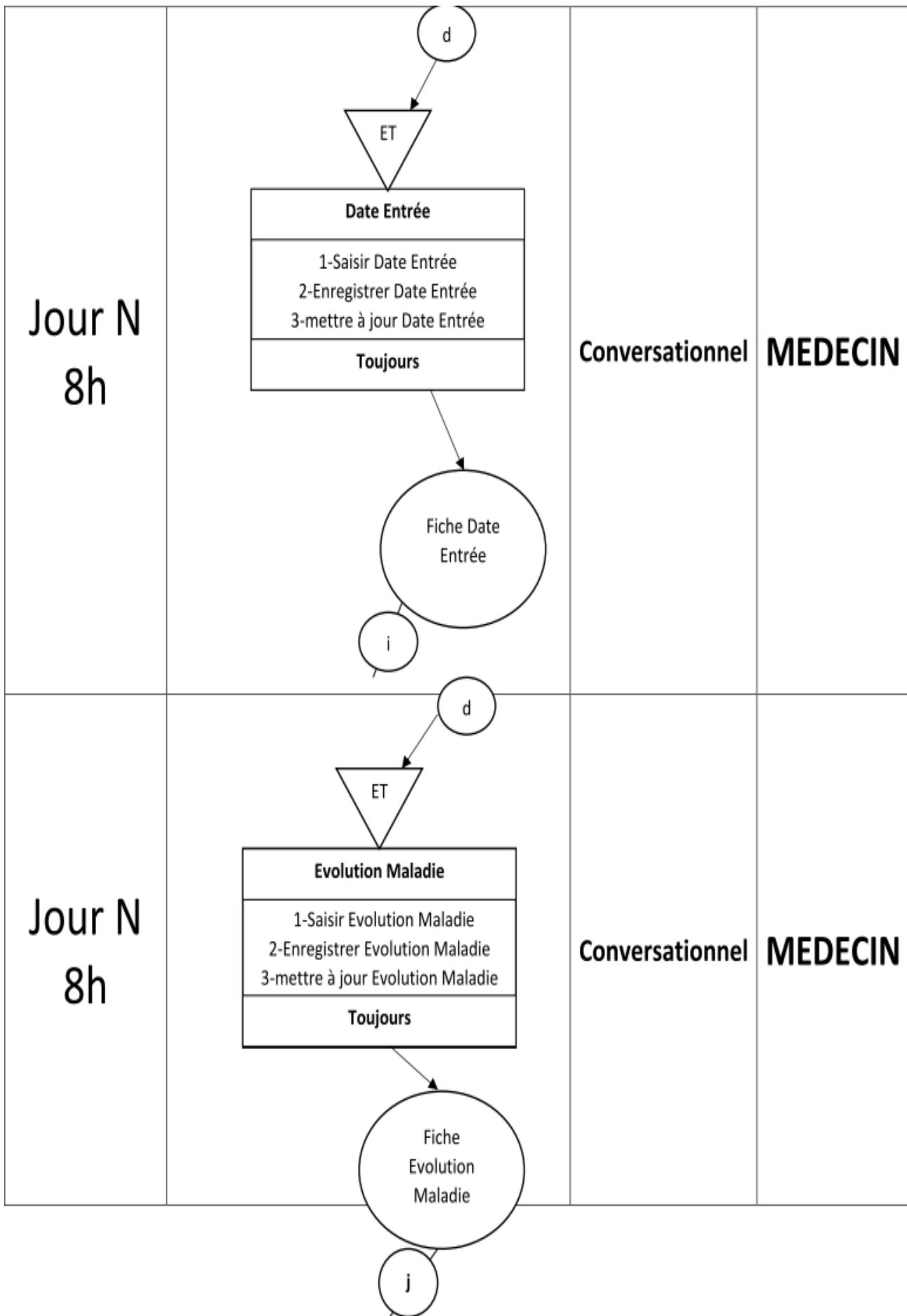
Tableau 3.1 : MOT Crédation du dossier patient

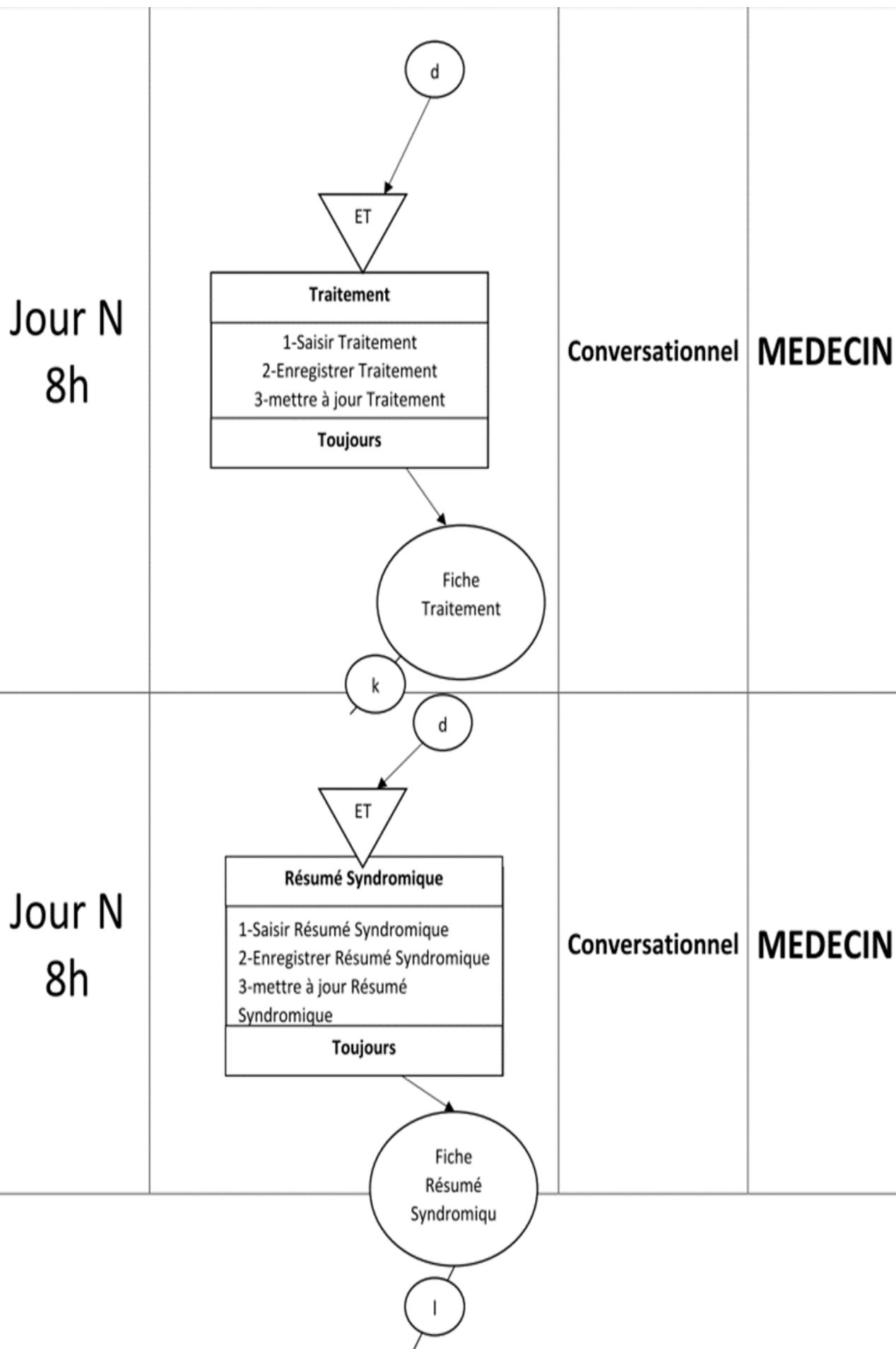
Déroulement	Enchainement des Procédures Fonctionnelles	Nature	Poste de travail
Jour N 8h	<pre> graph TD A([Demande de consultation]) --> ET1{ET} B([Arriver Patient]) --> ET1 ET1 --> C[TRAITEMENT DEMANDE] C --> D[1-Demander achat ticket
2-Vérifier le ticket
3-Donner les informations] D --> E[Demande conforme Demande non conforme] E --> F([Ticket vérifié et valide]) E --> G([Demande achat ticket consultation]) </pre>	Manuel	ACCUEIL
Jour N 8h	<pre> graph TD H([Payment patient]) --> ET2{ET} ET2 --> I[ÉLABORATION TICKET] I --> J[1-Recevoir paiement
2-Saisir information patient
3-Enregistrer les informations] J --> K[TOUJOURS] K --> L([Ticket de consultation]) </pre> <p>Diagram notes: - A circle labeled 'b' is connected to the bottom of the vertical line leading down from the 'Payment patient' oval. - A circle labeled 'a' is connected to the bottom of the vertical line leading down from the 'Ticket de consultation' oval.</p>	Conversationnel	CAISSE

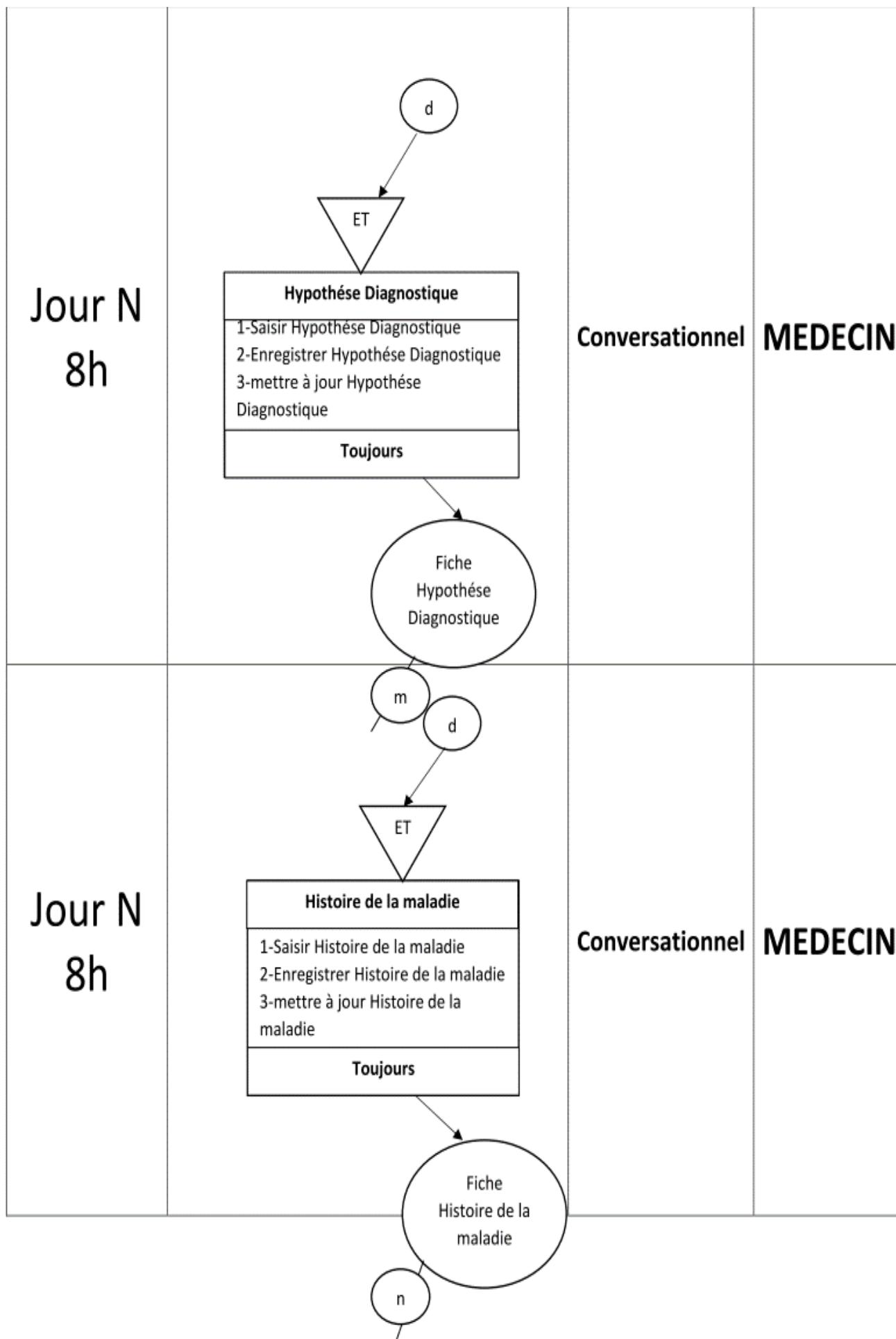


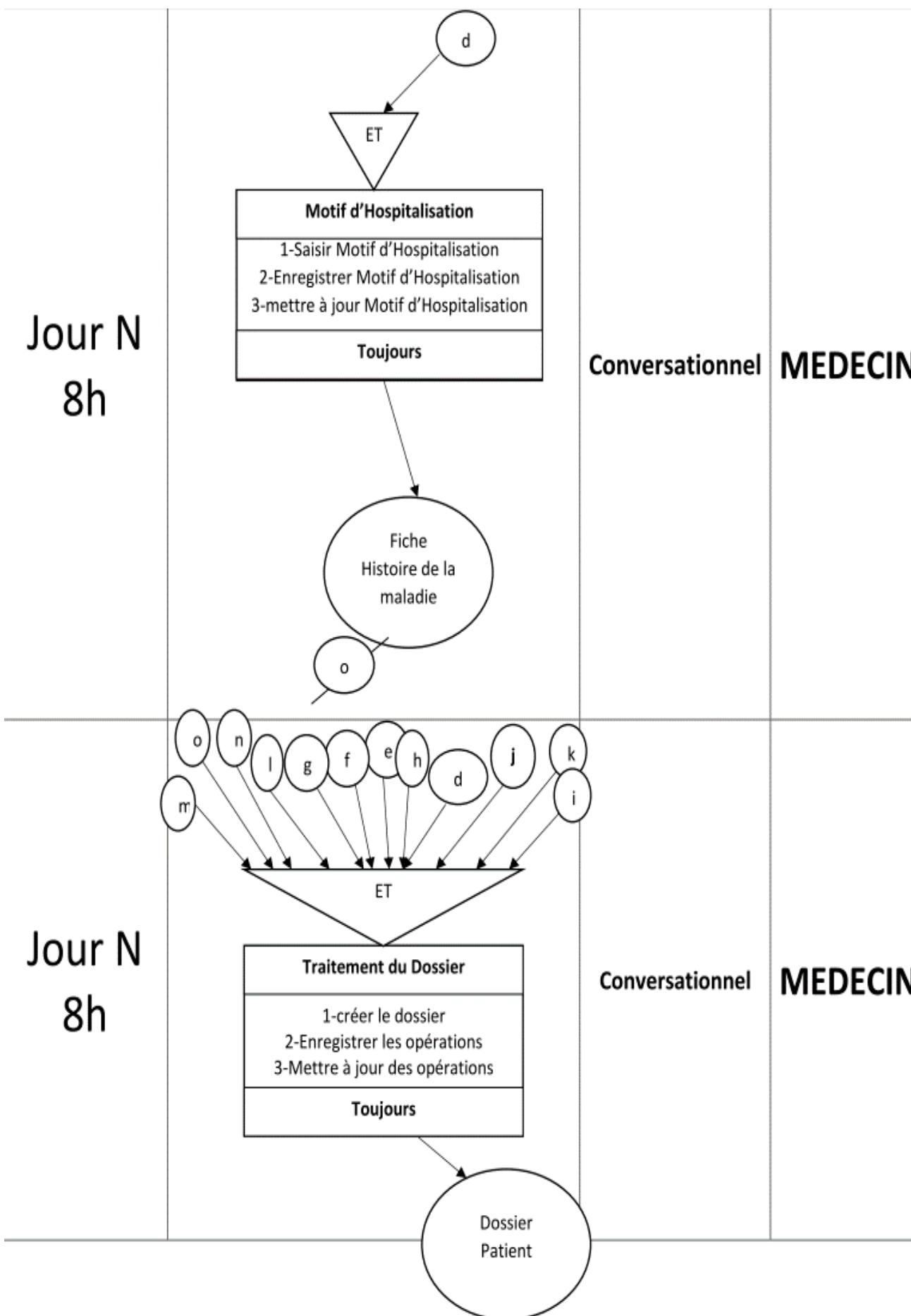












3.2-Implémentation

3.2.1-Introduction

L'implémentation est la réalisation, l'exécution ou la mise en pratique d'un plan, d'une méthode ou bien d'un concept, d'une idée, d'un modèle, d'une spécification, d'une norme ou d'une règle dans un but précis. L'implémentation est donc l'action qui doit suivre une réflexion pour la concrétiser.

Dans le contexte des technologies de l'information , l'implémentation d'un logiciel ou matériel englobe tous les processus après-vente impliqués dans le bon fonctionnement d'un élément dans son environnement, y compris l'analyse des besoins, l'installation, la configuration, la personnalisation, l'exécution, le test, l'intégration des systèmes, la formation des utilisateurs, la livraison et l'application d'éventuelles modifications. Le terme « déploiement » est parfois employé dans le même sens.

Pour qu'une implémentation fonctionne, de nombreuses tâches doivent être effectuées successivement dans différents services. Les entreprises s'efforcent d'utiliser des méthodes éprouvées et de faire appel à des professionnels pour les aider à implémenter un système. Cependant, de nombreux processus d'implémentation échouent par manque de planification appropriée en début de projet, en raison de ressources inadaptées ou de problèmes imprévus.

3.2.2-Le Système de gestion de base de données (SGBD)

3.2.2.1-Definition

Un système de gestion de base de données (abr. SGBD) est un logiciel système servant à stocker, à manipuler ou gérer, et à partager des informations dans une base de données, en garantissant la qualité, la pérennité et la confidentialité des informations, tout en cachant la complexité des opérations.

Un SGBD (en anglais DBMS pour database management system) permet d'inscrire, de retrouver, de modifier, de trier, de transformer ou d'imprimer les informations de la base de données. Il permet d'effectuer des comptes rendus des informations enregistrées et comporte des mécanismes pour assurer la cohérence des informations, éviter des pertes d'informations dues à des pannes, assurer la confidentialité et

permettre son utilisation par d'autres logiciels. Selon le modèle, le SGBD peut comporter une simple interface graphique jusqu'à des langages de programmation sophistiqués.

Les systèmes de gestion de base de données sont des logiciels universels, indépendants de l'usage qui est fait des bases de données². Ils sont utilisés pour de nombreuses applications informatiques, notamment les guichets automatiques bancaires, les logiciels de réservation, les bibliothèques numériques, les logiciels d'inventaire, les progiciels de gestion intégrés ou la plupart des blogs et sites web. Il existe de nombreux systèmes de gestion de base de données.

Les SGBD sont souvent utilisés par d'autres logiciels ainsi que les administrateurs ou les développeurs. Ils peuvent être sous forme de composant logiciel, de serveur, de logiciel applicatif ou d'environnement de programmation.

Exemple de SGBD : MySQL(que nous utiliserons), MariaDB, Firebird, SQLite, Oracle Database.

3.2.2.2-But visé

Les SGBD sont les logiciels intermédiaires entre les utilisateurs et les bases de données. Une base de données est un magasin de données composé de plusieurs fichiers manipulés exclusivement par le SGBD. Ce dernier cache la complexité de manipulation des structures de la base de données en mettant à disposition une vue synthétique du contenu.

L'ensemble SGBD et base de données est destiné à permettre le stockage de données d'une manière offrant de nombreux avantages par rapport à un enregistrement conventionnel dans des fichiers. Il permet d'obtenir et de modifier rapidement des données, de les partager entre plusieurs usagers. Il garantit l'absence de redondance, l'intégrité, la confidentialité et la pérennité des données tout en donnant des moyens d'éviter les éventuels conflits de modification et en cachant les détails du format de fichier des bases de données.

Les données sont enregistrées sous forme de suites de bits représentant des lettres, des nombres, des couleurs, des formes, ... Le SGBD comporte différents mécanismes destinés à retrouver rapidement les données et de les convertir en vue d'obtenir des informations qui aient un sens.

- À l'aide du SGBD plusieurs usagers et plusieurs logiciels peuvent accéder simultanément aux données. Le SGBD effectue les vérifications pour assurer qu'aucune personne non autorisée n'ait accès à des données confidentielles contenues dans la base de données, il arbitre les collisions lorsqu'il y a plusieurs modifications simultanées de la même information et comporte des mécanismes en vue d'éviter des pertes de données à la suite d'une panne ;
- La redondance désigne une situation de présence de plusieurs copies de la même donnée dont la modification peut amener à des incohérences se manifestant par des copies différentes. Le SGBD vérifie - voire refuse - la présence de redondances. Le SGBD effectue également sur demande des vérifications pour assurer que les données introduites soient correctes (valeurs dans les limites admises, format correct) et que les données soient cohérentes par rapport à ce qui se trouve déjà dans la base de données ;
- Les données sont typiquement manipulées par un logiciel applicatif qui fait appel aux services du SGBD pour manipuler la base de données. Alors qu'un logiciel applicatif qui manipule un fichier tient compte du format de données de ce fichier, un logiciel qui manipule une base de données par l'intermédiaire d'un SGBD n'a pas connaissance du format de la base de données, les données sont présentées par le SGBD sous une forme qui cache les détails du format des fichiers dans lesquels elles sont enregistrées.

3.2.2.3-Construction et fonctionnement

Un SGBD est composé de nombreux programmes, parmi lesquels le moteur, le catalogue, le processeur de requêtes, le langage de commande et des outils :

- le moteur de base de données est le cœur du SGBD, il manipule les fichiers de la base de données, transmet les données depuis et vers les autres programmes, et vérifie la cohérence et l'intégrité des données ;
- Un programme manipule le catalogue : le magasin qui contient la description de l'organisation de la base de données, les listes de contrôle d'accès, le nom des personnes autorisées à manipuler la base de données et la description des règles de cohérence (contraintes). Selon les modèles de SGBD ces informations peuvent être

modifiées en utilisant le langage de commande, ou alors à l'aide d'une interface graphique ;

- Le processeur de requête exécute les opérations demandées. Selon les modèles de SGBD, ces opérations peuvent être formulées dans un langage de commande, ou à l'aide d'une interface graphique du type QBE (Query by Example, en français requête par l'exemple) ;
- La majorité des SGBD comportent au moins un langage de commande. Ce langage de requête permet de manipuler le contenu de la base de données. Reconnu par la majorité des SGBD du marché, SQL est devenu le langage standard de facto.

Les outils du SGBD servent à créer des comptes rendus (reports), des écrans pour la saisie des informations, importer et exporter les données de et vers la base de données, et manipuler le catalogue. Ces outils sont utilisés par l'administrateur de bases de données pour effectuer des sauvegardes, des restaurations de données, autoriser ou interdire l'accès à certaines informations, et effectuer des modifications du contenu de la base de données - création, lecture, modification et suppression d'informations, abrégé CRUD(anglais create, read, update, delete). Ces outils servent également à surveiller l'activité du moteur et effectuer des opérations de tuning.

Les SGBD contemporains de haut de gamme comportent de nombreuses extensions qui offrent des fonctionnalités auxiliaires, leur construction reste cependant similaire à la plupart des SGBD.

Dans un SGBD relationnel, les demandes formulées au SGBD sont typiquement traitées en cinq étapes :

- les logiciels clients communiquent avec le SGBD en utilisant son interface de programmation via un réseau. Un dispositif de communication du SGBD vérifie l'identité du client, puis transmet les requêtes du client vers le noyau du SGBD et transmet au client les informations extraites par le SGBD ;
- le SGBD crée ensuite un thread en vue de traiter la requête. Un programme contrôle l'ensemble des threads et décide lesquels sont exécutés immédiatement et lesquels seront exécutés plus tard, en fonction de la charge de travail de l'ordinateur ;
- lors de l'exécution du thread, un compilateur transforme le texte exprimé dans le langage de requête du SGBD en un plan d'exécution dont la forme imite celle d'une

expression algébrique utilisant l'algèbre relationnelle, puis un ensemble de programmes "opérateurs" calculent le résultat de l'expression en effectuant des opérations telles que la jointure, le produit cartésien, le tri et la sélection ;

- Les opérateurs font appel au moteur de base de données, celui-ci exécute des algorithmes (appelés access method en anglais) en vue de retrouver les informations et entretenir les structures des fichiers de la base de données ;
- Une fois les informations obtenues par le programme de manipulation de fichiers, celles-ci sont envoyées au thread d'exécution puis au dispositif de communication qui les transmet au client.

3.2.3-Les langages et choix

3.2.3.1-Le SQL

Le **SQL** (Structured Query Language, traduisez Langage de requêtes structuré) est un langage de définition de données (LDD, ou en anglais DDL Data Definition Language) c'est-à-dire qu'il permet de créer des tables dans une base de données relationnelle, ainsi que d'en modifier ou en supprimer ;

Un langage de manipulation de données (LMD, ou en anglais DML, Data Manipulation Language) cela signifie qu'il permet de sélectionner, insérer, modifier ou supprimer des données dans une table d'une base de données relationnelle ;

et un langage de contrôle de données (LCD, ou en anglais DCL, Data Control Language) qui permet de définir des permissions au niveau des utilisateurs d'une base de données relationnelles.

Le modèle relationnel a été inventé par E.F. Codd (Directeur de recherche du centre IBM de San José) en 1970, suite à quoi de nombreux langages ont fait leur apparition :

- IBM Sequel (Structured English Query Language) en 1977
- IBM Sequel/2
- IBM System/R
- IBM DB2

Ce sont ces langages qui ont donné naissance au standard SQL, normalisé en 1986 par l'ANSI pour donner SQL-86 ou SQL-87. Puis en 1989 la version SQL-89 ou SQL-1 a été approuvée. La norme SQL-92 a désormais pour nom SQL 2. En 1999 nous avons la version SQL-99 ou SQL3. En 2003, la version SQL:2003. En 2008, la version SQL:2008. Et en 2011 nous avons la version SQL:2011 qui est la dernière version que nous utilisons pour le moment. Ses différentes versions sont dues à la révision du langage avec l'ajout, la modification, la suppression de certaines fonctionnalités dans le langage.

Exemple de Code SQL

```

drop table t1
Create table t1 (tim int, rem varchar(100))
select 86400
INSERT INTO t1 VALUES (1251781074, 'day1')
INSERT INTO t1 VALUES (1251781074 + 86400, 'day2')
INSERT INTO t1 VALUES (1251781074 + 2*86400, 'day3')
INSERT INTO t1 VALUES (1251781074 + 3*86400, 'day4')
INSERT INTO t1 VALUES (1251781074 + 4*86400, 'day5')

Select DATEADD(hour,-4,(dateadd(second ,tim, '1/1/1970'))) , * From t1

DECLARE @StartTime DATETIME
,@EndTime DATETIME

SELECT @StartTime = '2009-09-02 00:57:54.000'
SELECT @EndTime = '2009-09-03 00:57:54.000'

Select * from t1
Where
DATEADD(hour,-4,(dateadd(second ,tim, '1/1/1970')))) >= @StartTime
AND DATEADD(hour,-4,(dateadd(second ,tim, '1/1/1970')))) <= @EndTime

```

Figure 3.2 : Exemple de Code SQL

3.2.3.2-Le Html

Le **HTML** (« HyperText Mark-Up Language ») est un langage dit de « marquage » (de « structuration » ou de « balisage ») dont le rôle est de formaliser l'écriture d'un document avec des balises de formatage. Les balises permettent d'indiquer la façon dont doit être présenté le document et les liens qu'il établit avec d'autres documents.

Par définition, toutes les pages Web sont faites de code HTML. Nous avons eu plusieurs versions du langage, et la dernière version actuelle est le HTML 5.

Exemple de Code HTML

```

1  <!DOCTYPE html>
2  <html>
3      <head>
4          <title>Example</title>
5          <link rel="stylesheet" href="sty
6      </head>
7      <body>
8          <h1>
9              <a href="/">Header</a>
10         </h1>
11         <nav>
12             <a href="one/">One</a>
13             <a href="two/">Two</a>
14             <a href="three/">Three</a>
15         </nav>

```

Figure 3.3 : Exemple de Code HTML

3.2.3.3-Le CSS

Le CSS (Cascading Style Sheet) est un langage informatique servant à décrire la présentation et le style à l'aide d'un sélecteur, d'une propriété et d'une valeur un document HTML et XML.

Datant des années 90, ce langage sert principalement au développement de sites web. Nous avons eu plusieurs versions du langage, et la dernière version actuelle et le CSS3.

Exemple de Code CSS

```

body{
    margin: 0px;
    padding: 50px;
    font-size: 14px;
    font-family: Verdana, Cursiva, serif;
}

/*div centré dans la page*/
div{
    margin: 30px auto;
    width: 80%;
    padding: 15px;
    border: 10px solid #0D8;
}

/*Paragraphe centré dans le div, et texte centré
 *dans le paragraphe (dans l'élément p)*/
p{
    margin: 20px auto;
    width: 60%;
    padding: 5px;
    border: 3px solid #39D;
    text-align: center;
}

```

Figure 3.4 : Exemple de Code CSS

3.2.3.4-Le PHP

Le langage PHP a été créé en 1994 par Rasmus Lerdorf pour son site web. C'était à l'origine une bibliothèque logicielle en C dont il se servait pour conserver une trace des visiteurs qui venaient consulter son CV. Au fur et à mesure qu'il ajoutait de nouvelles fonctionnalités, Rasmus a transformé la bibliothèque en une implémentation capable de communiquer avec des bases de données et de créer des applications dynamiques et simples pour le Web. Rasmus a alors décidé, en 1995, de publier son code, pour que tout le monde puisse l'utiliser et en profiter. PHP s'appelait alors PHP/FI (pour Personal Home Page Tools/Form Interpreter). En 1997, deux étudiants, Andi Gutmans et Zeev Suraski, ont redéveloppé le cœur de PHP/FI. Ce travail a abouti un an plus tard à la version 3 de PHP, devenu alors PHP : Hypertext Preprocessor. Peu de temps après, Andi Gutmans et Zeev Suraski ont commencé la réécriture du moteur interne de PHP. C'est ce nouveau moteur, appelé Zend Engine (le mot Zend est la contraction de Zeev et Andi qui a servi de base à la version 4 de PHP).

Au fur et à mesure PHP , est devenu un langage de programmation libre, principalement utilisé pour produire des pages Web dynamiques via un serveur HTTP, mais pouvant également fonctionner comme n'importe quel langage interprété de façon locale. PHP est un langage impératif orienté objet. PHP a permis de créer un grand nombre de sites web célèbres, comme Facebook, Wikipédia, etc.

Il est considéré comme une des bases de la création de sites web dits dynamiques mais également des applications web.

Nous avons eu plusieurs versions de PHP, que l'on peut diviser en deux parties celles dites majeur ou versions principales : PHP 3, PHP 4, PHP 5, PHP 6, PHP 7

Et ceux dites mineurs appelés release ou sous versions : PHP 3.X, PHP 4.X, PHP 5.X, PHP 6.X, PHP 7.X.

Exemples de release de PHP 5 : PHP 5.0, PHP 5.1, PHP 5.2.4, PHP 5.2.5, PHP 5.2.6, PHP 5.2.9, PHP 5.2.10, PHP 5.2.13, PHP 5.2.14, PHP 5.2.17.

Ses différentes versions sont des améliorations du langage tant sur le niveau sécurité avec la correction des bugs ; ajout, suppression, modification des fonctions du langage, mais aussi des performances de ce dernier. Toutes ses améliorations peuvent être consulté au niveau du site web de PHP : www.secure.php.net/

Exemple de Code PHP

```
<?php
if (!isset($_POST['categorie']))
{
    include("formulaire_recherche.php");
}

// Pour une recherche sans mot_cle

elseif (!isset($_POST['mot_cle1']) AND !isset($_POST['mot_cle2']) AND !isset($_POST['mot_cle3']))
{
    $reponse = $bdd->prepare('
        SELECT * FROM document WHERE categorie = ?') or die(print_r($bdd->errorInfo()));

    $reponse->execute(array(
        $_POST['categorie']));

    $donnees = $reponse->fetch();
    print_r($donnees);
    var_dump($donnees);

    if (empty($donnees)) {
        include("formulaire_recherche.php");
        echo "Aucun document ne correspond à cette catégorie";
    }
}
```

Figure 3.5: Exemple de Code PHP

3.2.3.5-Le JavaScript

Le **JavaScript** est un langage de script incorporé dans un document HTML. Historiquement il s'agit même du premier langage de script pour le Web. Ce langage est un langage de programmation qui permet d'apporter des améliorations au langage HTML en permettant d'exécuter des commandes du côté client, c'est-à-dire au niveau du navigateur. Ainsi le langage JavaScript est fortement dépendant du navigateur appelant la page web dans laquelle le script est incorporé, mais en contrepartie il ne nécessite pas de compilateur, contrairement au langage Java, avec lequel il a longtemps été confondu. JavaScript a été mis au point par Brendan Eich pour le compte de Netscape en 1995. A l'origine, il se nommait moka puis LiveScript et était destiné à fournir un langage de script simple au navigateur Netscape Navigator 2. Il est à l'époque longtemps été critiqué pour son manque de sécurité, son développement peu poussé et l'absence de messages d'erreur explicites rendant dure son utilisation. Le 4 décembre 1995, suite à une association avec le constructeur Sun, Netscape rebaptise son langage JavaScript (un clin d'œil au langage Java développé par Sun). A la même époque, Microsoft mit au point le langage Jscript, un langage de script très similaire. Ainsi, pour éviter des dérives de part et d'autre, un standard a été défini pour normaliser les langages de script, il s'agit de l'ECMA 262, créé par l'organisation du même nom (ECMA, European Computer Manufacturers Association).

Le JavaScript est aussi utilisé côté serveur comme PHP .net ...etc il est même possible d'utiliser JavaScript comme langage serveur grâce à Node js.

Exemple de Code JavaScript

```

1  function quickSort(items, left, right) {
2      var index = 0;
3      if (items.length > 1) {
4          left = typeof left != 'number' ? 0 : left;
5          right = typeof right != 'number' ? items.length - 1 : right;
6          index = partition(items, left, right);
7          if (left < index - 1) {
8              quickSort(items, left, index - 1);
9          }
10         if (index < right) {
11             quickSort(items, index, right);
12         }
13     }
14     return items;
15 }
16 // first call
17 var result = quickSort(items);

```

Figure 3.6 : Exemple de Code JavaScript

3.2.4-Choix de l'outils de conception du système

Le choix de nos outils pour l'élaboration du système se fera avec des langages de programmation dits statiques comme le **HTML ou HyperText Markup Language** (Pour le contenu de nos page), le **CSS ou Cascading Style Sheets** (pour le style de nos pages), le **Framework Bootstrap**, qui est une collection d'outils utiles à la création du design (graphisme, animation et interactions avec la page dans le navigateur ... etc. ...) de sites et d'applications web et **JavaScript** qui est un langage de programmation de scripts pour les pages web interactives.

Elle se fera aussi grâce à des langages de type dynamique comme PHP ou HyperText Preprocessor, qui nous permettra de communiquer avec le serveur, de sécuriser notre page etc...Grâce à sa large communauté de développeurs, de sa documentation très fournie, de son ancienneté et des multiples possibilités qu'elle apporte notre choix peut se tourner qu'envers ce dernier.

Il y'a aussi l'utilisation du logiciel **xampp** qui est un ensemble de logiciels permettant de mettre en place facilement un serveur Web et un serveur FTP. Il s'agit d'une distribution de logiciels libres (MySQL, PHP, apache qui gère les requêtes http) offrant

une bonne souplesse d'utilisation, réputée pour son installation simple et rapide. Ainsi, il est à la portée d'un grand nombre de personnes puisqu'il ne requiert pas de connaissances particulières et fonctionne, de plus, sur les systèmes d'exploitation les plus répandus. Cette « distribution » se chargera donc d'installer l'ensemble des outils dont vous pourriez avoir besoin lors de la création d'un site Web. Plus d'une dizaine d'utilitaires sont intégrés, comme MySQL, PHP, Perl ou encore phpMyAdmin (qui gère la base de données).

Il y'a aussi l'utilisation d'un éditeur de texte comme **sublime text** qui propose des fonctions avancées comme la coloration syntaxique personnalisable, l'auto complétion, un système de plugins... Sublime text propose encore des fonctions plus avancées.

3.2.4.1-Certaines failles web et comment s'en protéger

3.2.4.1.1-La faille XSS

Le XSS consiste à injecter du code arbitraire dans une application web et `à le faire exécuter dans un navigateur web ou dans n'importe quelle autre solution permettant l'interprétation de code HTML [2,3,4,5,6,7]. C'est une faille de sécurité liée aux technologies du web, avec une présence possible sur n'importe quelle application indépendamment du langage utilisé (PHP, ASP, JSP, etc.) ou des technologies (serveurs, SGBD, etc.). Le XSS est une technique d'attaque côté client, et se sert du navigateur de ses victimes afin d'exécuter du code malveillant.

Comment s'en protéger

La solution la plus adaptée contre cette faille est d'utiliser la fonction `htmlspecialchars()` qui est une fonction PHP .Cette fonction permet de filtrer les symboles du type <, & ou encore ", en les remplaçant par leur équivalent en HTML.
Par exemple :

- Le symbole & devient &
- Le symbole " devient "
- Le symbole ' devient '
- ...

3.2.4.1.2- La faille Include

La faille include est également très dangereuse. Comme son nom l'indique, elle exploite une mauvaise utilisation de la fonction `include()`. Cette fonction est très souvent utilisée pour exécuter du code PHP se situant dans une autre page, et particulièrement pour la connexion à une base de données. Par exemple :

```
<?php include('config.php'); ?>
```

La page config.php contient les variables avec les renseignements de connexion à la base de données. Avant tout, il faut savoir que cette faille est extrêmement dangereuse, car le hacker peut se servir de cette négligence pour faire ce qui lui plaît comme modifier, supprimer, changer les droits... Il peut également forcer l'affichage de votre code PHP. Autant vous dire que vous avez tout intérêt à vous en protéger ! Il existe de très nombreuses manières d'utiliser la faille `include()`.

Comment s'en protéger

Pour commencer, il faut d'abord tester la présence de la faille include sur votre site. Pour cela rien de plus simple, il vous suffit d'essayer d'inclure une page qui n'existe pas. Si l'URL en question est vulnérable, vous obtiendrez un message d'erreur de la part de PHP. Cette faille est déjà plus complexe à protéger, dans la mesure où on ne peut pas complètement interdire l'inclusion sur le site (car sinon on ne pourra pas inclure nos propres fichiers). Il faut donc savoir au préalable ce que l'on veut inclure, rédiger un `htaccess` qui veille sur les fichiers importants.

htaccess : Fichier de configuration disponible pour les serveurs Apache. Il permet par exemple de gérer les droits d'accès au site (pages, dossiers...). Il permet également la réécriture d'URL.

On peut aussi faire s'assurer que les pages que l'on inclut sont bien sur notre serveur et non sur un autre. Pour ce faire, on peut avoir recours à la fonction `file_exists ('nom de fichier')` qui renvoie vrai si le fichier existe sur notre serveur, faux si non.

Plusieurs méthodes existent pour se protéger de cette faille, très puissante, elle est encore présente sur un nombre considérable de sites web.

3.2.4.1.3- L'injection SQL

L'injection SQL est une méthode d'attaque très connue. C'est un vecteur d'attaque extrêmement puissant quand il est bien exploité. Il consiste à modifier une requête SQL en injectant des morceaux de code non filtrés non prévu par le système, généralement par le biais d'un formulaire et pouvant en compromettre la sécurité.

Cette dernière peut permettre de manipuler une base de données, ce qui permettra à un hacker d'afficher notre base, d'accéder à notre site, etc.

Comment s'en protéger

Les injections SQL bien utilisées peuvent être redoutables, mais heureusement pour nous, il est très simple de s'en protéger.

La solution consiste à échapper les caractères spéciaux contenus dans les chaînes de caractères entrées par l'utilisateur. En PHP on peut utiliser pour cela la fonction `mysqli_real_escape_string`, qui transformera la chaîne `' --` en `\' --`

Une autre précaution est d'utiliser la fonction `ctype_digit` pour vérifier les variables numériques des requêtes. On peut aussi forcer la transformation de la variable en nombre en la faisant précédé d'un *transtypeur*, comme `(int)` si on attend un entier (la chaîne `0 or 1 --` sera alors transformée en l'entier `0` et l'injection SQL échouera).

Une autre solution consiste à utiliser des requêtes préparées : dans ce cas, une compilation de la requête est réalisée avant d'y insérer les paramètres et de l'exécuter, ce qui empêche un éventuel code inséré dans les paramètres d'être interprété.

3.2.4.1.4- L'attaque par force brute

Cette partie est un peu différente des autres dans la mesure où nous apprendrons à nous protéger contre les attaques par force brute (plus couramment appelées "bruteforce"). Le but n'est donc pas de se protéger contre une faille, mais de rendre ce type d'attaque inefficace.

L'attaque par force brute est une technique relativement courante, mais qui nécessite beaucoup de temps d'exécution. Comme son nom l'indique, c'est une technique très... barbare ! Le principe est simple : le hacker crée un formulaire identique à celui de votre site, puis avec un script il essaye toutes les combinaisons de mots de passe

possibles et imaginables. Il existe une autre technique de bruteforce où le hacker se base sur d'énormes dictionnaires contenant des millions voire milliards de mots de passe régulièrement utilisés. Pas très fin, mais redoutablement efficace ! Surtout si l'utilisateur s'est inscrit avec un mot de passe simple.

Comment s'en protéger

- **Le bannissement d'IP**

Cette technique est plus complexe, mais beaucoup plus efficace. Elle consiste à limiter le nombre de tentatives par personne et par jour. Bien évidemment, il ne faut pas bloquer le compte visé par l'attaque, mais empêcher le hacker de continuer à brutaliser notre formulaire. L'astuce consiste donc à créer une table qu'on appellera connexion et dans laquelle on enregistrera toutes les tentatives ratées de connexions au site. On y enregistrera simplement l'IP de la personne. Au-delà d'un certain nombre de tentatives, l'accès au compte avec cet IP devient impossible pendant un certain temps.

- **Ralentir le Processus**

Cette technique ne rend pas impossible l'attaque par force brute, mais elle va considérablement la ralentir. Tellement la ralentir qu'il faudrait des milliers d'années avant de trouver un mot de passe basique. Elle a également l'avantage d'être très facile à mettre en place. Le principe est simple. Comme indiqué dans le titre, il suffit de "ralentir" l'exécution du formulaire de connexion en ajoutant une pause de 1 seconde. Un utilisateur normal ne verra presque pas la différence, mais le pirate sera drôlement embêté.

- **La vérification par captcha**

Cette dernière technique est très simple. Elle consiste à insérer un captcha de vérification dans vos formulaires. Et c'est très efficace ! C'est presque imparable pour être certain qu'on a à faire à un humain et non à un script.

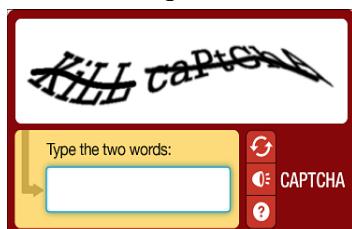


Figure 3.7 : Captcha de Vérification

Il existe plusieurs autres failles comme la faille Upload, la CSRF, La CRLF , les variables de session, mais chacune de ses failles fort heureusement peuvent être sécurisé de plusieurs manière.

Nous avons aussi vu qu'il y'a plusieurs moyens de sécuriser l'accès à notre site, ce ne sont pas les seuls nous pouvons aussi par exemple déconnecter notre utilisateur après quelques minutes d'inactivité...les possibilités de protection sont diverses et multiples.

3.2.4.2-Comment assurer la confidentialité des informations du système ?

A l'ère du numérique, on assiste aujourd'hui à une accélération et à une amplification des données disponibles. Dossiers électroniques des patients, consultations à distance, échanges entre professionnels dans le monde entier pour faire avancer la recherche, bilans médicaux, comptes en ligne d'assurance maladie, intranet des laboratoires pharmaceutiques... Le monde de la santé regorge d'informations confidentielles qui doivent également être protégées des regards indiscrets ; les données médicales font partie depuis de nombreuses années du quotidien des professionnels de santé, et leur protection, avec les règles de déontologie et de respect de la vie privée s'y afférent, a toujours été d'une grande importance.

C'est en sens que le système doit au maximum rendre confidentielle ou garder de manière confidentielle les données du patient. En ce sens plusieurs méthodes peuvent être utilisé :

3.2.4.2.1-Le hashage

C'est en fait un algorithme qui va, à partir d'une donnée de départ, proposer une empreinte irréversible. Une fois une donnée hashée, il est impossible de la "décrypter", et c'est là la plus grande différence avec un cryptage.

C'est-à-dire qu'il n'existe pas de fonction permettant à coup sûr de retrouver l'information initiale partant de la valeur hachée.

Il existe beaucoup d'algorithmes de hashage comme le SHA-0, le SHA-1, le md5, SHA-512, SHA-256, Whirlpool, ... il est même possible de créer le sien ce qui est plus

intéressant car le pirate ou hacker qui tombera devant ça ne pourra pas savoir comment ce hash a été produit.

Dans une fonction de hachage, les caractères majuscules ou minuscules donnent des hash différents par exemple avec le hash md5 le caractère a sera différent de A, ces derniers donneront :

a : 0cc175b9c0f1b6a831c399e269772661

A : 7fc56270e7a70fa81a5935b72eacbe29

Le mot sénégal avec s minuscule donnera 56d5b89e45b78463599d26c5e72e9351 tandis que Sénégal avec S majuscule lui sera b4906e864ced101f095828e6cebbf1cd.

Même s'il existe tout de même des techniques utilisées par les hackers pour "casser" les hash. La plus connue étant l'attaque par la force brute (souvent associée à des dictionnaires de mots appelées rainbow table), qui consiste à hasher tous les mots possibles et imaginables et à les comparer à la chaîne hashée juste qu'à trouver le mot correspondant (même principe que l'attaque par force brute évoquée précédemment). C'est pour cette raison qu'à ce jour, les algorithmes SHA-1 et md5 ne sont plus considérés comme sûrs ! C'est pour cette raison qu'on préférera donc des algorithmes de hash plus complexes comme par exemple le **SHA-256, SHA-512**, avec un temps d'exécution beaucoup plus élevé. Ils produiront un hash plus "lourd" qui sera maintes fois plus sécurisé qu'un simple SHA-1.

Dans notre système le hashage peut être utilisé par exemple lors de l'inscription d'un médecin pour la création de son mot de passe, on peut utiliser une fonction de hashage qui va hasher le mot de passe et le sauvegardera dans la base de données, ce qui permettra une confidentialité du mot de passe qui sera inconnu même de l'administrateur de la base de données car hasher. Lors de la connexion de cet utilisateur à notre site, son mot de passe sera hasher puis comparé à celle de la base si celle-ci correspond il pourra accéder au site. Pour mieux sécuriser notre mot de passe nous pouvons aussi ajouter des caractères à notre mot de passe (on appelle aussi se procéder **sel**) lors de sa création avant de la hasher, pour la rendre plus complexe. Lors de sa connexion aussi ses mêmes caractères seront ajoutés à notre mot de passe puis hasher et comparés à celle de la base pour ensuite pouvoir autoriser la connexion au site.

Même si le pirate dispose de notre hash, il ne peut accéder à notre page sauf s'il dispose du code PHP de notre page ce qui est quasi impossible ou du mot de passe réel.

Comme le hashage, le cryptage aussi est une manière de sécuriser notre site.

3.2.4.2.2-Le cryptage

Le cryptage est une notion très vaste. Il existe des milliers de possibilités pour crypter et décrypter des données. Je vais donc vous expliquer rapidement le principe du chiffrage et vous présenter une méthode. Contrairement au hashage, le cryptage est réversible. Il est donc possible de récupérer ce qui a été crypté, si on connaît l'algorithme utilisé ainsi que la clé de chiffrement. Une clé de chiffrement est une suite de caractères sur laquelle l'algorithme va se baser pour chiffrer les données. C'est un peu comme un mot de passe qui permet le déchiffrage des données.

En PHP une des fonctions comme **`mcrypt_encrypt()`** permettent de crypter ou décrypter nos données.

3.2.4.3-Comment est-il possible de partager les données d'un site à l'autre et assurer la sécurité du partage de ses informations ?

Partager des informations d'un site à un autre fait appel au réseau et à ses protocoles. Pour partager des gros fichiers comme des données d'imagerie par exemple les radios médicales, nous pouvons utiliser le protocole FTP qui correspond à un standard Internet permettant de transférer de gros volumes. Ce protocole est capable aujourd'hui d'accepter une sécurité du type SSL ou TLS (c'est ce dernier qui est utilisé dans le protocole https).

Pour faire transiter les données entre les sites nous pouvons par exemple utiliser des outils comme :

Le VPN

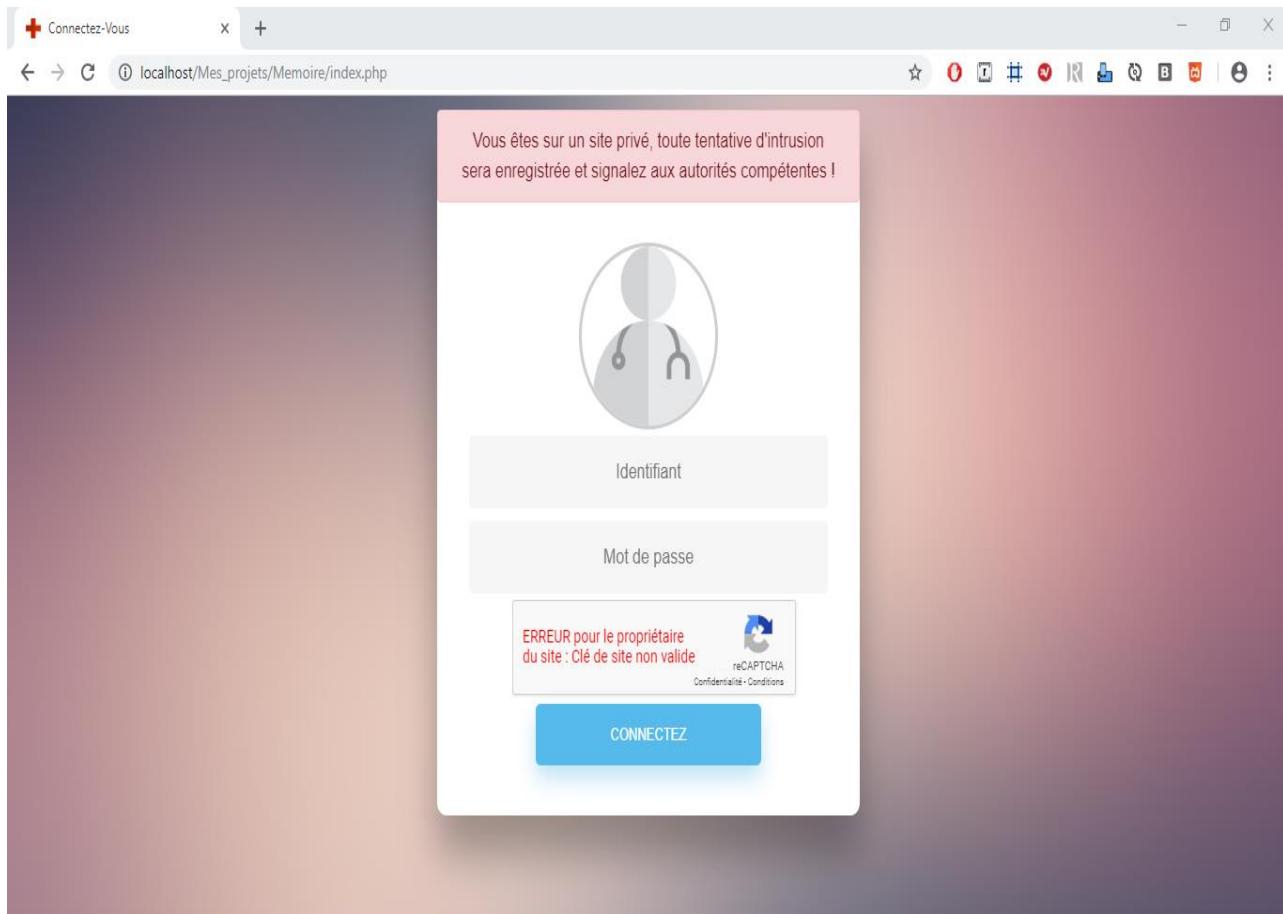
Le VPN où virtual private network est un système permettant de créer un lien direct entre des ordinateurs distants, en isolant ce trafic. On utilise notamment ce terme dans le travail à distance. Un VPN permet d'accéder à des ordinateurs distants comme si l'on était connecté au réseau local. Il permet d'avoir un accès au réseau interne (réseau d'entreprises, par exemple) ou de créer un réseau de pairs. Les VPN sont utilisés pour

protéger un trafic web privé contre les interférences, l'espionnage. C'est en ce sens que le VPN sera très important pour le transit de nos données entre sites de manière plus sécurisés nous pouvons aussi coupler ce dernier avec la notion de cryptages vu précédemment, les données seront cryptées à l'entrer et décrypter à la sortie.

Le partage de l'information de manière sécurisé sera très important pour notre site car ce dernier permettra à notre site de faire des recherches sur un patient grâce à une page de notre site qui est destiné à cette recherche, pour connaître ses antécédents ou certaines informations importantes pour une meilleure prise en charge de ce dernier, sur d'autres sites qui possèdent déjà des informations sur ce dernier qui va les retourner à notre site .Cette notion de partage des informations du patient, aura pour référence le DMP(Dossier Médical Partager) qui est un projet mis en place dans certains pays comme par exemple en France. Pourquoi ne pas adopter ce système au Sénégal ?

3.2.5-Présentation de l'application

3.2.5.1-La page de Connection



3.2.5.2-La page d'accueil

Et quoniam inedia gravi adflictabantur, locum petivere Paleas nomine, vergentem in mare, valido muro firmatum, ubi conduntur nunc usque commeatus distribui militibus omne latus Isauriae defendantibus adsueti. circumstetere igitur hoc munimentum per triduum et trinoctium et cum neque adclivitas ipsa sine discrimine adiri letali, nec cuniculis quicquam geri posset, nec procederet ullum obsidionale commentum, maesti excedunt postrema vi subigente maiora viribus adgressuri. Martinus agens illas provincias pro praefectis aerumnas innocentium graviter gemens saepeque obsecrans, ut ab omni culpa immunibus parceretur, cum non impetraret, minabatur se discessurum: ut saltem id metuens perquisitor malivolus tandem desineret quieti coalitos homines in aperta pericula projectare. Nihil morati post haec militares avidi saepe turbarum adorti sunt Montium primum, qui divertebat in proximo, levi corpore senem atque morbosum, et hirsutis resticulis cruribus eius innexis divaricatum sine spiramento ullo ad usque praetorium traxere praefecti.

Et quoniam inedia gravi adflictabantur, locum petivere Paleas nomine, vergentem in mare, valido muro firmatum, ubi conduntur nunc usque commeatus distribui militibus omne latus Isauriae defendantibus adsueti. circumstetere igitur hoc munimentum per triduum et trinoctium et cum neque adclivitas ipsa sine discrimine adiri letali, nec cuniculis quicquam geri posset, nec procederet ullum obsidionale commentum, maesti excedunt postrema vi subigente maiora viribus adgressuri. Martinus agens illas provincias pro praefectis aerumnas innocentium graviter gemens saepeque obsecrans, ut ab omni culpa immunibus parceretur, cum non impetraret, minabatur se discessurum: ut saltem id metuens perquisitor malivolus tandem desineret quieti coalitos homines in aperta pericula projectare. Nihil morati post haec militares avidi saepe turbarum adorti sunt Montium primum, qui divertebat in proximo, levi corpore senem atque morbosum, et hirsutis resticulis cruribus eius innexis divaricatum sine spiramento ullo ad usque praetorium traxere praefecti.

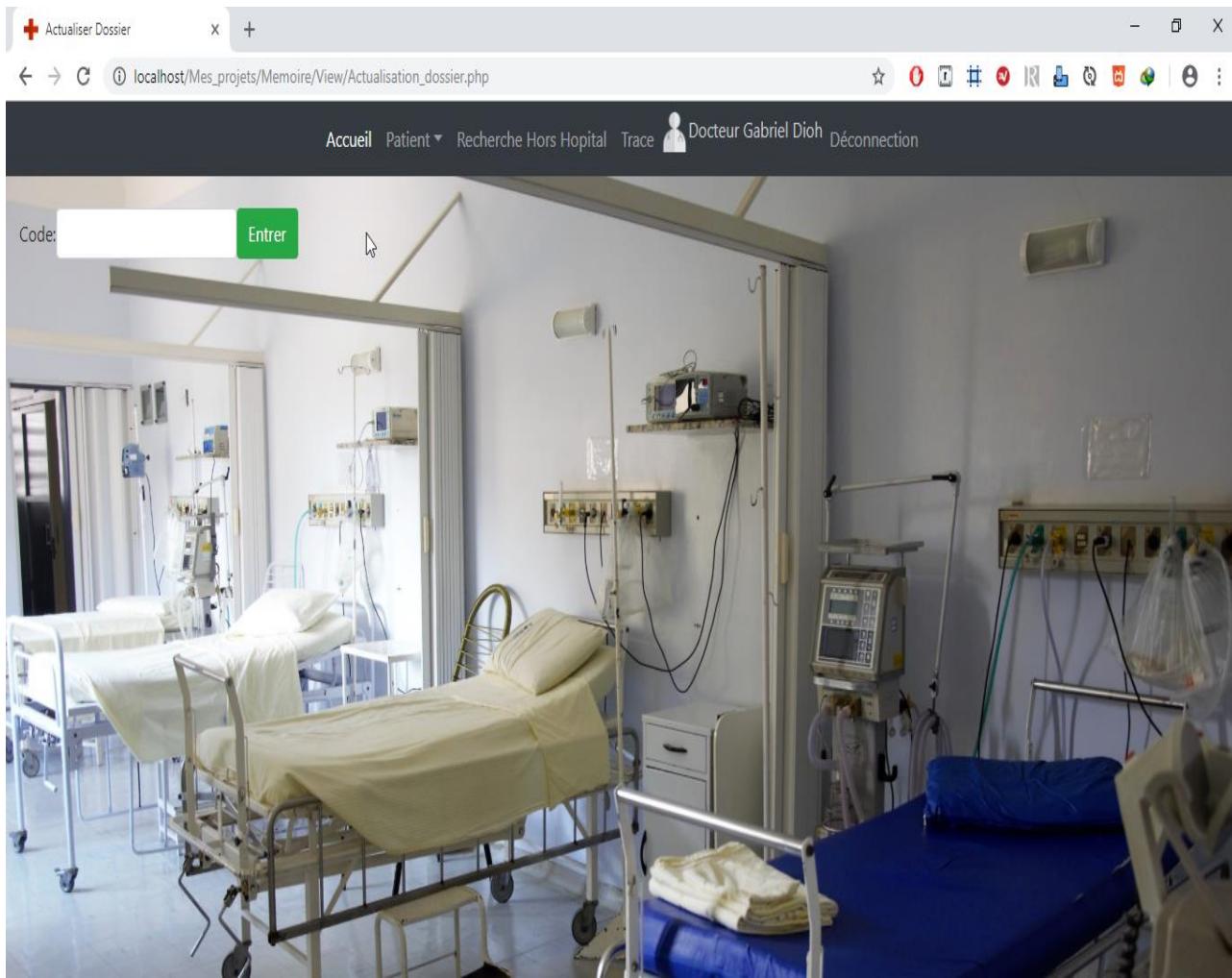
La page d'accueil est un élément personnalisable selon l'hôpital elle peut contenir les informations de l'hôpital comme le nom de l'hôpital et l'histoire de ce dernier, ou les nouvelles se trouvant dans l'hôpital c'est-à-dire son actualité.

3.2.5.3-Patient

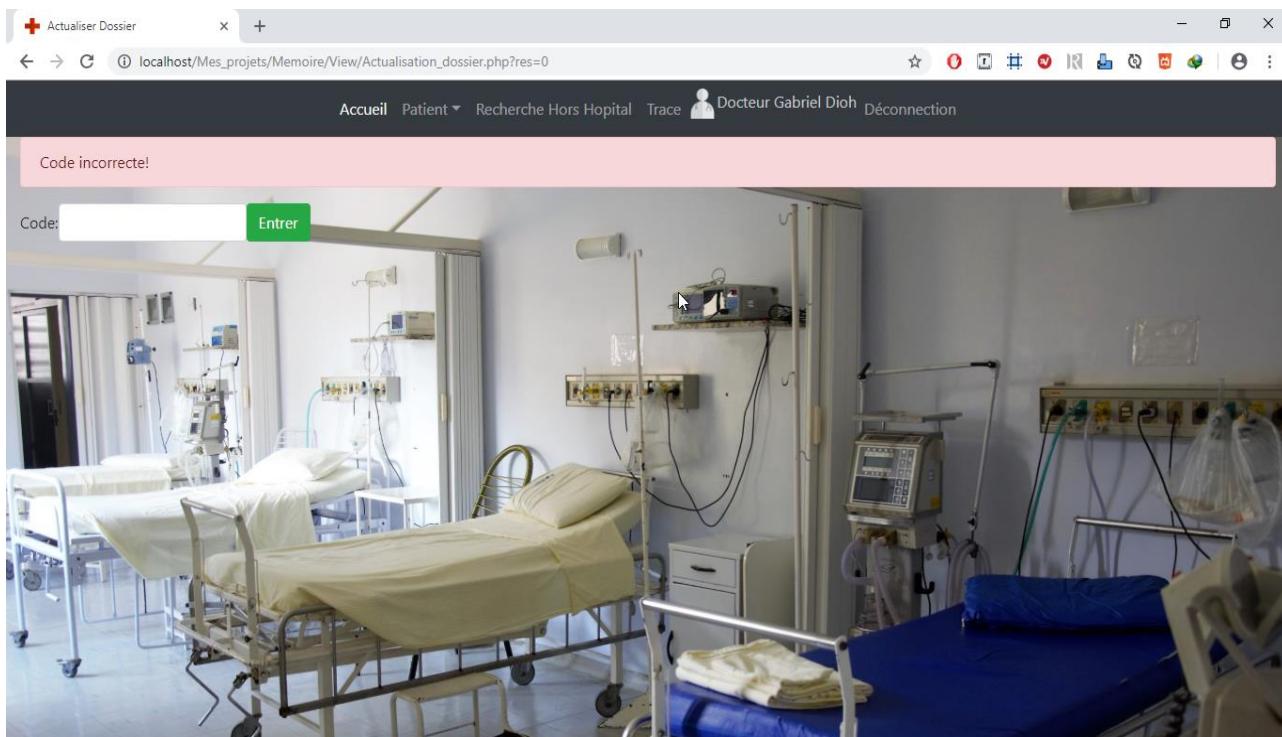
Il contient des éléments comme :

- La page d'ajout des patients
- La page Actualisation du dossier
- Consultation des informations patient
- Modification des informations patient
- Archivage du patient

Toutes ses pages mises à part la page d'ajout des patients seront précédées, d'une page où il faudra mettre un code que le médecin devra renseigner (qui est différent du mot de passe et donné par l'organe qui sera responsable de cette gestion par exemple le ministère de la santé).



Si le code est correct nous aurons les pages correspondantes sinon un message d'erreur (voir ci-dessous).



3.2.5.3.1-La page d'ajout des patients

Si l'identifiant, le numéro de carte d'identité ou de passeport avait déjà été attribué à un patient nous aurions un message d'erreur lorsque ce nouveau voudra être enregistré mais aussi ce patient ne sera pas sauvegardé dans notre base ainsi que tout autre erreur d'enregistrement.

The screenshot displays two identical web-based patient registration forms, one above the other, against a dark background. Both forms are titled "Ajouter un patient".

Top Form (Left Column):

- ADMINISTRATIF:**
 - Identifiant Patient*: [Input field]
 - Numéro Carte d'identité: [Input field]
 - Numéro Passeport: [Input field]
 - Nom*: [Input field]
 - Prenom*: [Input field]
- MOTIF D'HOSPITALISATION:** [Large input field]
- Hospitalisé par:** [Input field]

MIDDLE SECTION (Between columns):

EXAMEN CLINIQUE:

T^o: [Input field] TA: [Input field] T: [Input field] P: [Input field]

Autres [Input field]

Bottom Form (Left Column):

- Prenom*: [Input field]
- Date de Naissance*: [Input field] (Format: jj/mm/aaaa)
- Lieu de Naissance*: [Input field]
- Sexe*: [Select dropdown] (Value: M)
- Adresse: [Input field]
- Telephone: [Input field]
- Date d'entrée*: [Input field] (Format: jj/mm/aaaa --:--)

MIDDLE SECTION (Between columns):

EXAMENS PARACLINIQUE OU COMPLÉMENTAIRE:

- Ajout Examen Biologique: [File input field] Choisir un fichier Aucun fichier choisi
- Ajout Radio: [File input field] Choisir un fichier Aucun fichier choisi

Autres [Input field]

Options:

Partage [checkbox] Personne Sensible [checkbox]

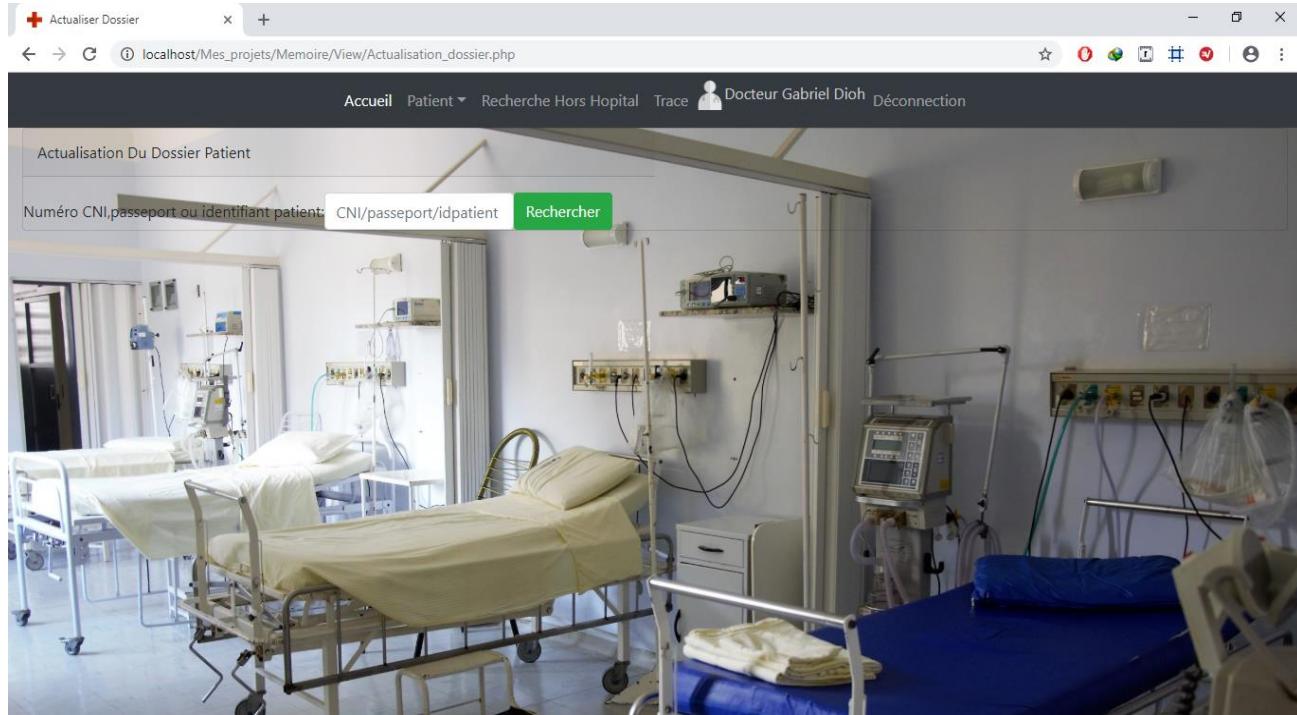
Buttons:

Enregistrer (Green button) Annuler (Red button)

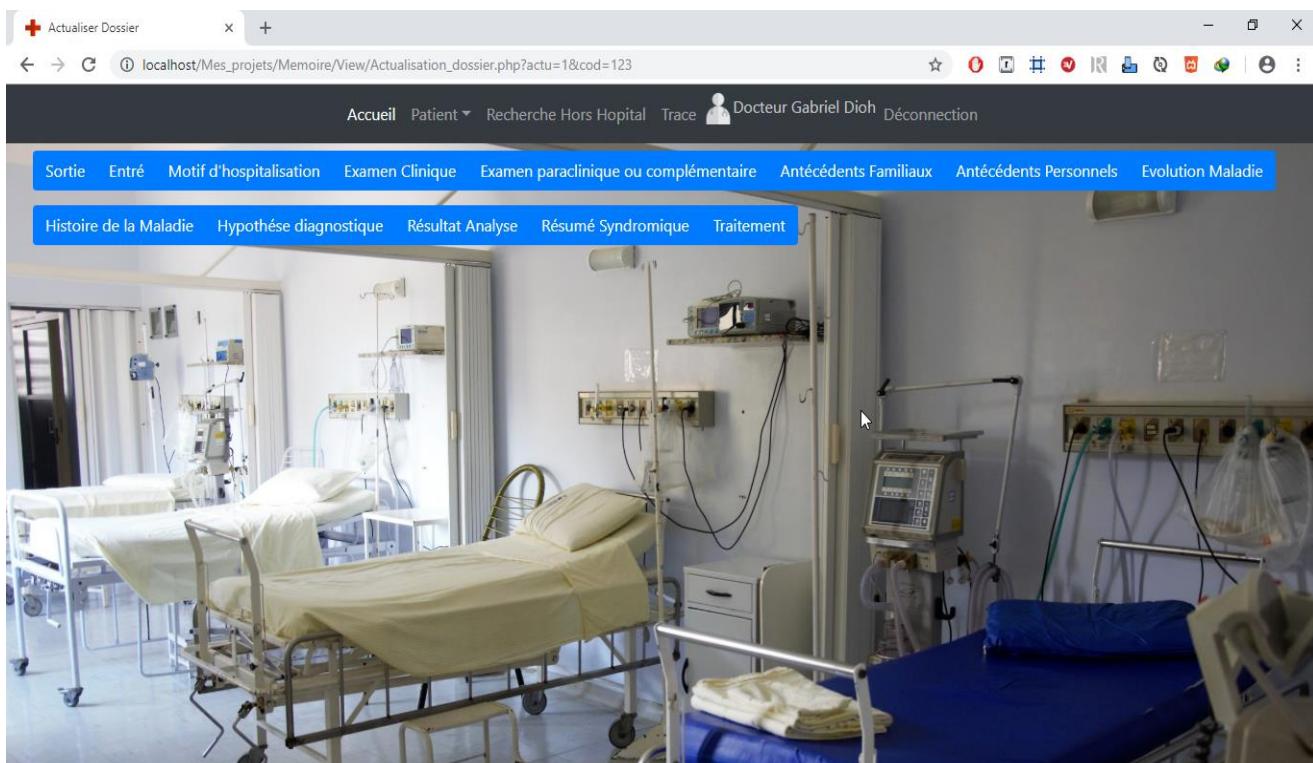
3.2.5.3.2-La page Actualisation du dossier

Elle se présente comme suit :

Après avoir mis le bon code nous aurons cette page.



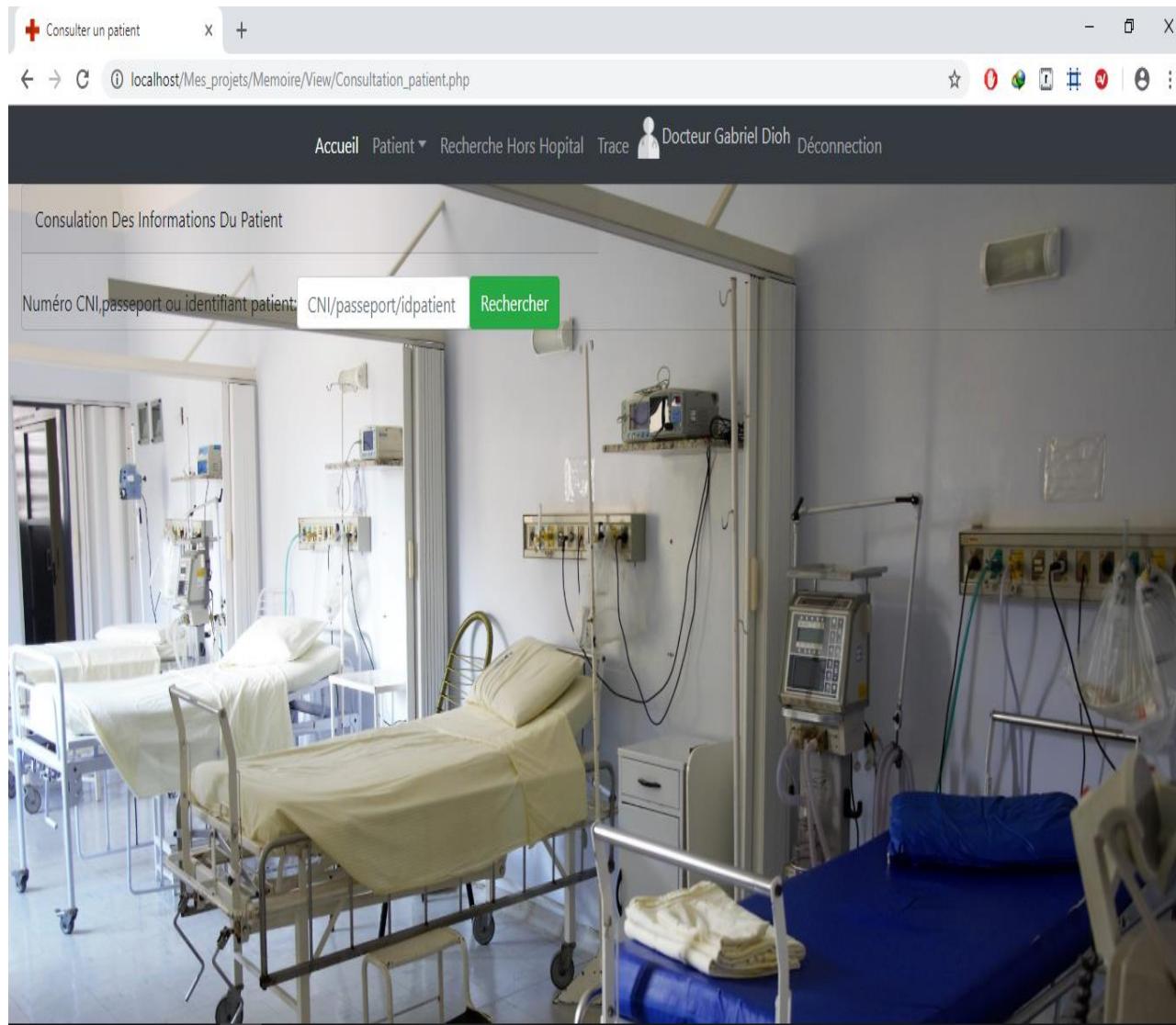
Il faudra mettre le numéro de Carte d'Identité ou passeport ou identifiant pour pouvoir actualiser ou mettre à jour le dossier d'un patient. Si ses informations ne sont pas correctes ou n'existent pas nous aurons un message d'erreur, sinon une autre page qui nous permettra d'actualiser le dossier du patient en choisissant les éléments à mettre à jour.



3.2.5.3.3-Consultation des informations patient

Elle se présente comme suit :

Après avoir mis le bon code nous aurons cette page.



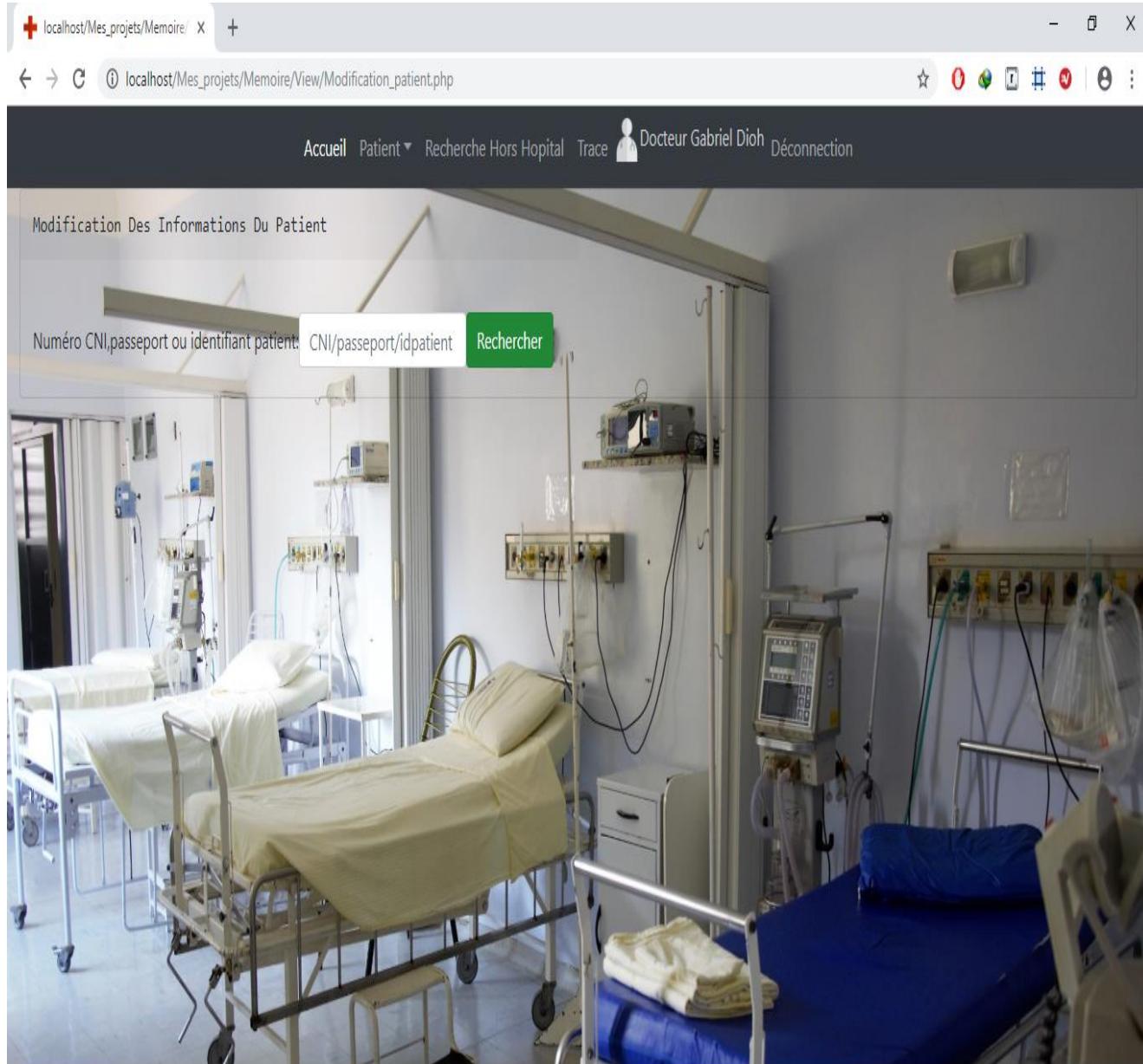
Il faudra mettre le numéro de Carte d'Identité ou passeport ou identifiant pour pouvoir consulter le dossier d'un patient.

Si ses informations ne sont pas correctes ou n'existent pas nous aurons un message d'erreur, sinon une autre page qui nous permettra de consulter le dossier du patient mais aussi de l'imprimer, ou mettre sous un format Word ou PDF.

3.2.5.3.4- Modification des informations patient

Elle se présente comme suit :

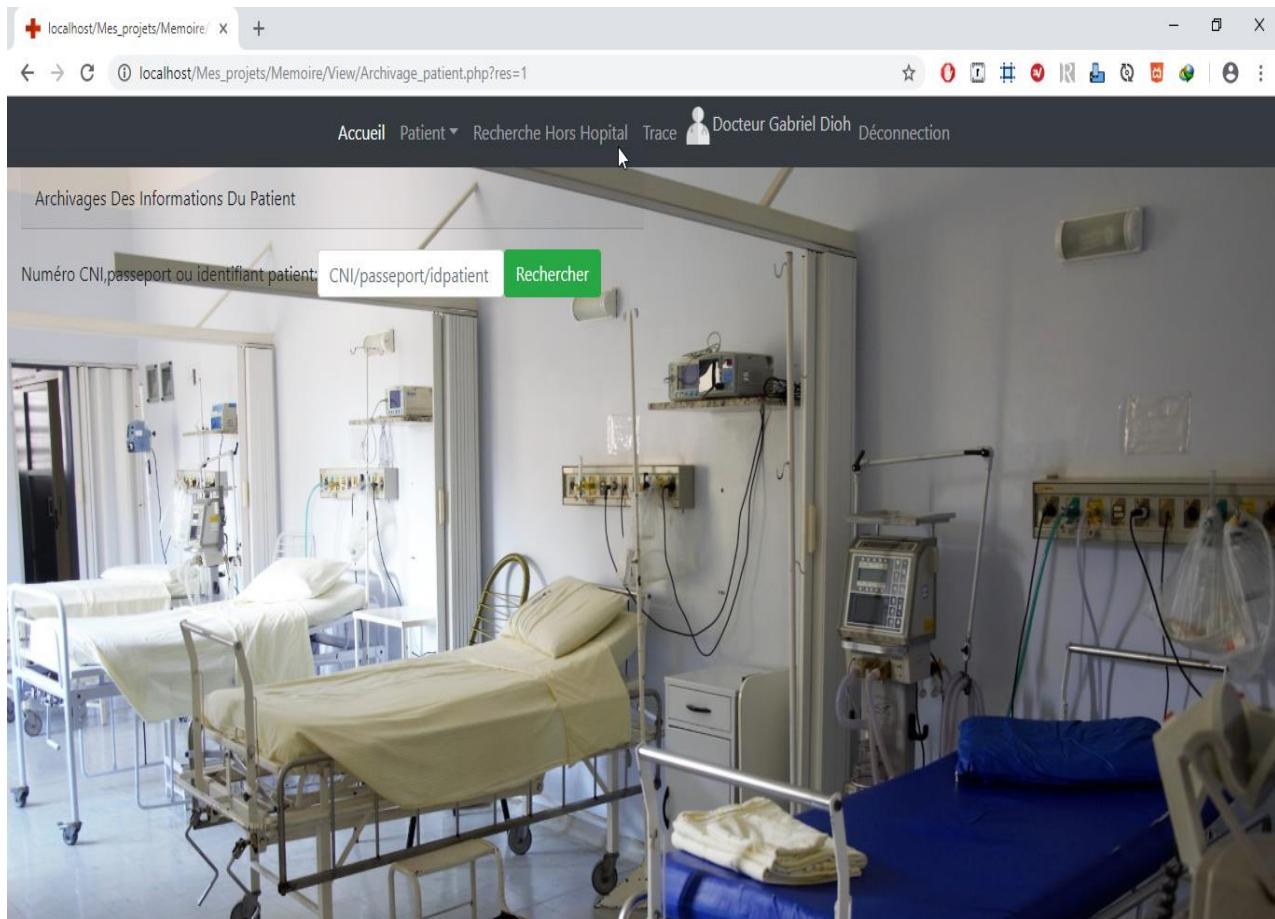
Après avoir mis le bon code nous aurons cette page.



Il faudra mettre le numéro de Carte d'Identité ou passeport ou identifiant pour pouvoir modifier les informations du dossier d'un patient. Si ses informations ne sont pas correctes ou n'existent pas nous aurons un message d'erreur, sinon une autre page qui nous permettra de modifier les informations du patient.

3.2.5.3.5-Archivage du patient

Elle se présente comme suit :



Il faudra mettre le numéro de Carte d'Identité ou passeport ou identifiant pour pouvoir archiver ce dossier patient. Si ses informations ne sont pas correctes ou n'existent pas nous aurons un message d'erreur, sinon une autre page qui nous permettra d'archiver le patient. Un patient archivé ne pourra être modifié mais sera consultable, cet archivage correspondra au décès du patient.

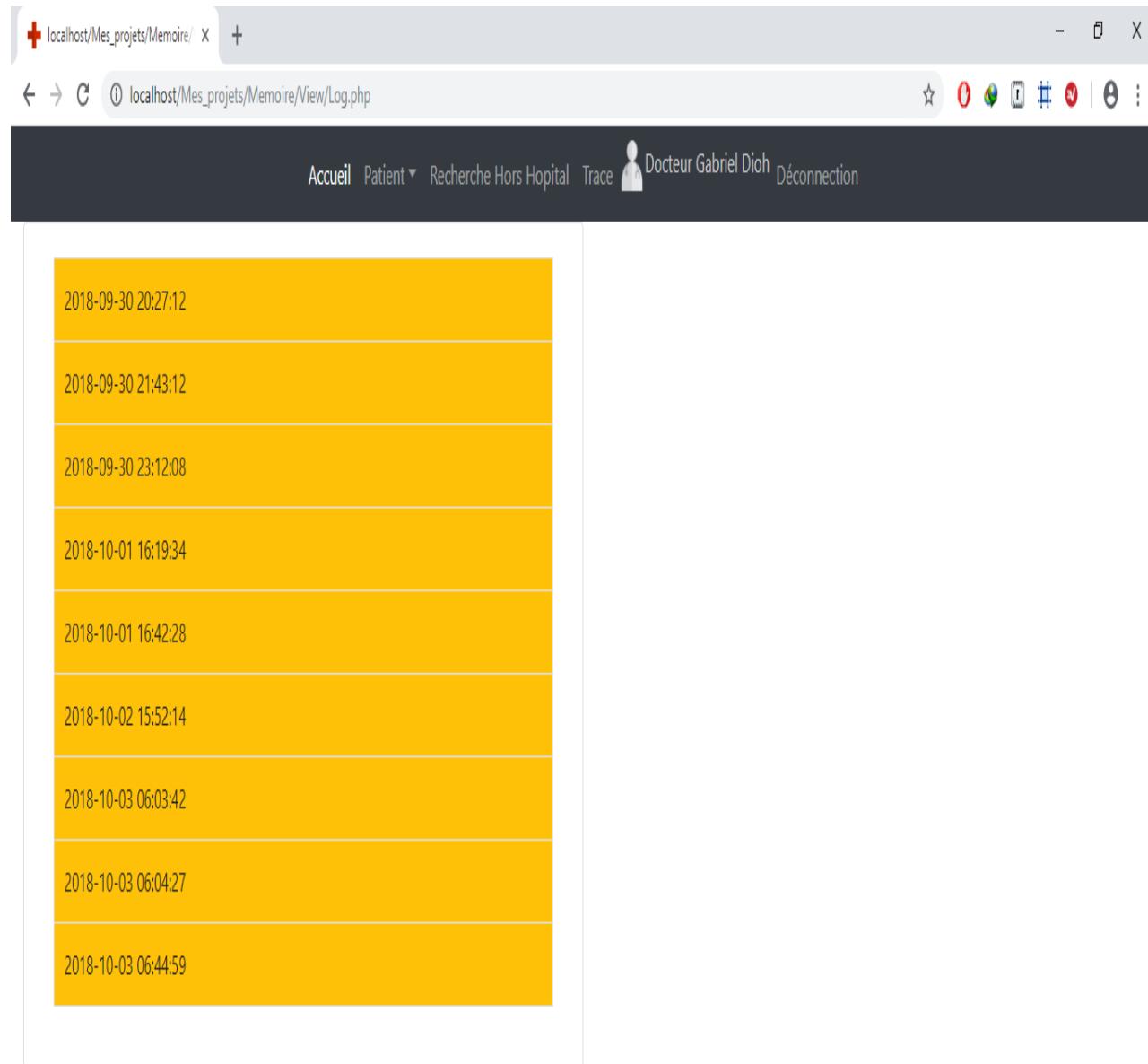
3.2.5.4-Recherche hors hôpital

Elle se présente comme suit :

Lorsque le code est donné, si le code correspond la page disparait pour amener un autre (voir ci-dessous) ou il faudra mettre le numéro de CI ou passeport ou identifiant pour pouvoir consulter certaines informations d'un patient d'un autre hôpital qui accepte le partage dossier. On pourra se déconnecter pour revenir à la page d'origine.

3.2.5.5-Trace

Présenter comme suit elle représente la date et l'heure des dernières connexions réaliser par le médecin.



The screenshot shows a web browser window with the URL `localhost/Mes_projets/Memoire/View/Log.php`. The page displays a list of connection times in a table format:

Date et Heure
2018-09-30 20:27:12
2018-09-30 21:43:12
2018-09-30 23:12:08
2018-10-01 16:19:34
2018-10-01 16:42:28
2018-10-02 15:52:14
2018-10-03 06:03:42
2018-10-03 06:04:27
2018-10-03 06:44:59

Conclusion

En somme l'étude réalisée dans ce mémoire nous a permis de dégager des solutions face au problème rencontré par les acteurs de soin mais aussi de créer une application web pour la gestion des dossiers patients, offrant un ensemble de services tels que l'ajout des patients, l'actualisation du dossier, la consultation, la modification, la Suppression des informations patient ; mais surtout le partage des informations entre les différents hôpitaux.

Plusieurs technologies ont été nécessaires pour la réalisation de notre projet, on citera donc le langage HTML pour la réalisation des pages statiques, le langage PHP pour la partie dynamique, SQL pour l'élaboration des requêtes d'interrogation de la base de données.

Après le passage par les différentes étapes de développement, l'application a abouti à un logiciel fonctionnel qui répond globalement aux critères imposés dans ce domaine.

Le présent travail nous a permis de conforter nos connaissances en conception logicielle mais aussi de nous ouvrir les yeux vers de plus grandes perspectives comme d'implémenter et d'adapter le système aux autres établissements de santé (comme les cabinets, les cliniques...) pour un meilleur suivi du patient et de ses traitements.

Références

Webographie

<https://www.agenda-medical.fr/therapeute-therapie-medicale/informatique-medicale/>

<https://fr.wikipedia.org/>

<https://www.opale.io/definition-dematerialisation/>

<https://www.xp-internet.com/xp-infos/securite-sites-internet.php>

<https://www.gralon.net/articles/sante-et-beaute/medecine/article-le-dossier-medical-partage-en-5-questions-10824.htm>

<https://www.google.fr/>

<https://developpez.com/>

https://www.memoireonline.com/01/13/6844/m_Conception-et-realisation-dun-site-web-pour-le-departement-dinformatique32.html

Thèse ou mémoire

➤ NDENE PH

Année de publication 2003-2004

Mise en Œuvre d'un logiciel de gestion des patients

Mémoire de Fin de Cycle pour l'obtention du Diplôme d'ingénieur Technologue (D.I.T)

Institut Supérieur de technologie industrielle (I.S.T.I)

149 pages

➤ NDIAYE CP

Année de publication 2003-2004

Gestion du cabinet dentaire Elhadji Malick

Mémoire de fin de cycle pour l'obtention du diplôme de technicien supérieur (D.T.S)

Institut Supérieur de technologie industrielle (I.S.T.I)

64 pages

Résumé en français +mots clés

L'hôpital a pour vocation d'apporter aux malades les soins appropriés aux patients. Pour assurer une qualité de ses soins ce dernier doit passer par une bonne gestion des dossiers du patient, mais au Sénégal cela est très difficile faute de moyens, les structures sanitaires se heurtent à des problèmes de manque d'espace, de désorganisation, de pertes des données médicales, d'une mauvaise alimentation du dossier, mais aussi d'accès au dossier patient externe au service... C'est dans ce sens qu'un système pour gérer ce dernier est important d'où le choix de notre sujet sur l'**Etude et mise en place d'un système de gestion des dossiers et Informations des patients dans les hôpitaux du Sénégal** qui sera implanté sous forme d'application web il va gérer le dossier patient mais aussi permettra de partager des informations (comme le DMP) dans les autres hôpitaux ayant le même système.

La réalisation de ce mémoire est faite selon un plan d'abord nous parlons du Cadre théorique et de l'étude préalable pour ensuite passer par l'étude conceptuelle et terminer par l'étude détaillée et l'implémentation.

Mots clés

- Dossier patient
- Hôpital Sénégal
- Gestion dossier patient
- Mémoire DTS informatique
- Site web patient

Résumé en anglais +mots clés

The purpose of the hospital is to provide patients with appropriate care. To ensure a quality of his care the latter must go through a good management of the patient's records, but in Senegal this is very difficult for lack of means, the sanitary structures are faced with problems of lack of space, disorganization, losses Medical data, poor nutrition of the file, but also access to the patient file external to the service... It is in this sense that a system to manage the latter is important hence the choice of our subject on the study and implementation of a system of management of patient records and information in hospitals of Senegal which will be implemented as a Web application. It will manage the patient file but will also allow to share information (like the DMP) in other hospitals having the same system.

The realization of this memory is made according to a plan first we talk about the theoretical framework and the preliminary study and then go through the conceptual study and end with the detailed study and implementation.

Mots clés

- Patient Record
- Hospital Senegal
- Patient File Management
- Computer DTS Memory
- Patient website

Annexes

*Service
Hôpital - Clinique
Annexe -*

Plan de l'observation médicale

- Date, identification de la personne ayant rédigé l'observation.
- Mode d'arrivée à l'hôpital, adressé par qui, coordonnées du patient et du médecin traitant.
- **Motif d'hospitalisation** (un ou plusieurs symptômes).
- **Antécédents**
 - Personnels :
 - Chirurgicaux.
 - Médicaux.
 - Allergiques.
 - Gynéco-obstétricaux (sans oublier la date des dernières règles !).
 - Familiaux (premier degré).
 - Traitements habituels ou occasionnels.
- **Mode de vie**
 - Contexte socioprofessionnel.
 - Contexte familial, marital, vie affective.
 - Eventuelles intoxications.
 - Voyages, animaux de compagnie.
- **Histoire de la maladie**
 - Localisation des troubles.
 - Début :
 - Quand (date, heure) ?
 - Occupation lors de la survenue des troubles.
 - Mode d'installation, facteur déclenchant.
 - Rythme, durée.
 - Qualité des troubles.
 - Intensité, sévérité, retentissement sur le quotidien.
 - Facteurs qui aggravent et soulagent.
 - Manifestations associées.
- **Examen clinique**
 - Poids et signes vitaux.
 - Apparence générale.
 - Recherche des signes physiques (inspection, palpation, percussion, auscultation) :
 - Cardio-vasculaire.
 - Pleuro-pulmonaire.
 - Abdomen.
 - Neurologique.
 - Locomoteur.
 - Uro-génital.
 - Seins.
 - Aires ganglionnaires.
 - Cutanéo-muqueux.
 - Tête et cou.
 - Nez, bouche, gorge, oreilles, yeux
 - Thyroïde
 - Examen pelvien
- **Conclusion**
 - Synthétiser le problème en une courte phrase.
 - Liste des hypothèses diagnostiques par ordre de vraisemblance.
 - Ebaucher une stratégie de prise en charge.
- **Examens complémentaires ou para-cliniques**
 - analyse de sang
 - Radio - Scanner - Echo .

Contenu d'un dossier Patient Complet (Recto)

- Résultats des analyses
- Diagnostic retenu :
ex : pneumopathie, diabète, gale etc.
- Traitement
 - médical
 - chirurgical
- Évolution :
bonne sa mauvaise etc.

Contenu d'un dossier Patient Complet (Verso)

Hôpital

Adresse

***Motif d'hospitalisation**

Mr x né le ... ou âgé de ... hospitalisé dans le service de ... pour motif (toux, fièvre, amaigrissement)

Adresse de la personne-n° de Téléphone

Nom et Titre du médecin traitant

***Antécédents**

- Personnel

-Médicaux : Différentes maladies avec éventuelle hospitalisation.

-Chirurgicaux : Différentes interventions chirurgicaux.

- Familiaux

-Pathologie Familiale : diabète, hypertension artérielle, etc...

-Cause décès des parents, âge décès, etc...

***Histoire de la maladie(interrogatoire)**

Raconter le début de la maladie avec les différentes signes et éventuels traitements reçu

Amélioration ou pas, bilan médicaux ou pas, durée du bilan, facteurs aggravants, etc...

***Examens Cliniques**

Examen clinique de tous les appareils (Tous les membres du malade)

Examen de l'appareil malade (le membre qui est malade)

Prise du poids, de la température, de la fréquence, de la tension artérielle,taille

***Examens Paraclinique**

-Analyse de sang

-Radio et autres

***Hypothèse diagnostic** : évoquer les différentes hypothèses

***Résumé syndromique** : exemple Syndrome d'hypertension, syndrome d'hyperglycémie

Donner tous les syndromes observer.

***Résultat des analyses**

***Diagnostics retenus** : pneumopathie, palu, diabète, etc...

***Traitement**

-Médical

-Chirurgical

Evolution : Bonne ou mauvaise...