

République Algérienne Démocratique et Populaire
Université Abou Bakr Belkaid– Tlemcen
Faculté des Sciences
Département d'Informatique

Mémoire de fin d'études
Pour l'obtention du diplôme de Master en Informatique
Option : Génie Logiciel (G.L)

Thème

Réalisation d'une plateforme de gestion des patients dialysés et conception et exploitation d'un entrepôt de données (SMARTDIALYSE)

Réalisé par :

- LACHACHI Mohammed Charaf Eddine
- GHERNAOUT Malik

Présenté le 20 juin 2017 devant le jury composé de MM :

- | | |
|----------------------------|-----------------|
| - Mohamed MESSABIHI | (Président) |
| - Yassamine SELADJI | (Encadrant) |
| - Mohammed TADLAOUI | (Examineur) |
| - Amal HALFAOUI | (Co-encadrante) |
| - Mahfoud CHERIF BENMOUSSA | (Co-encadrant) |

Année universitaire : 2016-2017

Mohammed Charaf Eddine LACHACHI

Dédicace

Je dédie ce travail à la mémoire de ma très chère sœur qui m'a soutenue tout au long de ma vie. Que dieu la garde dans son vaste paradis.

Remerciements

Tout d'abord je tiens à remercier le bon dieu pour m'avoir donné le courage, la force et la volonté pour réussir et de m'avoir éclairci le chemin tout au long de ma vie.

Je tiens à remercier mon encadrante Madame SELADJI pour son aide, sa disponibilité, sa confiance et ses conseils judicieux qui ont contribué à la finalisation de ce travail.

Je tiens à remercier le membre du jury qui me font l'honneur d'évaluer ce travail.

Ce travail n'aurait pas été possibles sans l'aide d'un néphrologue, c'est pour ça que je tiens à remercier Monsieur CHERIF BENMOUSSA pour les détails qu'il nous a fournis, sa collaboration et sa disponibilité.

Je tiens à exprimer toute ma reconnaissance à Madame HALFAOUI pour ses conseils, sa disponibilité et sa gentillesse, sans elle le fonctionnement de l'entrepôt de données n'aurait pas abouti.

Je remercie aussi mes parents et mes frères qui ont tout fait et donné pour ma réussite.

Enfin, je ne peux clore cette page sans remercier mon cher binôme pour sa sincère amitié et confiance

Mohammed Charaf Eddine LACHACHI

Dédicace

Je dédie mon travail à mes très chers parents qui ont fournis tant d'efforts avec moi depuis mes toutes premières années d'études et sans qui tout cela n'aurait jamais été possible. Aussi en écrivant ces mots je ne peux m'empêcher de dédier ce travail à la mémoire de la sœur de mon binôme Allah yerhamha, que dieu l'accueille dans son vaste paradis inchallah.

Remerciements

Je remercie tous ceux qui ont contribué à la réalisation de notre projet, à leur tête Madame SELADJI Yassamine. Et ce pour avoir accepté de nous encadrer dans ce thème et pour tous les efforts et sacrifices qu'elle a réalisés avec nous.

Je remercie aussi Monsieur Mahfoud CHERIF BENMOUSSA pour sa grande gentillesse et sa disponibilité, et aussi pour nous avoir mis à l'aise dans le service de néphrologie dès notre premier entretien.

Enfin je ne remercierai jamais assez Madame HALFAOUI Amal pour son dévouement et ses grandes qualités humaines et professionnelles, sans son aide la dernière partie de notre travail n'aurait jamais été fonctionnelle.

Malik GHERNAOUT

Tables des matières

Introduction Générale	4
La Problématique	5
CHAPITRE I : Contexte de travail	1
1 Introduction.....	7
2 Présentation du CHU de Tlemcen	7
3 Service de néphrologie.....	7
3.1 Insuffisance rénale	8
3.2 Dialyse et hémodialyse :	9
3.2.1 Dialyse péritonéale :	10
3.2.2 Hémodialyse :	11
3.3 Volonté et les acquis du service de néphrologie du C.H.U de Tlemcen :	12
3.3.1 Description de l'existant	13
3.3.2 Cahier des charges :	13
4 Conclusion	14
CHAPITRE II	15
Modélisation du système	15
1 Introduction.....	16
2 Les diagrammes de SMARTDIALYSE	17
2.1 Diagramme de cas d'utilisation.....	17
2.2 Diagrammes de séquences	21
2.3 Digramme de classes	24

2.4	Design pattern	26
3	Conclusion	28
CHAPITRE III		29
Réalisation de SMARTDIALYSE		29
1	Introduction.....	30
2	Structuration de l'application.....	30
3	Architecture J2EE	31
3.1	Définition	31
3.2	Couche métier	32
3.2.1	Enterprise Javabeans.....	33
3.2.2	Java ServerFaces (JSF)	35
3.3	Couche accès aux données	38
3.3.1	Caractéristique de MySQL	38
3.4	Couche présentation	38
3.4.1	Facelets	38
3.4.2	Primefaces.....	39
3.4.3	JasperSoft studio	40
3.4.4	JasperReport.....	40
3.4.5	Lazy loading	41
4	Présentation de SMARTDIALYSE	42
5	Conclusion	48
CHAPITRE IV		49
Entrepôt de données		49

1	Introduction.....	50
2	Business intelligence (BI).....	50
3	Processus de construction de l'entrepôt « SMARTDIALYSE ».....	51
4	Conception et modélisation de l'entrepôt « SMARTDIALYSE ».....	52
4.1	Définition de l'entrepôt de données	52
4.2	Schéma en étoile de la base de l'entrepôt	53
4.3	Modélisation du cube « SMARTDIALYSE »	54
5	Préparation des données : Extraction, transformation et alimentation de la base de l'entrepôt.....	56
6	Restitution des données : exploitation du cube.....	57
7	Conclusion	60
	Conclusion Générale.....	61
	Bibliographie	62
	Liste des figures	65
	Liste des abréviations.....	67
	Résumé.....	69

Introduction Générale

Avec l'évolution fulgurante et continue de la technologie l'outil informatique est devenu indispensable dans la vie quotidienne de l'individu dans pratiquement toutes les tâches qu'il entreprend, que ça soit sur le plan personnel ou professionnel au niveau des administrations et institutions, notamment avec la facilité et le gain de temps considérable qui en résulte.

L'adaptation avec ce changement et l'informatisation de tous les secteurs (administration, santé...) est devenue une nécessité. Dans cette optique, il y'a une volonté de notre part d'être une partie active pour l'intégration du CHU de Tlemcen dans cette évolution technologique par l'amélioration de ses services sur différents plans notamment celui de l'administration, et comme étant étudiant au département de l'informatique on a développé un sens de l'initiative afin d'améliorer le rendement de l'administration hospitalière par l'introduction de cet outil dans son travail quotidien.

Nous nous intéressons particulièrement au service de néphrologie du CHU de Tlemcen. Le service de néphrologie prend en charge les personnes atteintes de maladies rénales aiguës ou chroniques. Au cours de leur évolution, certaines de ces maladies vont se compliquer d'insuffisance rénale chronique nécessitant à leur stade terminal un traitement par dialyse (Hémodialyse ou dialyse péritonéale).

A partir de ce constat on vise à faciliter la tâche du corps médical du service de néphrologie dans la gestion des patients par l'informatisation de leur méthode de travail afin de garantir une gestion fiable et optimale et par conséquent améliorer les services aux bons soins des patients souffrant d'insuffisance rénale.

La Problématique

Ce travail consiste à contrer une réalité non commode vécue sur le terrain dont les lacunes sont nombreuses, à savoir :

- Le caractère manuel dans le travail de l'administration.
- La lenteur dans la recherche du dossier d'un patient.
- Aucune solution numérique n'a été proposée auparavant dans ce cadre.
- Risque de perte des dossiers papiers des patients
- Absence de vision globale de l'ensemble des patients.

CHAPITRE I

Contexte de travail

1 Introduction

Dans le cadre de notre projet de fin d'études universitaires et parmi les nombreux thèmes qui nous ont été proposé nous avons choisi celui nous amenant vers C.H.U de Tlemcen et l'un de ses services les plus importants qui est celui de la néphrologie, notre tâche a été de développer et de mettre en place un système destiné à l'unité de dialyse du service de néphrologie du CHU de Tlemcen.

2 Présentation du CHU de Tlemcen

La construction de l'hôpital Tlemcen a débutée en 1947 et achevée en 1950. En 1986, il est érigé en centre hospitalier universitaire, Il prend le nom du docteur TIDJANI DAMERDJI médecin, patriote de la 1ère heure, martyr de la révolution algérienne. Actuellement il est composé de 44 services et laboratoires spécialisés. La capacité d'accueil est de 646 lits et couvre une population de 1.5 millions d'habitants [1].

3 Service de néphrologie

Parmi les 44 services du centre hospitalo-universitaire de Tlemcen on trouve le service de néphrologie qui est sous la direction du Professeur Benmansour.

Il prend en charge les personnes atteintes de maladies rénales aiguës ou chroniques. Au cours de leur évolution, certaines de ces maladies vont se compliquer d'insuffisance rénale chronique nécessitant à leur stade terminal un traitement par dialyse (Hémodialyse ou dialyse péritonéale) ou transplantation.

Ce service est composé de quatre unités principales.

- Hospitalisation.
- Dialysé péritonéale.
- Hémodialyse.
- Greffe rénale.

3.1 Insuffisance rénale

Faisant partie des cinq organes nobles dont dispose l'être humain, les reins sont indispensables au bon fonctionnement du corps de ce dernier. Ces organes filtrent 180 litres de notre sang par jour et éliminent les déchets de l'organisme.

Les reins sont localisés dans la partie postérieure de l'abdomen, de part et d'autre de la colonne vertébrale sous les dernières côtes. Chaque rein mesure environ 11 cm de long et est relié à l'artère aorte et à la veine cave inférieure par l'artère et la veine rénale [2].

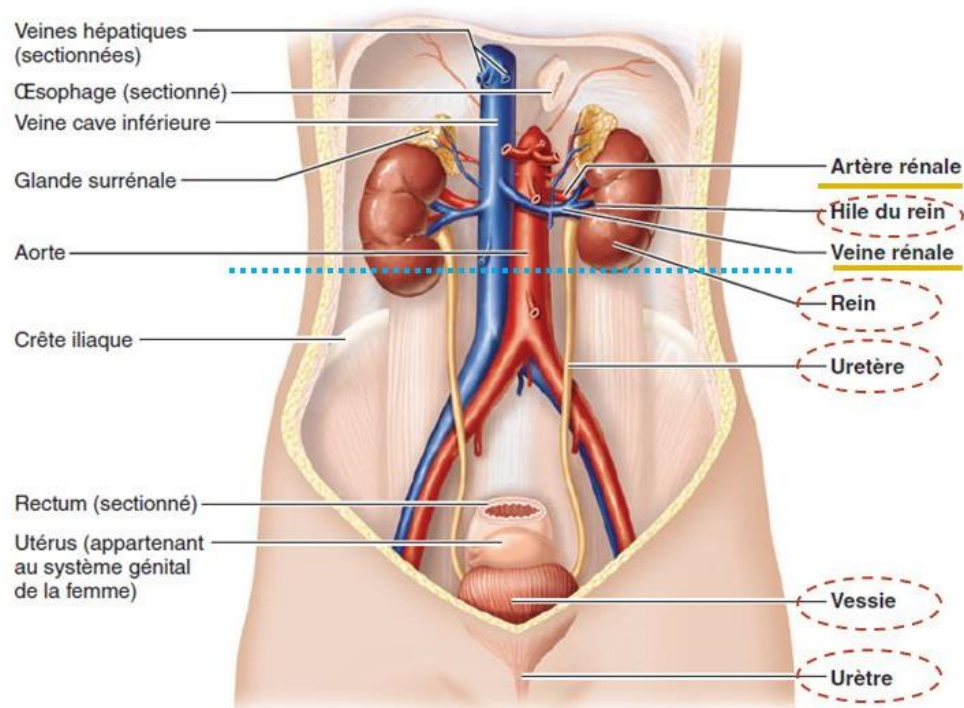


Figure 1 Anatomie macroscopique du système urinaire [3]

Chaque rein est composé d'environ un million de néphrons (glomérule tubule), dont le rôle est de filtrer les différentes substances contenues dans le sang pour ensuite réabsorber ce qui est encore utile en laissant les déchets s'éliminer par l'urine [2].

L'urine est donc générée par les reins, elle transite dans les uretères pour atteindre la vessie où elle est stockée puis éliminée lors d'une miction via l'urètre [2].

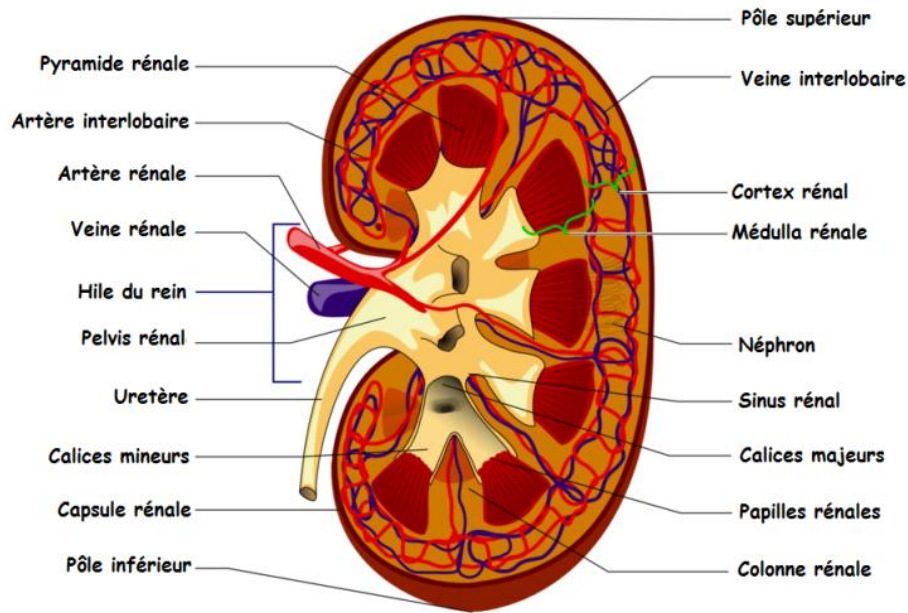


Figure 2 Vue générale du rein [4]

L'insuffisance rénale correspond à une modification du fonctionnement des deux reins qui ne filtrent plus correctement le sang. Ce qui peut engendrer de graves pathologies chez les personnes souffrant de cet insuffisance pouvant aller jusqu'au décès en cas de soins non adéquats [5] , car en effet il existe deux techniques de soin permanent qui aide les patients à avoir une vie relativement normale il s'agit de l'hémodialyse et la dialyse péritonéale.

3.2 Dialyse et hémodialyse :

La dialyse est une technique médicale qui permet de purifier le sang. Cette technique est utilisée pour aider des patients atteints d'insuffisance rénale, c'est à dire dont les reins ne parviennent plus à évacuer certaines toxines présentes dans le sang. La dialyse reproduit donc la fonction principale du rein qui est l'élimination des déchets créés ou absorbés par l'organisme.

On trouve actuellement deux types de dialyse dont la fonction et l'application sont différentes : la dialyse péritonéale et l'hémodialyse. Toutes deux visent à atteindre le même objectif de reproduction de l'activité rénale. La différence est que la dialyse péritonéale agit à l'intérieur de l'organisme, alors que l'hémodialyse se fait par l'intermédiaire d'un appareillage de circulation externe à l'organisme.

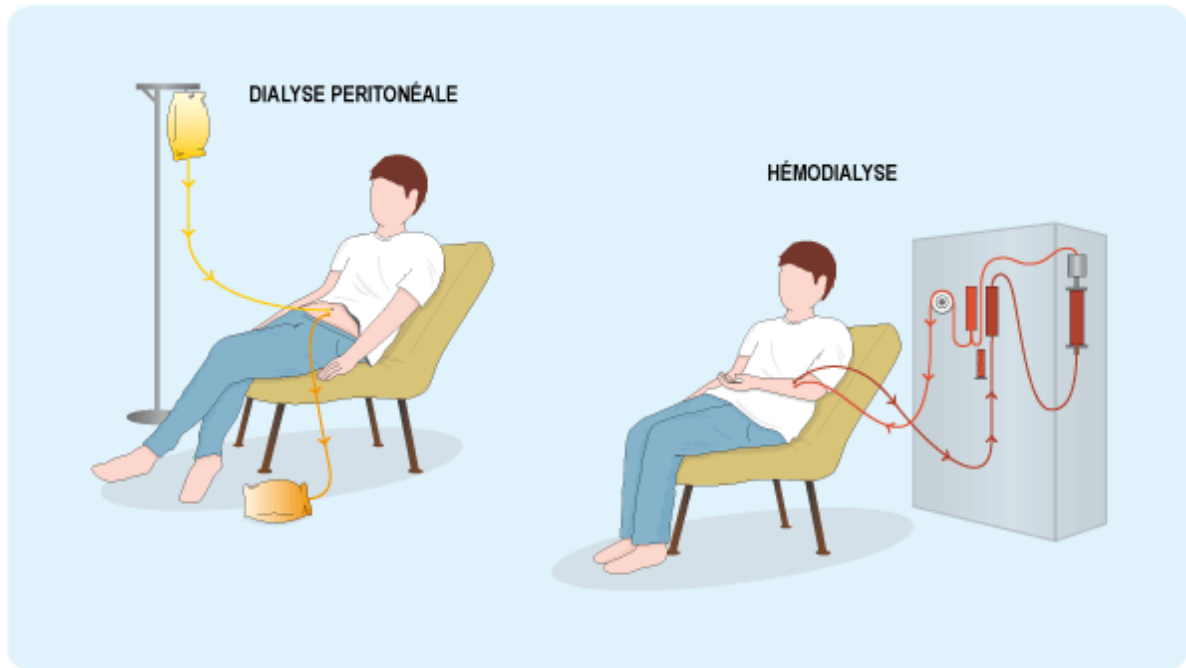
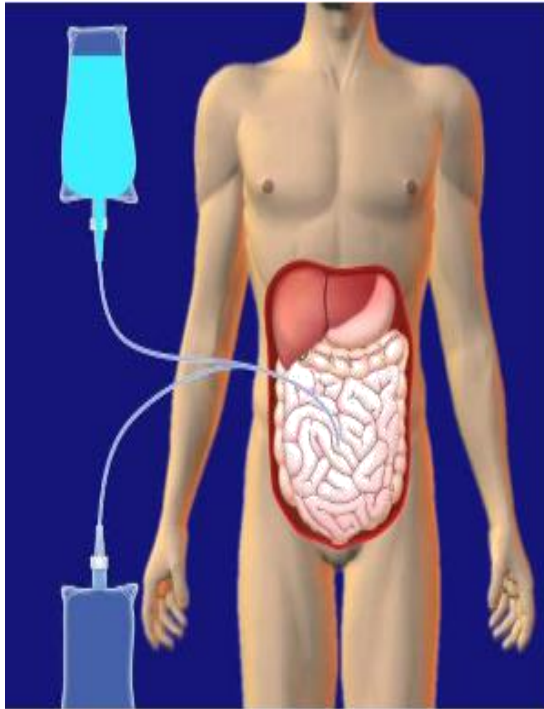


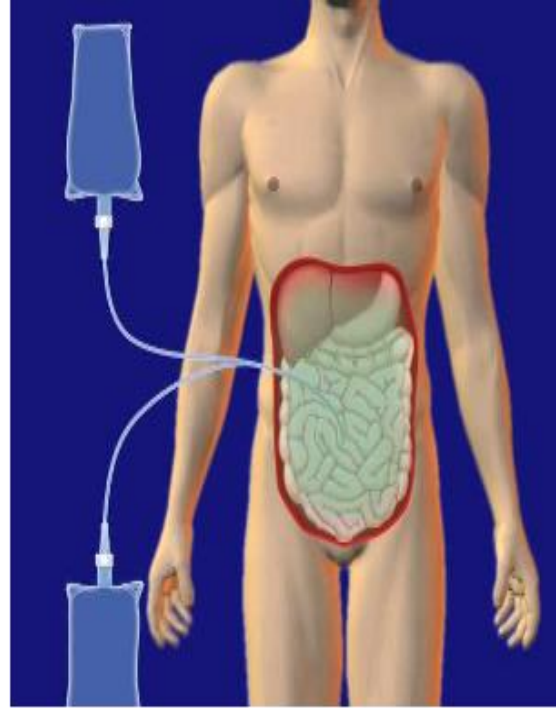
Figure 3 : Méthodes de dialyse [6]

3.2.1 Dialyse péritonéale :

La dialyse péritonéale est une méthode qui utilise le péritoine (membrane de l'abdomen) comme filtre pour épurer le sang des substances toxiques. Un liquide appelé dialysat est introduit dans la cavité abdominale par un cathéter afin de rééquilibrer le plasma sanguin. Ses composants favorisent l'élimination de l'eau excédentaire et des substances toxiques produites par l'organisme. L'avantage de cette technique pour le patient est d'être autonome et à domicile. Les gestes de connexion et de déconnexion des poches au cathéter doivent faire l'objet d'une hygiène rigoureuse car elles peuvent engendrer des infections de gravité diverse [7].



**Injection du liquide dans
l'abdomen**



**Drainage du liquide chargé des
toxines**

Figure 4 : Dialyse péritonéale [8]

3.2.2 Hémodialyse :

L'hémodialyse rénale est une technique permettant l'épuration du sang à l'aide d'une machine extracorporelle reproduisant la fonction rénale. Cette méthode permet de remplacer la fonction du rein déficient, soit de manière définitive pour l'insuffisance chronique, soit de manière non définitive, dans l'attente de la récupération de la fonction rénale pour l'insuffisance aiguë. L'hémodialyse est donc un traitement palliatif des insuffisances rénales majeures [6].

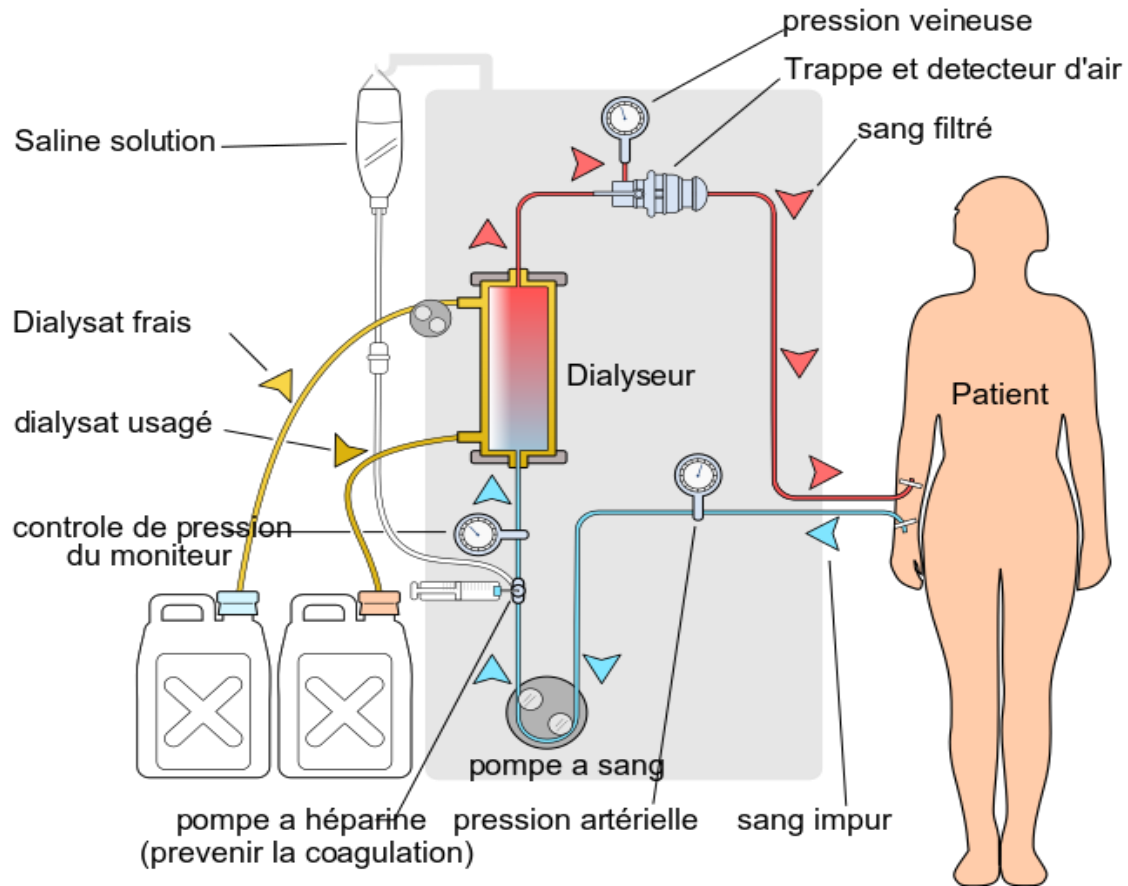


Figure 5 Le Processus d'hémodialyse [9]

3.3 Volonté et les acquis du service de néphrologie du C.H.U de Tlemcen :

Composé de quatre unités, le service de néphrologie de Tlemcen dispose d'un personnel qui cherche à proposer une meilleure qualité de soin. L'informatisation des dossiers des patients devient indispensable et nécessaire pour gérer quelques centaines de patients qui dialyse à vie, avec des dossiers avec un volume important de données. Ainsi un support logiciel est bien que nécessaire pour améliorer cette gestion et cette prise en charge. Le service de néphrologie de Tlemcen a déjà tenté l'expérience informatique dans l'unité de greffe rénale (logiciel GREFFON) qui a encouragé le personnel à s'investir dans ce nouveau projet. Ce travail que nous avons été amenés à réaliser englobe le fonctionnement de deux unités de ce service qui sont

- Unité des dialysé péritonéale.
- Unité d'hémodialyse.

3.3.1 Description de l'existant

La gestion actuelle des dossiers médicaux, personnels et des examens de tous les patients du service de néphrologie de Tlemcen se fait à travers des fiches et des documents en format papier ce qui favorise très fortement la perte d'informations et engendre de grandes difficultés dans tout ce qui est archivage des dossiers des patients mais aussi rallonge le temps de recherche de ces dossiers dans le cadre des rendez-vous récurrent des patients.

A l'instar de tous les autres services du C.H.U de Tlemcen celui de la néphrologie est complètement investi pour assurer le bien-être et la bonne gestion de ces patients pour ce faire les responsables de ce service ont décidé de suivre la voie de l'informatisation de leur travail afin d'assurer la fluidité dans le fonctionnement de ce dernier.

Le dossier d'un patient du service de néphrologie se compose essentiellement en 3 parties :

- Le dossier personnel contenant toute les informations relatives au patient (Nom, Prénom, Profession, Fratrie, Statut matrimonial ...).
- Le dossier médical qui contient les informations nécessaires pour les médecins (Groupage, Rhésus, Antécédents médicaux, Antécédents chirurgicaux, Antécédents Familiaux, Première séance de dialyse ...).
- Les différents examens et tests effectué par le patient au sein de ce service.

3.3.2 Cahier des charges :

Dans le cadre de ce projet les responsables nous ont confié la tâche de développer une plateforme numérique qui répond à la problématique présentée précédemment, nous avons été amené à avoir des discussions et des entretiens avec le personnel médical du service de néphrologie du C.H.U de Tlemcen, un document spécifiant les exigences du personnel médical nous a été remise par le docteur CHERIF BENMOUSSA Mahfoud ,document sur lequel nous nous sommes basé pour rédiger un cahier des charges contenant tous les besoins et exigences recensées ,le cahier des charges rédigé sera annexé à ce mémoire

4 Conclusion

La volonté des responsables d’informatiser le service de néphrologie du C.H.U de Tlemcen est en soi un pas très important vers le développement de ce secteur hospitalier. La création de cette application constitue une étape de plus vers l’informatisation total de ce service et aide de manière non négligeable les médecins à se concentré que sur leurs patients.

CHAPITRE II

Modélisation du système

1 Introduction

Après avoir présenté le service de néphrologie et évoqué les différentes techniques de dialyse effectuées au sein du service, nous entamons une phase fondamentale dans le cycle de vie d'un logiciel, la phase de modélisation.

La modélisation apporte une grande rigueur, offre une meilleure compréhension des logiciels, et facilite la comparaison des solutions de conception avant leur développement.

Cette démarche se fonde sur des langages de modélisation, qui permettent de s'affranchir des contraintes des langages d'implémentation. Pour cela le système doit être modélisé selon une méthodologie d'analyse et de conception. La méthode choisie est basée sur le langage de modélisation unifié UML.

Dans le cadre d'un projet informatique, le recours à la modélisation UML procure de nombreux avantages qui agissent sur :

- La modularité
- L'abstraction
- La dissimulation
- La structuration cohérente des fonctionnalités et des données

Il permet aussi dans un premier temps de bien définir les besoins clients, et ainsi d'éviter des surcoûts liés à la livraison d'un logiciel qui ne satisfait pas le client. De plus, la modélisation UML permet de vulgariser les aspects liés à la conception et à l'architecture, propres au logiciel, au client. Aussi, elle apporte une compréhension rapide du programme à d'autres développeurs externes en cas de reprise du logiciel et facilite sa maintenance.

2 Les diagrammes de SMARTDIALYSE

UML permet de construire plusieurs modèles d'un système : certains montrent le système du point de vue des utilisateurs, d'autres montrent sa structure interne, d'autres encore en donnant une vision globale ou détaillée. Les modèles se complètent et peuvent être assemblés [10]. Ils sont élaborés tout au long du cycle de vie du développement d'un système (depuis le recueil des besoins jusqu'à la phase de conception).

Dans ce chapitre, nous allons étudier trois modèles qui sont :

- Diagramme de cas d'utilisation.
- Diagramme de séquence.
- Diagramme de classes.

2.1 Diagramme de cas d'utilisation

Le premier à construire est le diagramme de cas d'utilisation. Il permet de recueillir, d'analyser et d'organiser les besoins. Avec lui débute l'étape de conception du système.

En prenant en compte des besoins spécifiés dans le cahier des charges, nous avons élaboré le diagramme suivant :

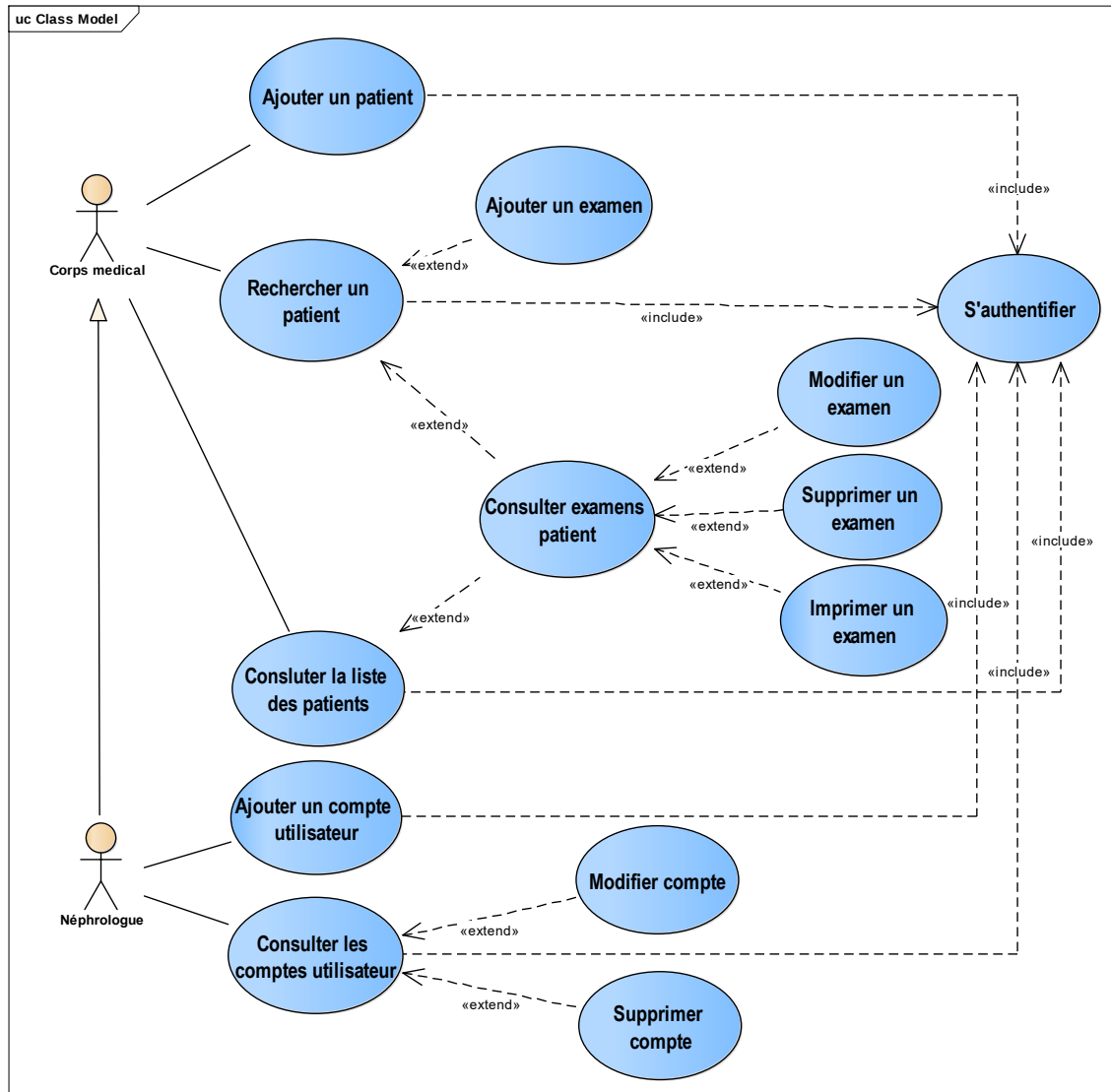


Figure 6 Diagramme de cas d'utilisations

➤ Identification des acteurs

Un acteur représente un rôle joué par une entité externe (utilisateur humain, dispositif matériel ou autre système) qui interagit directement avec le système étudié [10]. Conformément à la demande du client, nous avons recensé trois acteurs potentiels dont la différence majeure entre eux se situe au niveau des privilèges qui leurs sont octroyés. Ces derniers ont la possibilité d'interagir avec l'application, c'est-à-dire consulter et/ou modifier directement l'état du système, en émettant et/ou en recevant des messages susceptibles d'être porteurs de données.

Ces acteurs sont définis par, l'utilisateur, le corps médical et le néphrologue.

➤ *Identification des cas d'utilisations*

Un cas d'utilisation représente un ensemble de séquences d'actions qui sont réalisées par le système et qui produisent un résultat observable intéressant pour un acteur particulier.

Chaque cas d'utilisation spécifie un comportement attendu du système considéré comme un tout, sans imposer le mode de réalisation de comportement. Il permet de décrire ce que le futur système devra faire, sans spécifier comment il le fera. [10]

Le système comporte plusieurs cas d'utilisation que nous allons décrire ci-dessous :

- **Authentification** : Avant d'effectuer des opérations sur le système, l'utilisateur doit impérativement passer par cette étape de distribution des droits d'accès
- **Ajouter un compte utilisateur** : Un utilisateur avec les privilèges de néphrologue aura le droit d'ajouter d'autres utilisateurs.
- **Consulter les comptes utilisateur** : Procure la possibilité au néphrologue de consulter les comptes des utilisateurs.
- **Modifier le compte** : Une fois le compte utilisateur identifié le néphrologue à la possibilité d'apporter des modifications.
- **Supprimer compte** : Le néphrologue dispose de la capacité de suppression des comptes utilisateurs
- **Ajouter un patient** : Cette fonctionnalité offre la possibilité aux utilisateurs du système d'insérer de nouveaux patients.
- **Rechercher un patient** : L'une des fonctionnalités les plus utiles, la recherche permet de retrouver un patient parmi tous ceux qui sont enregistré, elle peut être réalisée selon trois critères : nom, prénom et date de naissance.
- **Consulter la liste des patients** : Pour présenter l'ensemble des patients se trouvant dans le système.
- **Ajouter un examen** : permet au néphrologue d'ajouter une multitude d'examens que le patient devra effectuer dans ou en dehors du service de néphrologie.

- **Consulter examens patients** : fonctionnalité représentant l'ensemble des examens relatif à un seul patient.
- **Modifier un examen** : tout examen existant pourra être modifier a la guise du néphrologue.
- **Supprimer un examen** : afin d'avoir que les examens ayant une importance aux yeux du néphrologue cette fonctionnalité a été ajouté
- **Imprimer un examen** : faisant partie des précieuses fonctionnalités qu'offre le système elle permet d'imprimer tous les examens et dossier médicale souhaités par le néphrologue et ce par intervalle de date désiré.

➤ Les relations entre acteurs

La seule relation entre acteurs est la relation de généralisation. Quand un acteur fils hérite d'un acteur père, il hérite en réalité de toutes les associations du père [11].

Suite à la demande du client, notre application va permettre des futures évolutions, et donc l'ajout de nouveaux utilisateurs par la suite.

Pour cela, on a utilisé cette relation de généralisations dans notre diagramme.

- L'acteur fils « corps médical » hérite de l'acteur père « utilisateur » : le corps médical peut faire avec le système tout ce que peut faire une personne, plus d'autres choses.

➤ Les relations entre cas d'utilisation

Dans notre diagramme de cas d'utilisation nous avons utilisé les relations d'inclusion et d'extension qui servent à enrichir des cas d'utilisation par d'autres cas d'utilisation.

- L'inclusion « include » : Cela implique obligatoirement l'inclusion d'un cas d'utilisation dans un autre comme ici « Ajouter un patient » fait obligatoirement appel à « S'authentifier ».
- L'extension « extended » : Cela permet éventuellement l'extension d'un cas d'utilisation par un autre comme ici « Consulter examens patient » peut étendre « imprimer un examen ».

2.2 Diagrammes de séquences

Les diagrammes de séquences sont la représentation graphique des interactions entre les acteurs et le système selon un ordre chronologique dans la formulation.

Voici quelques notions de base du diagramme :

- **Scénario** : Représente une succession particulière d'enchaînements, s'exécutant du début à la fin du cas d'utilisation, un enchaînement étant l'unité de description de séquences d'actions [10].
- **Ligne de vie** : Représente l'ensemble des opérations exécutées par un objet [10].
- **Message** : Un message est une transmission d'information unidirectionnelle entre deux objets, l'objet émetteur et l'objet récepteur. Dans un diagramme de séquence, deux types de messages peuvent être distingués [10]:
 - **Message synchrone** : Dans ce cas l'émetteur reste en attente de la réponse à son message avant de poursuivre ses actions.
 - **Message asynchrone** : Dans ce cas, l'émetteur n'attend pas la réponse à son message, il poursuit l'exécution de ses opérations.

Pour bien comprendre le dialogue qui se fait entre l'utilisateur, le système et la base de données, nous allons décrire les trois plus importants cas d'utilisation sous forme de séquences chronologique.

❖ Séquence « Authentification »

Une fois l'application lancé, une page d'authentification s'affiche dans le navigateur du client, ce dernier introduit le nom d'utilisateur et son mot de passe. Le système envoie une requête contenant les paramètres saisis vers le serveur de base de données, qui retournera deux réponses possibles, selon ces derniers l'utilisateur est redirigé vers la page d'accueil si ses informations son correcte, sinon une alerte apparait l'invitant à réessayer. Ce processus est illustré par la figure suivante :

❖ Séquence « Ajouter un patient »

Avant toute opération effectuée sur le système, une authentification est requise. L'utilisateur remplit le formulaire d'ajout avec les informations nécessaires et achève cette opération. La vérification et validation se feront au niveau du système. Deux possibilités sont donc envisagées, Dans le cas où les données sont validées, ces dernières sont envoyées vers la base de données et enregistrées, un message de succès est alors envoyé à l'utilisateurs. Dans le cas contraire une erreur de validation est notifiée, La figure suivante illustre le déroulement de cette séquence.

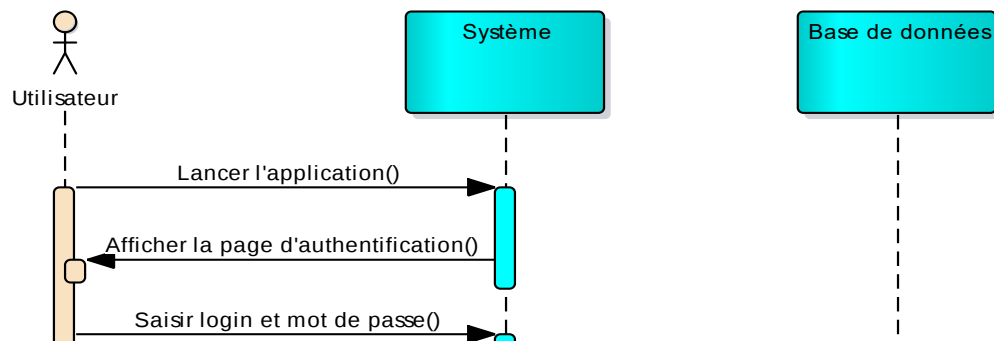
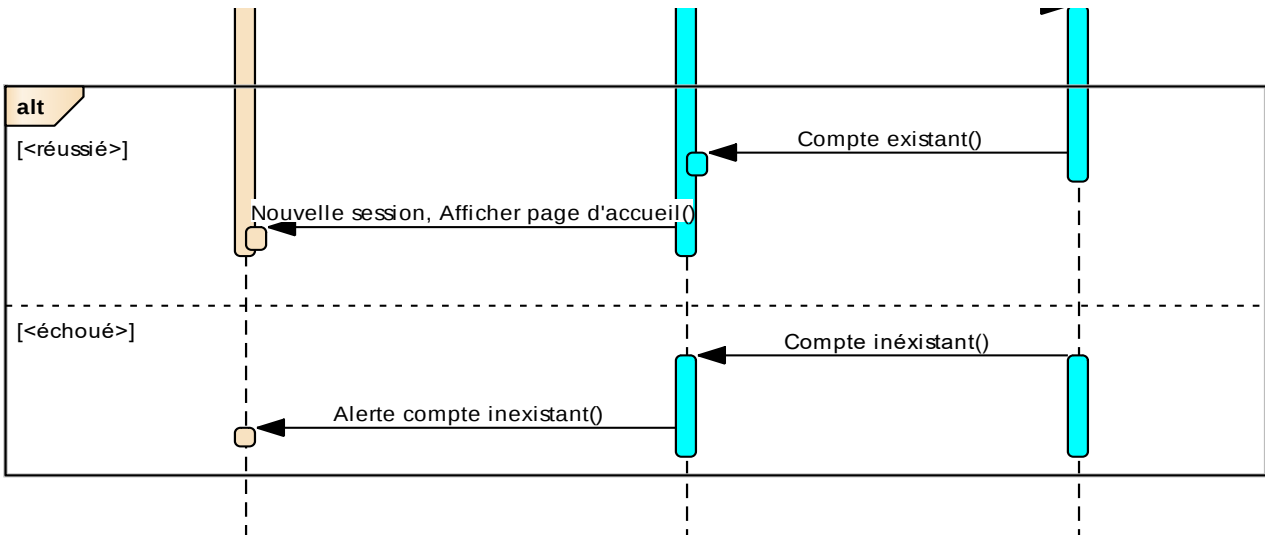


Figure 7 Séquence Authentification



❖ Séquence « Recherche de patient »

Comme toute opération, la recherche requiert aussi l'authentification de la part de l'utilisateur. Une fois connecté, deux possibilités de recherche lui sont offertes. Pour la recherche rapide l'utilisateur introduit un seul paramètre qui est pris en considération (nom, prénom ou date de naissance), Par contre la recherche paramétrée utilise les trois paramètres simultanément pour filtrer les patients. Le système récupère les paramètres de recherche et

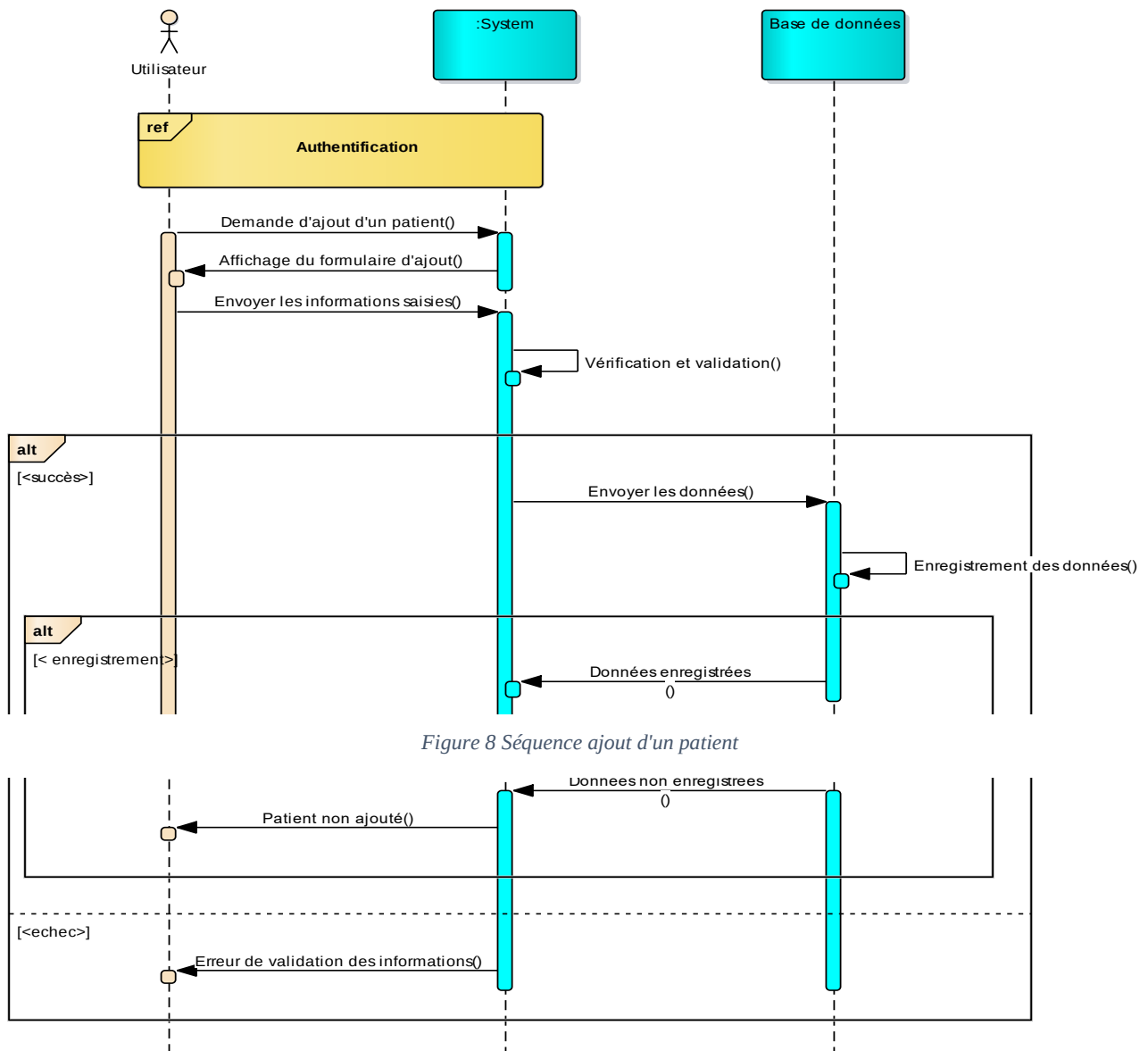


Figure 8 Séquence ajout d'un patient

les envois vers de base de données qui a son tour retourne les patient retrouvé ou une liste vide dans le cas contraire.

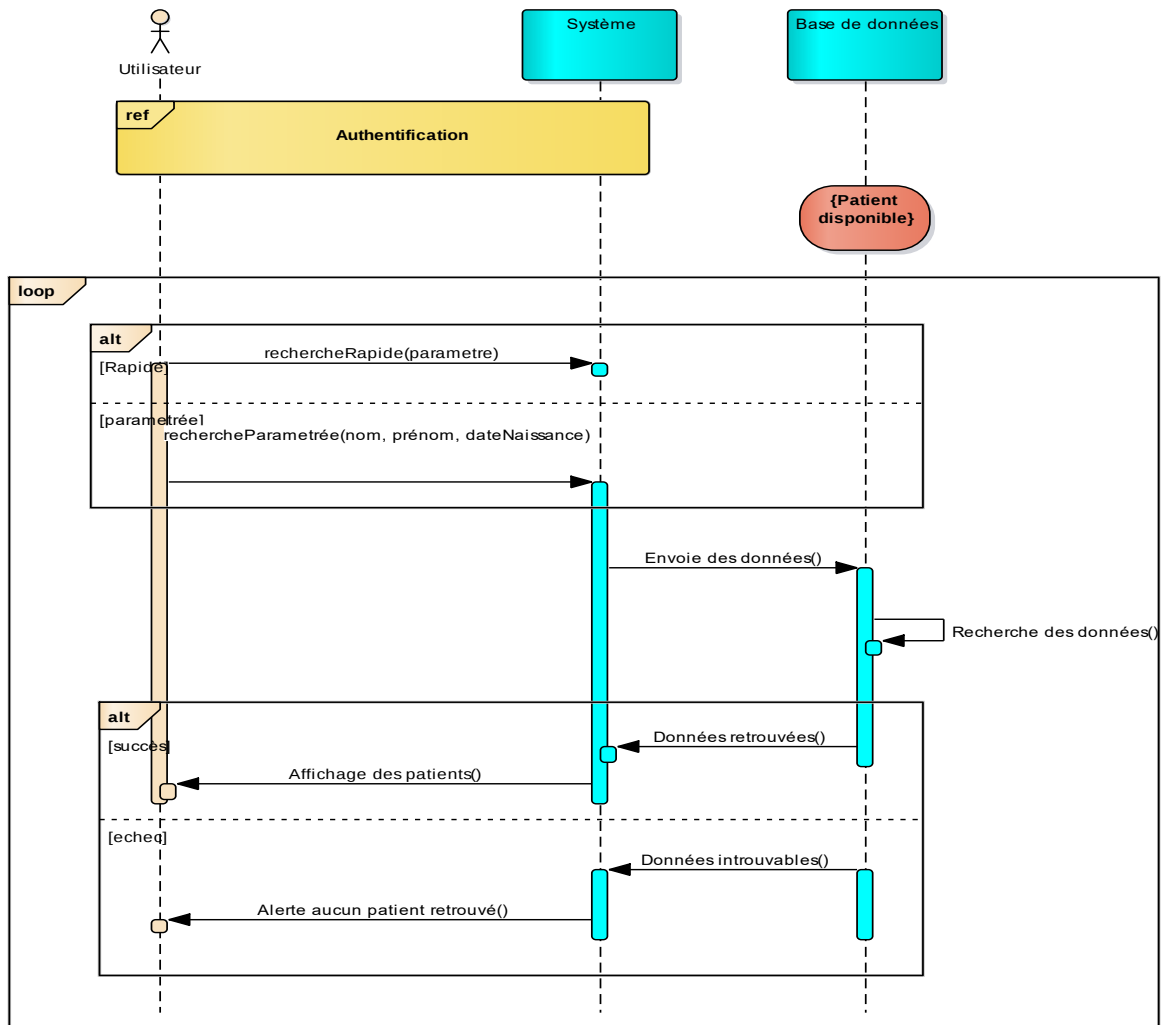


Figure 9 séquence recherche

2.3 Diagramme de classes

Le diagramme de classes est considéré comme le plus important de la modélisation orientée objet. Alors que le diagramme de cas d'utilisation montre un système du point de vue des acteurs, le diagramme de classes en montre la structure interne. Il contient principalement des classes. Une classe contient des attributs et des opérations. Le diagramme

CHAPITRE II : Modélisation du système

de classes n'indique pas comment utiliser les opérations : c'est une description purement statique d'un système [10].

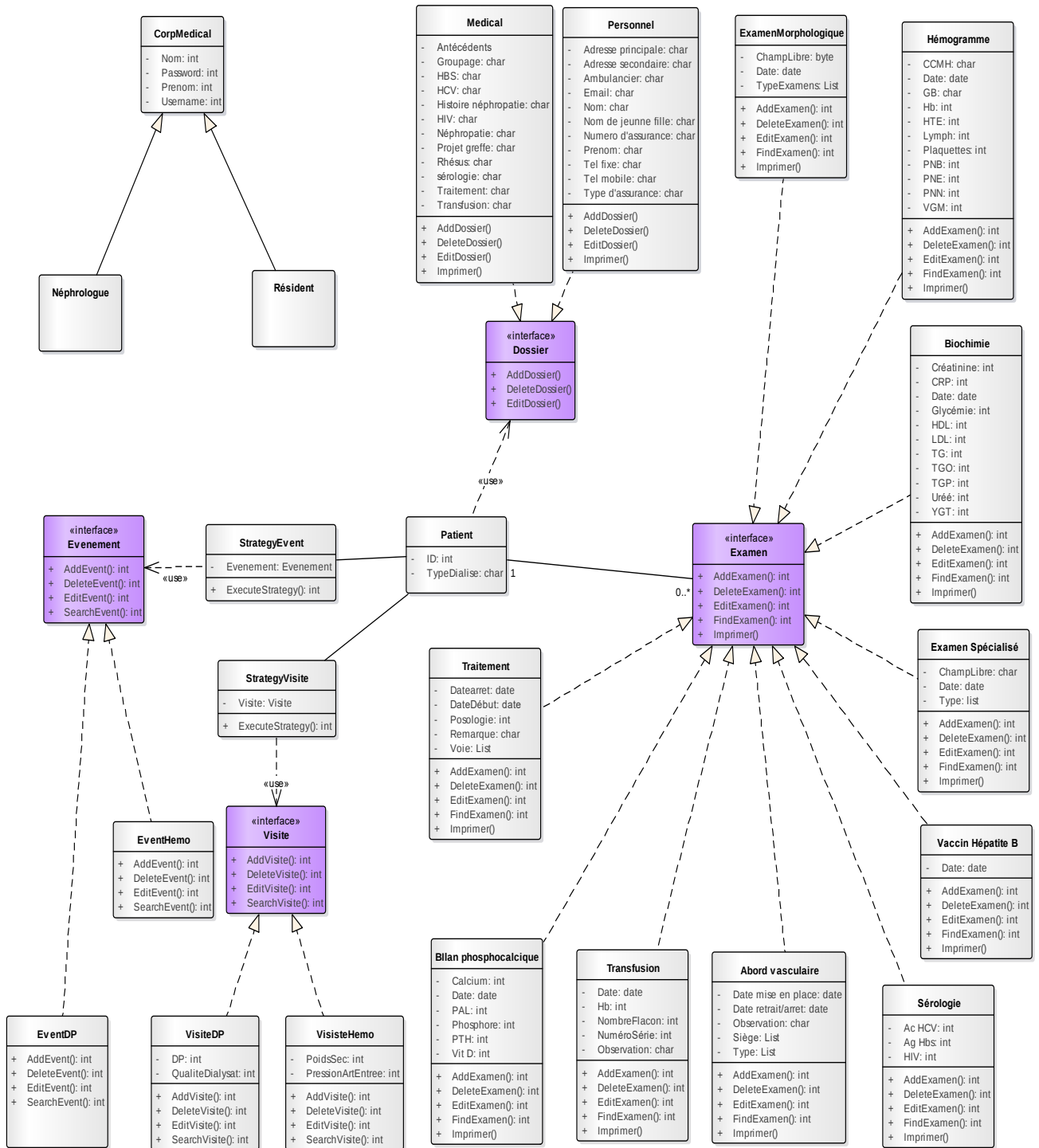


Figure 10 Diagramme de classes SMARTDIALYSE

2.4 Design pattern

Afin de mettre en avant les bonnes pratiques de conception. L'utilisation des Design Patterns offre de nombreux avantages. Tout d'abord cela permet de répondre à un problème de conception grâce à une solution éprouvée et validée par des experts. Ainsi on gagne en rapidité et en qualité de conception ce qui diminue également les coûts. Dans cette optique nous avons choisi d'utiliser les designs pattern suivants :

❖ *Design pattern « strategy »*

Le patron « Strategy » cherche principalement à séparer un objet de ses comportements/algorithmes en encapsulant ces derniers dans des classes à part. Pour ce faire, on doit alors définir une famille de comportements ou d'algorithmes encapsulés et interchangeables. [12]

Nous pouvons donc en tirer plusieurs avantages parmi eux :

- Mise en place de bonnes pratiques,
- Code extensible avec aisance d'inter changer les algorithmes,
- Code facile à maintenir en cas d'évolution/changement.

Pour en tirer profit de ces avantages, et comme la médecine et en évolution continue, et grâce à ce patron de conception nous pouvons dire que notre application supportera les évolutions. Nous avons introduit le patron de conception « strategy » dans SMARTDIALYSE, et cela pour permettre une extensibilité optimale des évènements et des visites periodiques effectuées pour chaque patient. Cela est représenté par la figure suivante.

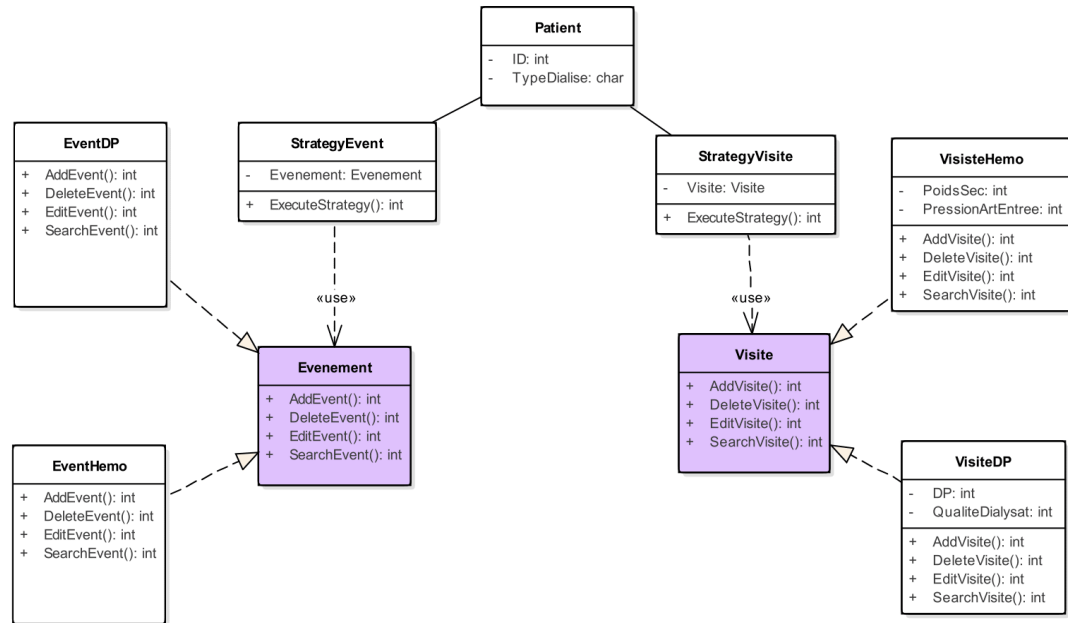


Figure 11 Design pattern strategy adapté à SMARTDIALYSE

❖ Design pattern « façade »

Une bonne pratique de conception est d'essayer de limiter le couplage existant entre des fonctionnalités proposées par différentes entités. Dans la pratique, il est préférable de développer un petit nombre de classes et de proposer une classe pour les utiliser. C'est ce que propose le motif de conception façade. [13]

Le but est de proposer une interface facilitant la mise en œuvre d'un ensemble de classes généralement regroupées dans un ou plusieurs sous-systèmes. Le motif Façade permet d'offrir un niveau d'abstraction entre l'ensemble de classes et celles qui souhaitent les utiliser en proposant une interface de plus haut niveau pour utiliser les classes du sous-système. [14]

Dans le cas de notre étude et plus précisément lors de l'implémentation du lien entre la couche métier basée sur les EJB et la couche accès aux données, nous avons utilisé le patron de conception « façade » pour la réalisation des sessions beans, qui donne accès à la base de données. Ce patron est indépendant du diagramme de classe, car il représente une particularité de la couche métier.

CHAPITRE II : Modélisation du système

La figure suivante nous avons pris l'exemple des sessions (dossierPersonnel et examenBiochimie)

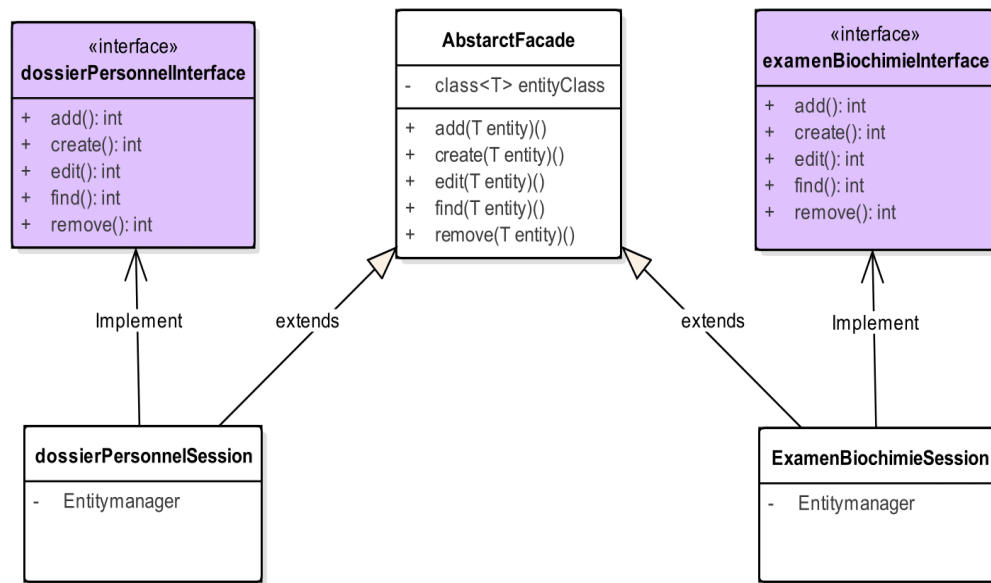


Figure 12 Design pattern façade à SMARTDIALYSE

3 Conclusion

A l'issue de ce chapitre, nous avons réussi à élaborer une structure de base pour le système désiré, et ce en réalisant un diagramme de classe primordiale pour l'implémentation orientée objet de la logique applicative. Cette partie nous a permis de bien organiser notre travail, afin d'arriver à la finalité du projet. Ce qui rend cette étape de conception un intermédiaire entre l'idée à l'état brut et à la concrétisation sur le terrain.

CHAPITRE III

Réalisation de

SMARTDIALYSE

1 Introduction

Ce chapitre permet de recenser les différents outils de programmation utilisés qui ont servi à la réalisation de l'application et de ses fonctionnalités. Aussi, l'application sera expliquée et les différentes interfaces seront présentées

2 Structuration de l'application

Lors de l'étape de réflexion intervenant en amont du projet, il existait certaines questions qui subsistait quant à la structuration de cette application parmi elles figuraient la gestion d'un flux de données important, en effet un patient dialysé effectue un grand nombre d'examen chaque semaine, ce qui constitue une quantité de données non négligeables à traiter et à persister, et tout cela en gardant une stabilité accrue en ce qui concerne la performance que sollicite un accès simultané de plusieurs utilisateurs.

Pour standardiser le développement d'une application web robuste et distribuée, déployée et exécutée sur un serveur d'application, nous avons opté pour la plateforme Java Enterprise Edition appelée aussi J2EE. Cette technologie est basée sur l'architecture 3 tiers qui s'impose comme une évidence afin de garantir la flexibilité et la performance requises. Cette architecture est illustrée dans la « Figure 13 ».

La « Figure 13 » représente les trois tiers actifs dans une architecture J2EE. Pour effectuer toute opération l'utilisateur interagit avec le client, qui à son tour dirige les requêtes vers le serveur d'application, afin de communiquer avec la base de données.

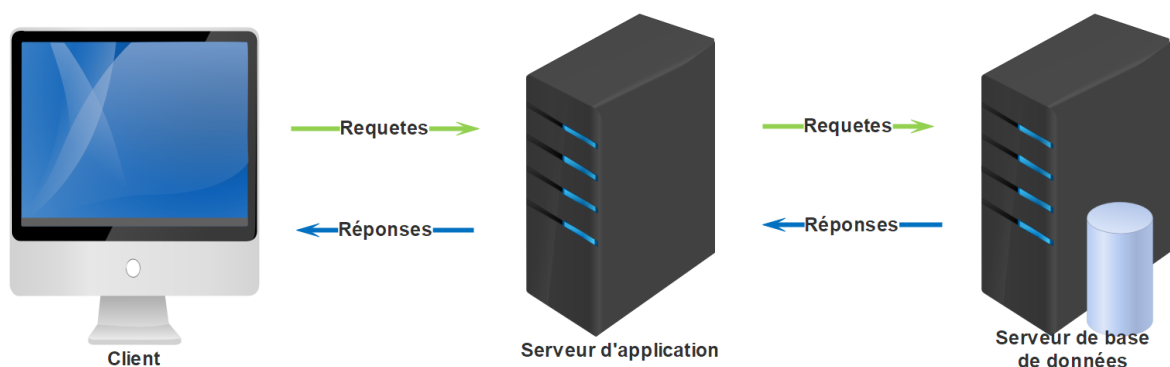


Figure 13 Architecture J2EE globale

3 Architecture J2EE

Dans le cadre de notre travail et afin de répondre aux besoins exigés de l'application, il nous a été favorable d'opter pour cette architecture logique qui va être détaillé dans ce qui suit.

3.1 Définition

Les applications au niveau serveur sont délocalisées, c'est-à-dire que chaque serveur est spécialisé dans une tâche (serveur web/ serveur de base de données par exemple). De plus, l'architecture J2EE repose sur des composants distincts, interchangeables et distribués [15], ce qui signifie notamment :

- Une plus grande flexibilité/souplesse.
- Une sécurité accrue car la sécurité peut être définie indépendamment pour chaque service, et à chaque niveau.
- Qu'un système reposant sur l'architecture J2EE peut posséder des mécanismes de hautes disponibilités.
- De meilleures performances, étant donné le partage des tâches entre les différents serveurs [15] .

SMARTDIALYSE est basée sur cette architecture qui dispose de trois niveaux, le client, le serveur d'application et le SGBD, étant donné que le deuxième est le plus important nous avons schématisé son comportement à travers la figure ci-dessous.

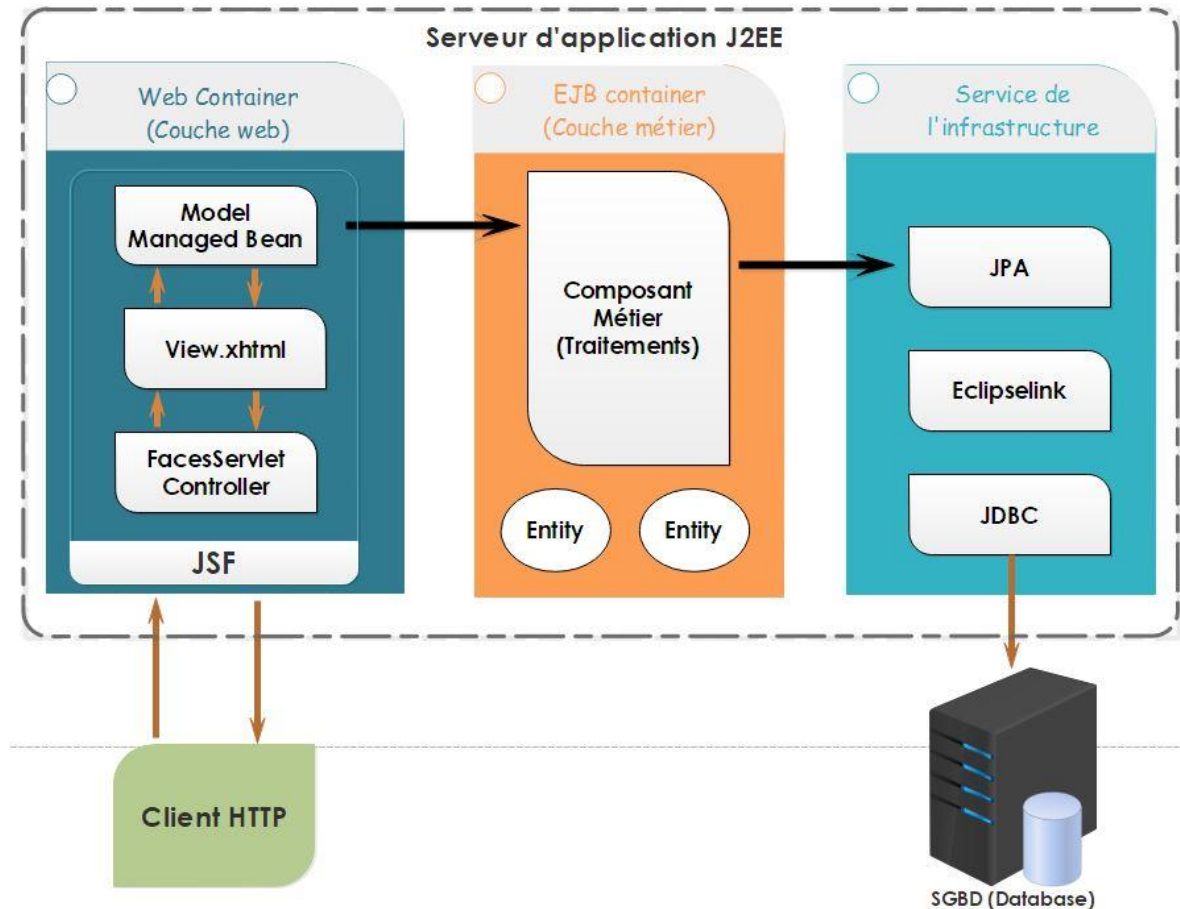


Figure 14 Architecture J2EE avec JSF et EJB

3.2 Couche métier

La couche métier correspond à la partie fonctionnelle de SMARTDIALYSE, celle qui fait le lien entre la couche présentation et la couche accès aux données, en fonction des requêtes des utilisateurs effectuées coté présentation. La couche métier applique les traitements nécessaires sur les données situées dans la base de données.

Etant donné que le serveur d'application est subdivisé en deux sous couche, la couche métier et la couche web, la gestion de la flexibilité et l'extensibilité devient primordiale. A cet effet nous avons choisi des technologies à la fois robustes et riches.

Pour répondre aux exigences citées au préalable en ce qui concerne la couche métier et en coordination avec l'architecture J2EE, nous avons utilisé la technologie de programmation par composants Enterprise JavaBeans (EJB 3.0).

3.2.1 Enterprise Javabeans

Afin de créer une séparation entre la partie accès aux données et la partie métier tout en respectant l'architecture J2EE, l'utilisation des EJB s'impose comme une évidence.

Comme étant des composants J2EE réutilisables et portables coté serveur, Les EJB consistent en des méthodes qui encapsulent la logique métier.

Une des principales caractéristiques des EJB est de permettre aux développeurs de se concentrer sur les traitements orientés métiers, car en coordination avec l'environnement dans lequel ils s'exécutent, ces derniers prennent en charge un certain nombre de traitements tel que la gestion des transactions, la persistance des données, la sécurité [16].

Les principaux composant d'EJB sont :

- **Entité bean** : modélise des concepts métier pouvant être désignés par des noms. Par exemple un bean entité peut représenter un client, un objet d'un stock. Autrement dit, les bean entité correspondent à des objets réels. Ces objets sont des enregistrements persistants stockés dans une base de données.
- **Session bean** : Encapsule une logique métier qui peut être invoquée par un client local, distant ou des web services. Pour accéder à une application déployée sur un serveur, le client fait appel aux méthodes du session bean qui effectuent les traitements nécessaires en exécutant la logique métier adéquate.

Afin de bien comprendre l'utilité des EJB dans SMARTDIALYSE, nous allons étudier l'exemple d'ajout d'un patient, en mettant l'accent sur le conteneur EJB.

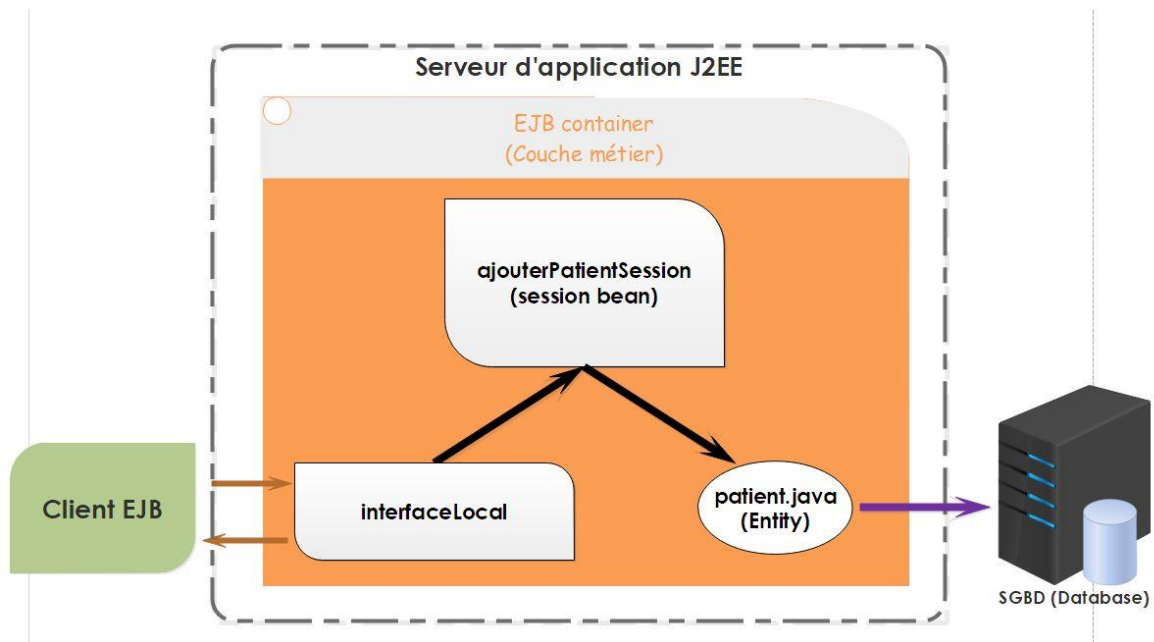


Figure 15 Exemple d'implémentation des EJB dans SMARTDIALYSE

Dans la « Figure 15 », dès que la demande d'ajout d'un patient est sollicitée à travers le client EJB, cette dernière est transférée vers l'interface local qui englobe les signatures des méthodes. Le bean session « ajouterPatientSession » implémente l'interface local en définissant le code logique de chaque méthode, ensuite cette session fait appel à l'entité patient qui permet l'accès aux données.

Une fois la logique métier de l'application implémentée, nous devons entamer la réalisation de la couche web et ce pour créer le lien entre les deux couches présentation et métier tout en gardant une séparation logique. Afin de mettre en œuvre ce lien et réaliser les attentes du projet nous avons opté pour le framework MVC basé sur la notion de composants qui est Java ServerFaces (JSF).

3.2.2 Java ServerFaces (JSF)

JSF est une technologie utilisée côté serveur, dont le but est de faciliter le développement de l'interface utilisateur en séparant clairement la partie « interface » de la partie « métier », d'autant que la partie interface n'est souvent pas la plus compliquée mais la plus fastidieuse. [17]

Cette séparation avait déjà été initiée avec la technologie JSP et particulièrement les bibliothèques de tags personnalisés. Mais JSF va encore plus loin en reposant sur le modèle MVC et en proposant de mettre en œuvre :

- La conversion et la validation des données avant leur utilisation dans les traitements.
- La gestion de l'état des composants de l'interface graphique.
- Le mapping HTML/Objet.
- Une liaison simple entre les actions côté client de l'utilisateur et le code Java correspondant côté serveur.
- Un modèle riche de composants graphiques réutilisables.

Pour son fonctionnement, JSF utilise une structuration similaire à celle d'une application web J2EE MVC traditionnelle. Dans SMARTDIALYSE :

- La vue est assurée par des pages .xhtml on parle alors de facelets.
- Le modèle est représenté par les entités beans des EJB.
- Le contrôleur est décomposé en deux éléments :
 - Une unique servlet mère servant de point d'entrée à toute requête, la FacesServlet.
 - Un JavaBean particulier, appelé aussi sous contrôleur, déclaré via une annotation et désigné par le terme *managed-bean*. [18]

Un bean managé est une classe java annotée (@ManagedBean) dont le cycle de vie est géré par le framework JSF. Ces beans font appel aux EJB pour effectuer les traitements qui

ne sont pas liés directement à l'interface utilisateur. Au sein d'une application ces beans peuvent prendre trois portées fondamentales qui sont :

- **Request** : permet de préciser au serveur que ce bean a pour portée une requête HTTP, c'est à dire entre l'émission de la requête et l'envoi de la réponse.
- **Session** : permet de sauvegarder l'état des objets durant de multiples requêtes HTTP, en d'autres termes la durée d'une session utilisateur.
- **Application** : permet l'accès aux données pour toutes les pages d'une même application quel que soit l'utilisateur.

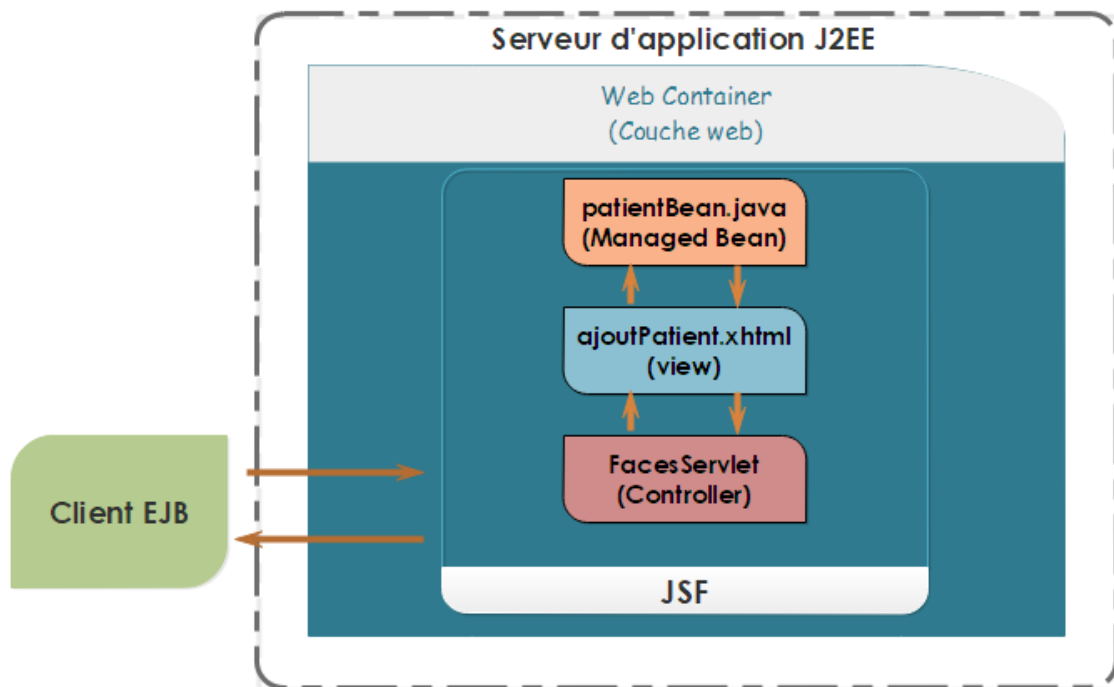


Figure 16 Conteneur web avec JSF

Afin de comprendre le fonctionnement de JSF dans SMARTDIALYSE, nous continuons de dérouler l'exemple de l'ajout d'un patient illustré dans la « Figure 16 », et cette fois en s'intéressant à la couche web.

Avant de solliciter le conteneur d'EJB comme nous l'avons cité au préalable, le conteneur web prend en charge le lien entre le client et la couche métier.

Dès que la demande d'ajout est envoyée, le « facesServlet » la transmet vers la vue adéquate, dans le cas de notre exemple c'est « ajoutPatient.xhtml ».

```
1. <h:form>
2.     <p:inputText value="#{patientBean.patient.nom}"/>
3.     <label>Nom</label>
4.     <p:inputText value="#{patientBean.patient.prenom}" />
5.     <label>Prénom</label>
6.     <p:commandButton value="Add" action="#{patientBean.addPatient()}" />
7. </h:form>
```

Figure 17 Bout de code ajoutpatient.xhtml

Comme nous pouvons le voir dans ce bout de code « Figure 17 », chaque composant (nom, prénom et bouton) de la vue est relié à une propriété du côté du managedBean « patientBean.java », ainsi le nom est associé à l'attribut nom de la propriété patient qui est instancier dans le managedBean. Ce dernier s'occupe de tous les traitements nécessaires à la vue et qui sont effectués du côté serveur. La « Figure 18 » décrit le fonctionnement du managedBean coté serveur.

```
1. @RequestScoped
2. @ManagedBean
3. public class patientBean{
4.     @EJB
5.     private ajoutPatientSession patientSession;
6.     private Patient patient= new Patient();
7.
8.     public void addPatient(){
9.         ...
10.        patientSession.create(patient);
11.        ...
12.    }
13. }
```

Figure 18 Bout de code patientBean.java

Pour conclure, la couche métier représente la partie la plus importante de cette architecture, vu son rôle d'intermédiaire indispensable entre la couche présentation et la couche accès aux données. L'accomplissement de cette couche aura bâti un socle déterminant en vue du bon fonctionnement de l'application.

3.3 Couche accès aux données

Dans la réalisation de SMARTDIALYSE nous devons impérativement choisir un SGBD qui prend en charge la structuration, la mise à jour et la maintenance de notre base de données. Sans négliger le flux de données important qui doit être géré. Alors pour toutes ces raisons nous avons opté pour MySQL l'un des SGBD les plus utilisés grâce à sa performance et sa fiabilité.

3.3.1 Caractéristique de MySQL

MySQL est un serveur de bases de données relationnelles SQL développé dans un souci de performances élevées en lecture, ce qui signifie qu'il est davantage orienté vers le service de données déjà en place que vers celui de mises à jour fréquentes et fortement sécurisées. Il est multi-thread et multi-utilisateurs.

C'est un logiciel libre développé sous double licence en fonction de l'utilisation qui en est faite : dans un produit libre ou dans un produit propriétaire. [19]

3.4 Couche présentation

Après avoir achevé la logique métier et implémenté le SGBD pour la couche accès aux données, nous nous sommes penché sur le côté client du système et comme nous avons utilisé le Framework JSF, les vues sont représentées par des facelets.

3.4.1 Facelets

Comme JSP, Facelets est une technologie de présentation pour le développement d'applications web en Java.

Une page JSP est transformée en une Servlet qui possède un cycle de vie différent de celui de JSF, ce qui peut être source de confusion et de problèmes. A l'inverse, Facelets est spécifiquement développé pour JSF et est plus performant et léger [20].

Afin de créer une facelet le langage le plus utilisé est le XHTML qui est une version de la syntaxe HTML conforme au standard XML, donc une facelet est constituée de balises.

La réalisation des vues s'annonçait longue et fastidieuse vu le nombre important d'interfaces et de données à présenter aux utilisateurs en respectant l'ergonomie et l'interactivité, donc l'utilisation de jeux de composant d'interface utilisateur (UI) complémentaires a JSF nous ai paru comme une évidence, il existe de nombreux composants d'interface parmi eux il y a entre autres Myfaces, Bootsfaces, Richfaces et Primefaces.

3.4.2 Primefaces

Il fut sélectionné dans le cadre de la réalisation des interfaces utilisateurs de SMARTDIALYSE, car il permet de créer des interfaces plus conviviales grâce à sa grande fluidité et la panoplie des composants mis à disposition, ainsi qu'à technologie AJAX qu'elle utilise nativement. On parle alors d'interfaces riches.

Parmi les composants de JSF utilisés dans l'application :

- **Datatable** : est le composant le plus utilisé lors de la réalisation des interfaces, vu la quantité de paramètres à afficher pour chaque examen médical.

```
1. <p:dataTable id="patientTable" value="#{lazyPatientDatatable}" lazy="true" var="patient">
2.     <p:column sortBy="#{patient.nom}">
3.         <h:outputText value="#{patient.nom}"/>
4.     </p:column>
5. </p:dataTable>
```

Figure 19 Exemple dataTable affichage des patients

- **Commandbutton** : permet d'envoyer les informations saisies d'une manière globale ou partielle en utilisant AJAX.

- **Polling** : fait appel à une méthode du côté serveur périodiquement, dans l'une des vues réalisée le polling a été utilisé pour la sauvegarde automatique chaque vingt secondes.

```
1. |      <p:poll interval="20" listener="#{patientBean.addPatient()}" >  
2. |      </p:poll>
```

Figure 20 Exemple polling ajout d'un patient

Parmi les services que l'application doit fournir, le service d'impression des rapports d'examens. Pour cela nous utilisons l'outil JasperSoft studio anciennement appelé IReport afin de créer les rapports, et le framework JasperReports pour les générer sous format PDF et éventuellement, les enregistrer ou les imprimer.

3.4.3 JasperSoft studio

JasperSoft studio est un outil de design de type WYSIWYG (What You See Is What You Get). Il s'adresse aux développeurs et aux utilisateurs qui utilisent la librairie JasperReports pour créer des rapports.

Il est développé en Java et permet d'éditer de façon visuelle des rapports au format XML contenant des diagrammes, des images, des sous rapports, etc. Modifier les fichiers de type JRXML à la main n'est pas une opération facile, ce qui fait de JasperSoft studio un outil indispensable pour la mise en forme des rapports [21].

3.4.4 JasperReport

Le framework JasperReports est utilisé dans l'application afin de générer des rapports. Lorsque l'utilisateur consulte le dossier d'un patient, il a la possibilité de générer à la volée une fiche du dossier dans un format PDF.

JasperReports est un framework Open Source de reporting très populaire. Il permet de générer des fichiers aux formats PDF, XML, HTML, Excel, Word, OpenOffice (ODF) ou CSV à partir de données provenant de sources diverses : une base de données relationnelle (par exemple Oracle) de JavaBeans, de requêtes EJBQL...

Il peut être intégré dans une application Java EE pour fournir au développeur un moyen puissant afin de créer facilement des rapports personnalisés contenant des données de tout type : textes, images, diagrammes...

Le framework JasperReports se compose d'une librairie Java. La compilation du fichier de design (.Jrxml) est effectuée par la méthode `compileReport()` [21].

3.4.5 Lazy loading

Après avoir terminé les différentes interfaces de SMARTDIALYSE, nous avons entamé la phase des tests, pour cela nous avons peuplé la base de données et ce en injectant plus près d'un million de patient fictifs, ce qui a induit une lenteur sur le chargement des différentes vues. En sachant que cette situation est potentiellement la même que celle qui sera imposée à SMARTDIALYSE une fois déployé, donc nous devons impérativement trouver un moyen afin d'optimiser la présentation des données aux utilisateurs. Dans cette optique nous avons opté pour la solution proposée par Primefaces. Cette solution consiste en un mécanisme de chargement progressif (lazy loading).

Le lazy loading est caractérisé par deux intérêts, le premier est d'optimiser le temps de chargement d'une page web, En effet si nous n'utilisons pas le lazy loading et que nous tentons d'afficher des centaines de patients en un seul coup, le délai d'affichage sera plus long que si l'on affiche uniquement la partie visible d'une table.

Le deuxième intérêt ne s'adresse pas aux utilisateurs mais surtout aux développeurs et leur serveur. En implémentant le lazy loading, les requêtes générées sur le serveur seront allégées et qui dit moins de requêtes, dit un affichage plus rapide, C'est indéniablement un gain de ressources côté serveur et un gain de temps pour l'utilisateur.

4 Présentation de SMARTDIALYSE

La réalisation de SMARTDIALYSE s'est basée sur tous les outils et technologies précédemment définis. Cette application satisfait et englobe l'ensemble des exigences recueillies lors de nos différents entretiens avec les néphrologues. Par ailleurs nous avons réussi à faire de SMARTDIALYSE une application évolutive et adaptative aux besoins du personnel médical du service de néphrologie de Tlemcen. Dans cette section nous allons présenter SMARTDIALYSE à travers un scénario de navigation tout en expliquant quelques techniques d'optimisation que nous avons implémenté dans ce projet.

Avant d'effectuer toute opérations sur SMARTDIALYSE, une authentification s'impose, pour cela l'utilisateur a deux champs à remplir, le nom d'utilisateur et son mot de passe pour accéder au tableau de bord comme montré dans la « Figure 21 ».

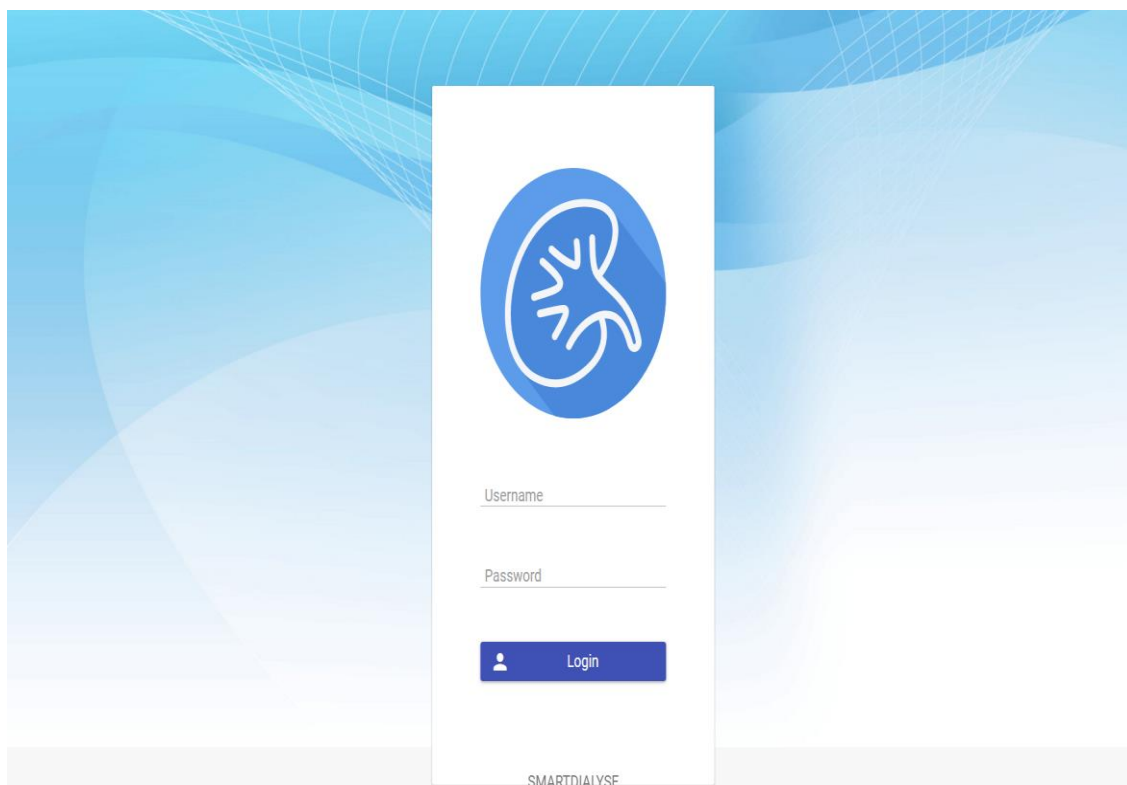


Figure 21 Page d'authentification "SMARTDIALYSE"

CHAPITRE III : Réalisation de SMARTDIALYSE

Une fois l'authentification réussie, SMARTDIALYSE dirige l'utilisateur vers une interface où il lui sera possible de naviguer librement. Il pourra entre autres effectuer des recherches sur les patients, mais aussi visualiser en détail les informations du patient en cliquant sur le bouton « plus d'infos » situé au pied du tableau, comme illustré sur la « Figure 22 ».

The screenshot displays the SMARTDIALYSE application interface. On the left is a sidebar with the user profile 'Lachachi charaf (Professeur)' and navigation links: Profile, Logout, Gestion des utilisateurs, Liste des patients, and Ajouter un patient. The main area is titled 'List des patients' and shows a table with 6 rows of patient data. The table has columns for Id, Nom, DateNaissance, Prenom, and Dossiers. Below the table is a 'Plus d'infos' button. The footer contains the SMARTDIALYSE logo and '© All Rights Reserved'.

Id	Nom	DateNaissance	Prenom	Dossiers
1	Lachachi	17-08-1992	charaf	
2	Ghernaout	02-07-1992	Malik	
4	Mahmoudi	15-05-1993	Karim	
5	Lachachi	11-09-1979	Djalal	
6	Kazi aoul	11-05-1993	salim	

Figure 22 page d'accueil "SMARTDIALYSE"

A l'arrivée d'un patient au sein du service, les médecins devront récupérer les informations personnelles relatives à ce dernier, SMARTDIALYSE donne cette possibilité aux utilisateurs. Dans ce cas l'utilisateur devra alors remplir le formulaire du patient.

CHAPITRE III : Réalisation de SMARTDIALYSE

Une fois le formulaire rempli, SMARTDIALYSE dispose d'une fonctionnalité qui relève les champs indispensables qui ne sont pas insérés et les affiche sous forme d'alertes. Une fois que toutes les informations personnelles du patient ont été insérées une alerte de succès sera alors observée.

The screenshot displays the SMARTDIALYSE application interface for adding a new patient. The top header is blue with the 'SMARTDIALYSE' logo and a back arrow. The left sidebar contains navigation links: 'Profile', 'Logout', 'Gestion des utilisateurs', 'Liste des patients', and 'Ajouter un patient' (highlighted). The main form area is divided into two columns for data entry. A success message box on the right states: 'Successful Your message: Le patient a bien été ajouté'.

Nom	Prénom
Ghernaout	Malik
Nom de jeune fille	Date de naissance
	02/07/1992
Adresse principale	Adresse secondaire
09 rue de la liberté	
Tel Fixe	Tel Mobile
043211626	0556077035
Email	Type assurance
malikghern@gmail.com	C.A.S.N.O.S
N° assurance	Nom ambulancier
FT6SGD108B	Taleb Aymen
Profession	Niveau d'étude
Etudiant	Master
Fraterie	Nom mère
0.00	Bouabdallah
Nom pere	statut Matrimonial
Amine	Célibataire
Nombre d'enfants	Nom contact 1
0.00	Ghernaout Nabil
Tel contact1	Nom contact 2
055425474	
Tel contact 2	Nom contact 3
Tel contact 3	

Add

Figure 23 Page d'ajout d'un patient "SMARTDIALYSE"

Une fois le patient ajouté, l'utilisateur est redirigé vers la page d'accueil comme donné dans la « Figure 21 », ou il pourra désormais consulter les dossiers et examens du patient. Et cela grâce au bouton « consulter », situé dans la colonne dossier de la table.

CHAPITRE III : Réalisation de SMARTDIALYSE

La figure ci-dessous représente le tableau de bord regroupant les différents examens et dossiers d'un patient.

The screenshot displays the SMARTDIALYSE interface. On the left, a sidebar lists various medical and administrative actions: Dossier personnel (selected), Dossier medical, Examen morphologique, Examen hemogramme, Examen biochimie, Examen phosphocalcique, Sérologie, Examen spécialisé, Calendrier vaccinal hépatite B, Transfusion, Traitement, Abord Vasculaire, Evènements, and Vistes Periodique. The main area shows a patient profile for Karim Mahmoudi (ID 4). The profile includes personal details like date of birth (15/05/93), address (Tlemcen), and contact information (mobile and email). It also lists medical history, including a Master's degree and a profession of Etudiant. At the bottom, there are buttons for 'Edit' and 'Print Dossier'.

id	Nom
4	Mahmoudi
Prénom	Karim
Date de naissance	15/05/93
Adresse principale	Tlemcen
Adresse secondaire	Algerie
Tel Mobile	0782896354
Type assurance	CASNOS
Nom ambulancier	Master
Niveau d'étude	0.00
Nom mere	Mahmoudi
statut Matrimonial	Célibataire
Nom contact 1	Mahmoudi amine
Nom contact 2	
Nom contact 3	

Nom de jeune fille	Adresse principale	Tel Fixe	Email	N° assurance	Profession	Fraterie	Nom pere	Nombre d'enfants	Tel contact1	Tel contact 2	Tel contact 3
	Tlemcen		karim@gmail.com	MLKTOI4568LK	Etudiant	0.00	Mahmoudi	0.00	0556987412		

[Edit](#) [Print Dossier](#)

Figure 24 Tableau de bord des examens et des dossiers d'un patient "SMARTDIALYSE"

A l'aide de ce tableau de bord, le médecin a la possibilité de :

- Modifier/Imprimer le « dossier personnel » comportant toute les informations renseignées lors de l'ajout du patient.
- Ajouter/Imprimer le « dossier médical » contenant les informations médicales relevées lors de l'admission d'un nouveau patient.

CHAPITRE III : Réalisation de SMARTDIALYSE

De plus l'utilisateur, aura la possibilité de gérer la série d'examens que propose SMARTDIALYSE. Nous allons montrer comment fonctionne la gestion des examens médicaux à travers quelques figures. L'interface de la visite periodique a été sélectionnée dans cette optique.

The screenshot shows the SMARTDIALYSE application interface. On the left, a sidebar displays the user profile 'MAHMOUDI KARIM' and a list of medical exams: Dossier personnel, Dossier medical, Examen morphologique, Examen hemogramme, Examen biochimie, Examen phosphocalcique, Sérologie, Examen spécialisé, Calendrier vaccinal hépatite B, Transfusion, Traitement, Abord Vasculaire, Evénements, and Visites Periodique (selected). The main area is titled 'Liste des visites periodiques hémodialyse' and shows a table with one entry. The table has columns: Résumé, Id, Diastole Entrée, Diastole Sortie, Systole Entrée, Systole Sortie, Poids Sec, Poids Inter Dialytique, and Date. The entry has Id 51, Diastole Entrée 1.2, Diastole Sortie 1.3, Systole Entrée 1.5, Systole Sortie 1.8, Poids Sec 80, Poids Inter Dialytique 75, and Date 2017-06-05. Below the table, there is a text description of hemodialysis and a list of actions: retirer du sang des substances en excédent, ajouter au plasma des substances qui manquent au sujet, and retirer l'eau qui s'est accumulée dans l'organisme. A calendar pop-up is visible, showing the date 1st of June 2017 selected.

Résumé	Id	Diastole Entrée	Diastole Sortie	Systole Entrée	Systole Sortie	Poids Sec	Poids Inter Dialytique	Date
	51	1.2	1.3	1.5	1.8	80	75	2017-06-05

Figure 25 Interface de gestion des visites periodiques

L'interface représentant cet examen a été confectionnée de façon à être la plus représentative et ergonomique possible, à cet effet un tableau représentant les différentes données de cet examen est réalisé, il est possible entre autres d'appliquer un filtre sur la date des examens effectués et ce en sélectionnant l'intervalle désiré à travers le calendrier visible sur la figure.

CHAPITRE III : Réalisation de SMARTDIALYSE

SMARTDIALYSE offre la possibilité de supprimer les examens que souhaite l'utilisateur, il est aussi possible d'ajouter un examen de visite periodique d'hémodialyse à travers l'interface suivante :

The screenshot shows the SMARTDIALYSE application interface. On the left is a sidebar with a user profile for 'GHERNAOUT (RESIDENT)' and a list of medical exams. The main area displays a 'Liste de' section. A modal dialog titled 'Ajouter/visite periodique hémodialyse' is open in the center. The dialog contains a table with three columns and two rows of data. Below the table is a text area with a rich text editor toolbar. The text area contains a definition of hemodialysis and a list of bullet points. At the bottom of the dialog are 'Add' and 'Cancel' buttons.

Date	Value 1	Value 2
05/06/17	1.20	1.30
1.50	1.80	80.00

75.00

L'hémodialyse est une méthode d'épuration du sang par la création d'un circuit de **circulation extra-corporelle** et son passage dans un **dialyseur**. Lorsqu'il existe une **insuffisance rénale sévère**, l'organisme se charge progressivement de substances qu'il convient d'éliminer par la dialyse.

- retirer du sang des substances en excédent
- ajouter au plasma des substances qui manquent au sujet
- retirer l'eau qui s'est accumulée dans l'organisme

Q Add Cancel

Figure 26 Interface d'ajout d'une visite periodique

CHAPITRE III : Réalisation de SMARTDIALYSE

Enfin après avoir ajouté/modifier l'examen de visite periodique l'utilisateur pourra alors imprimer le rapport relatif au patient souhaité.



PATIENT: Mahmoudi

Karim

RAPPORT VISITEHEMO

Id	Diastole entrée	Diastole sortie	Systole entrée	Systole Sortie	Poids sec	Poids intra dialytique	Résumé	Date
51	1.3	1.3	1.5	1.8	80	75	L'hémodialyse est une méthode d'épuration du sang par la création d'un circuit de circulation extra-corporelle et son passage dans un dialyseur . Lorsqu'il existe une insuffisance rénale sévère, l'organisme se charge progressivement de substances qu'il convient d'éliminer par la dialyse. retirer du sang des substances en excédent ajouter au plasma des substances qui manquent au sujet retirer l'eau qui s'est accumulée dans l'organisme	05/06/17 00:00
101	5.3	5.5	5.37	5.9	65	60	Dans laquelle l'eau est déplacée par le gradient de pression trans-membranaire , du côté où la pression est la plus forte vers le côté où la pression est la plus faible	08/06/17 00:00

Rapport du 01/06/17 00:00 au 10/06/17 00:00

juin 01, 2017

Figure 27 Exemple du rapport de la visite periodique

5 Conclusion

Ce chapitre nous a permis de définir et de justifier le choix des outils et technologies utilisés afin de réaliser notre application, car en utilisant l'architecture J2EE il devient très important de penser à la fiabilité et l'optimisation. De plus la présentation de SMARTDIALYSE offre aux lecteurs une vue d'ensemble sur cette application et la qualité de ses interfaces qui seront proposées aux utilisateurs.

CHAPITRE IV

Entrepôt de données

1 Introduction

Une fois la réalisation de notre application terminée, nous avons réfléchi aux perspectives d'enrichissement de cette dernière lorsqu'elle aura été exploitée, et qu'une grande quantité de données aura été injectée, et à la lumière des exigences que nous avons relevée, il nous est venu à l'idée de s'orienter vers les entrepôts de données afin de permettre aux utilisateurs de pouvoir exploiter au maximum ces données recueillies à des fins statistiques et décisionnels.

2 Business intelligence (BI)

Business Intelligence «BI »ou L'informatique décisionnelle se définit par les différents outils, moyens et méthodologies qui permettent la collecte de différentes sources de données brutes afin de les transformer en informations exploitable pour y appliquer des requêtes On en tire alors diverses vues, ou modes de visualisation, tels que des rapports et des tableaux de bord, pour mettre les résultats analytiques à la disposition des décideurs et des acteurs de l'entreprise. [22]

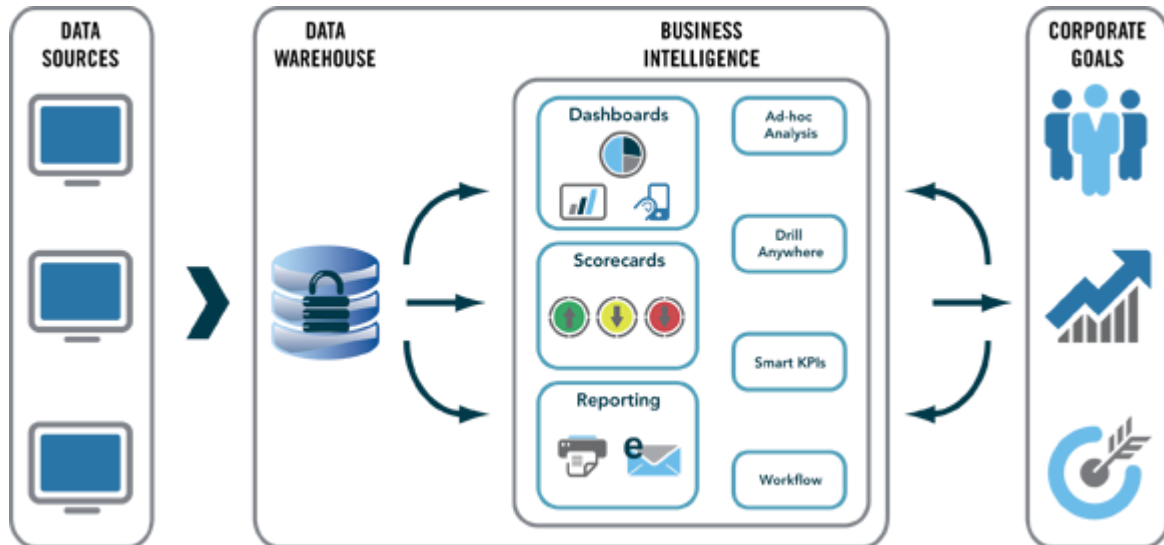


Figure 28 Entrepôt de données [23]

En vue de la quantité importante de données générées par SMARTDILAYSE, leur gestion et leur transformation en informations pouvant être mises en action peut devenir une tâche difficile. En combinant les données stockées dans l'entrepôt et l'informatique décisionnelle (BI), nous offrons la possibilité aux utilisateurs d'accéder instantanément aux informations pertinentes et d'appliquer des analyses puissantes pour une utilisation dans les prévisions et la surveillance des performances.

3 Processus de construction de l'entrepôt « SMARTDIALYSE »

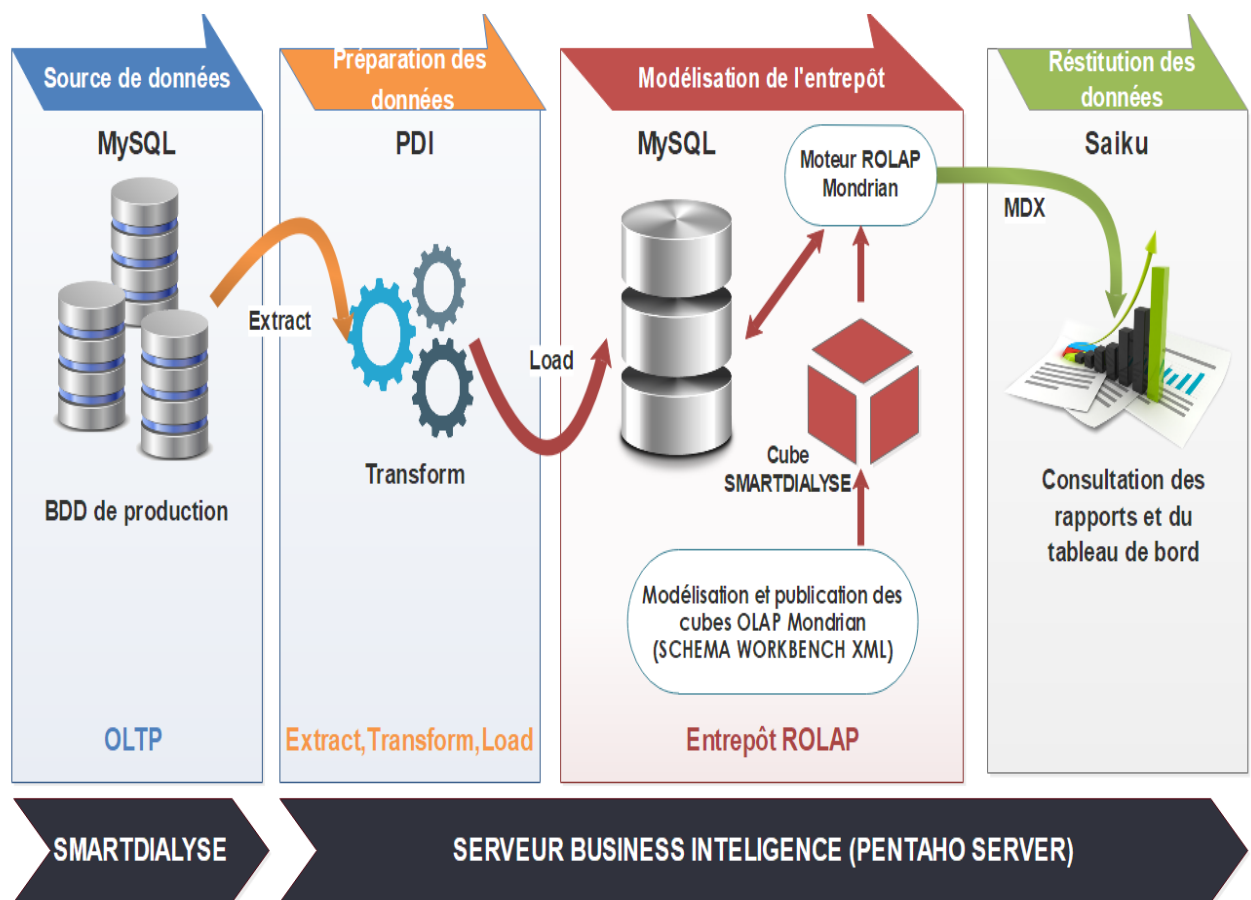


Figure 29 Architecture technique de l'entrepôt de données « SMARTDIALYSE »

La construction et réalisation de l'entrepôt se fait en respectant un enchaînement d'étapes importantes. (Voir figure 29)

Ces étapes sont :

- Modélisation de l'entrepôt.
- Préparation des données qui vont être extraites et transférées depuis la base de données de production vers l'entrepôt de données.
- Restitution des données : c'est dans cette étape que notre cube va être exploité pour générer des rapports via des requêtes MDX

Pour chacune de ces étapes un ensemble d'outils est offert. La modélisation et le fonctionnement de l'entrepôt dépendent fortement du choix technique du serveur multidimensionnel, car le fonctionnement est largement différent selon un serveur MOLAP, ROLAP, HOLAP ... etc. Dans notre cas, nous avons utilisé un serveur du type ROLAP (Mondrian). Dans ce cas, nous aurons à modéliser la base de données de l'entrepôt (Schéma en étoile) et le cube (Schéma en XML) qui à eux deux constituent notre entrepôt.

Dans ce qui suit nous allons présenter ces différentes étapes à commencer par la modélisation.

4 Conception et modélisation de l'entrepôt « SMARTDIALISE »

Dans le cas de notre projet, les entrepôts de données figurent parmi les perspectives de SMARTIDALYSE et par faute de temps, nous ne pouvions pas implémenter l'ensemble des entrepôts, pour cela que nous avons choisi l'examen de biochimie, celui qui comporte le plus d'informations susceptible d'être utilisées pour des statistiques et des prises de décision.

4.1 Définition de l'entrepôt de données

Un entrepôt est un ensemble de données historisées et non volatil variant dans le temps, organisé par sujets, consolidé dans une base de données unique, géré dans un environnement de stockage particulier, aidant à la prise de décision dans l'entreprise [24].

Trois fonctions essentielles :

- Collecte de données de bases existantes et chargement
- Gestion des données dans l'entrepôt
- Analyse de données pour la prise de décision

La construction d'un entrepôt de données est réalisée à partir de la modélisation multidimensionnelle. Les données de la modélisation multidimensionnelle sont organisées de manière à mettre en évidence le sujet de l'analyse (la table des faits) et les différentes perspectives de l'analyse (les tables des dimensions) qui symbolisent les différentes valeurs de l'activité analysée [24], cette modélisation est représentée par le schéma en étoile.

4.2 Schéma en étoile de la base de l'entrepôt

Ce model tire son nom de sa configuration, en effet il se forme d'un objet central appelé table des faits et est relié à un ensemble d'objets de manière radiale, les « tables de dimension».

La table des faits :

- Contient les différentes mesures
- Possède une clé étrangère pour chacune des tables de dimension.
- Est dénormalisées.

Les tables des dimensions :

- -contiennent les attributs définissant chacun des membres des niveaux hiérarchiques d'une dimension.
- Contiennent une clé primaire.
- Sont dénormalisées.

La figure ci-dessous représente la modélisation en étoile exposant le nombres examens biochimies effectuées par patient, date et lieu.

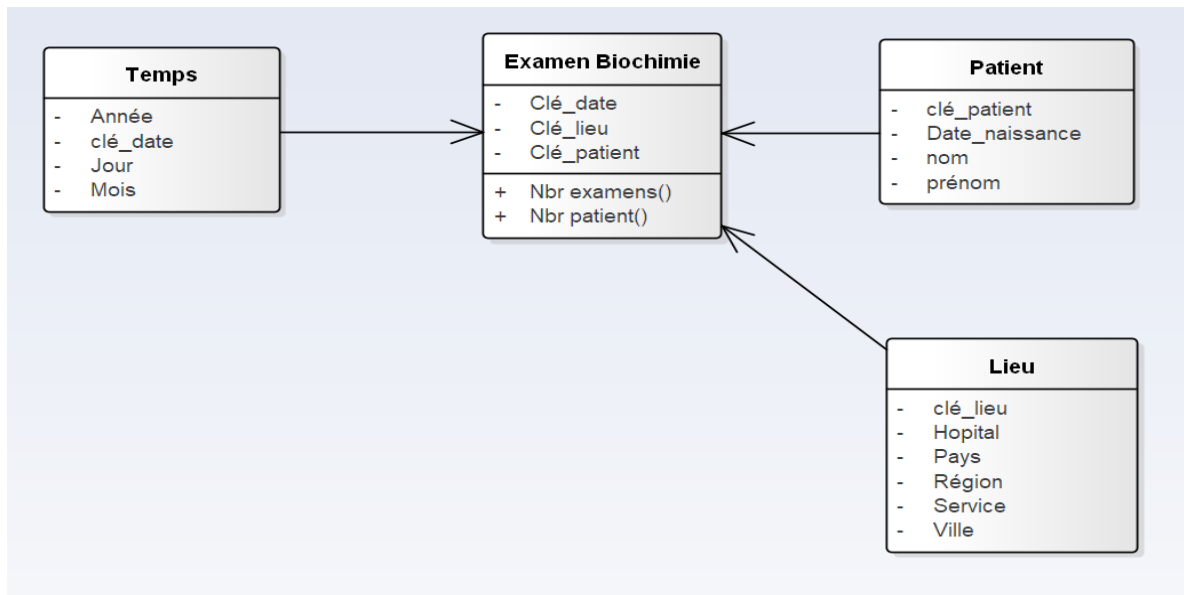


Figure 30 Schéma en étoile de l'examen biochimie

4.3 Modélisation du cube « SMARTDIALYSE »

La création d'un cube au sens Mondrian est tout simplement la rédaction d'un fichier XML. Ce fichier permet de lier les informations du cube que nous souhaitons faire apparaître et la base de données.

Pour faciliter la création de ce fichier, nous allons utiliser l'outil Schéma Workbench qui offre une interface graphique pour effectuer cette tâche. Après avoir configuré dans les « Préférences » la base de données sur laquelle nous souhaitons nous baser, nous pouvons commencer à créer notre cube.

CHAPITRE IV : Entrepôt de données

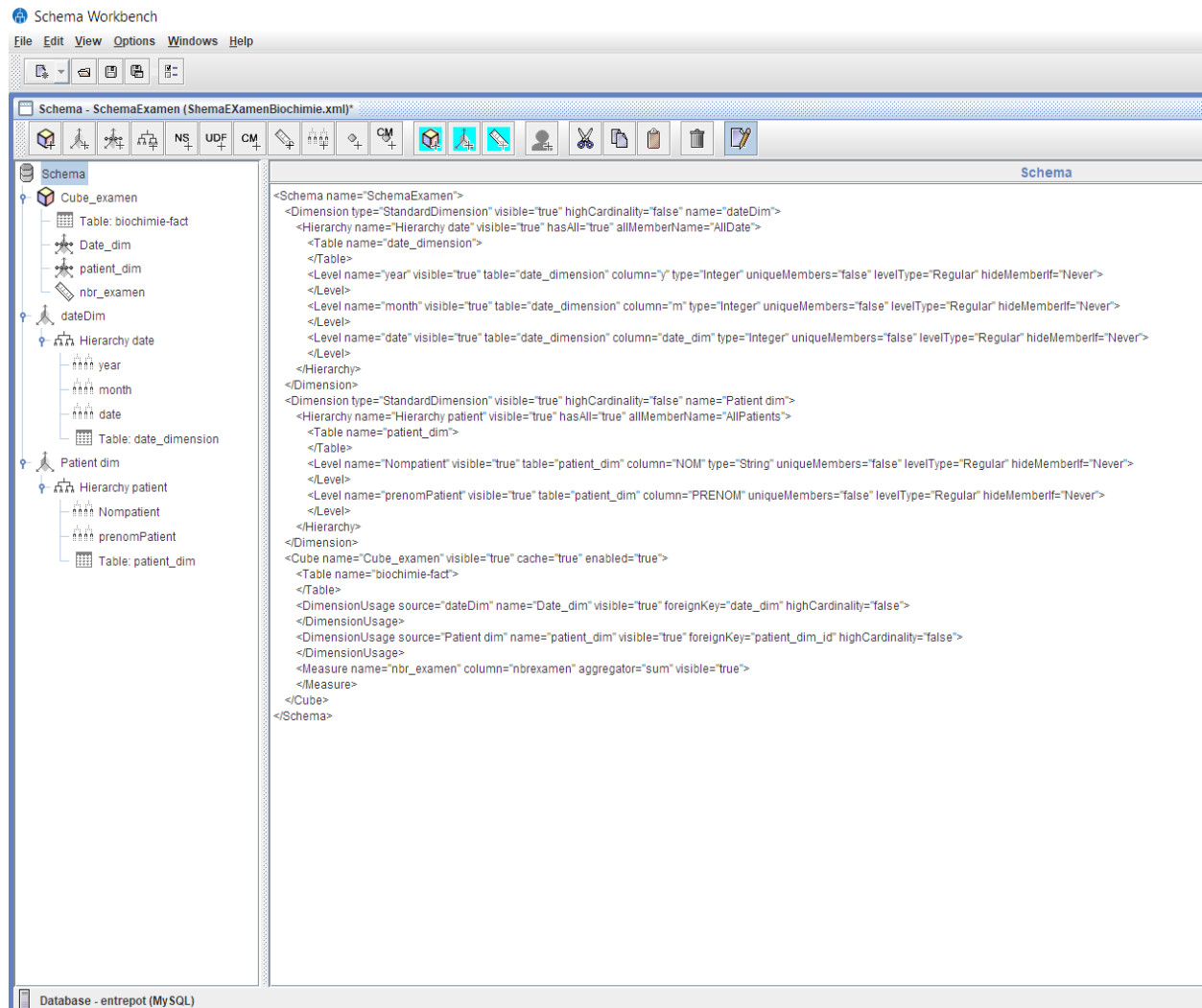


Figure 31 Schéma XML du Cube "SMARTDIALYSE"

La figure ci-dessus montre que pour la création de notre cube nous avons eu besoin :

- De la table de fait « examen biochimie »
- Des dimensions « Patient » « temps »
- De hiérarchies dimension et property pour chaque dimension
- De mesures dans notre cas « le nombre d'examen »

Une fois le cube créé nous procédons à sa publication dans le serveur Pentaho Mondrian en lui donnant l'URL de la console administration de Pentaho, le nom d'utilisateur et le mot de passe.

5 Préparation des données : Extraction, transformation et alimentation de la base de l'entrepôt

Nous procédons à la création et à l'alimentation des tables pour la base de données préalablement créée intitulée « Entrepôt » avec MysqlWorkbench via ETL (PentahoData Intégration (spoon)), qui est l'étape la plus critique et la plus importantes pour l'alimentation de notre entrepôt.

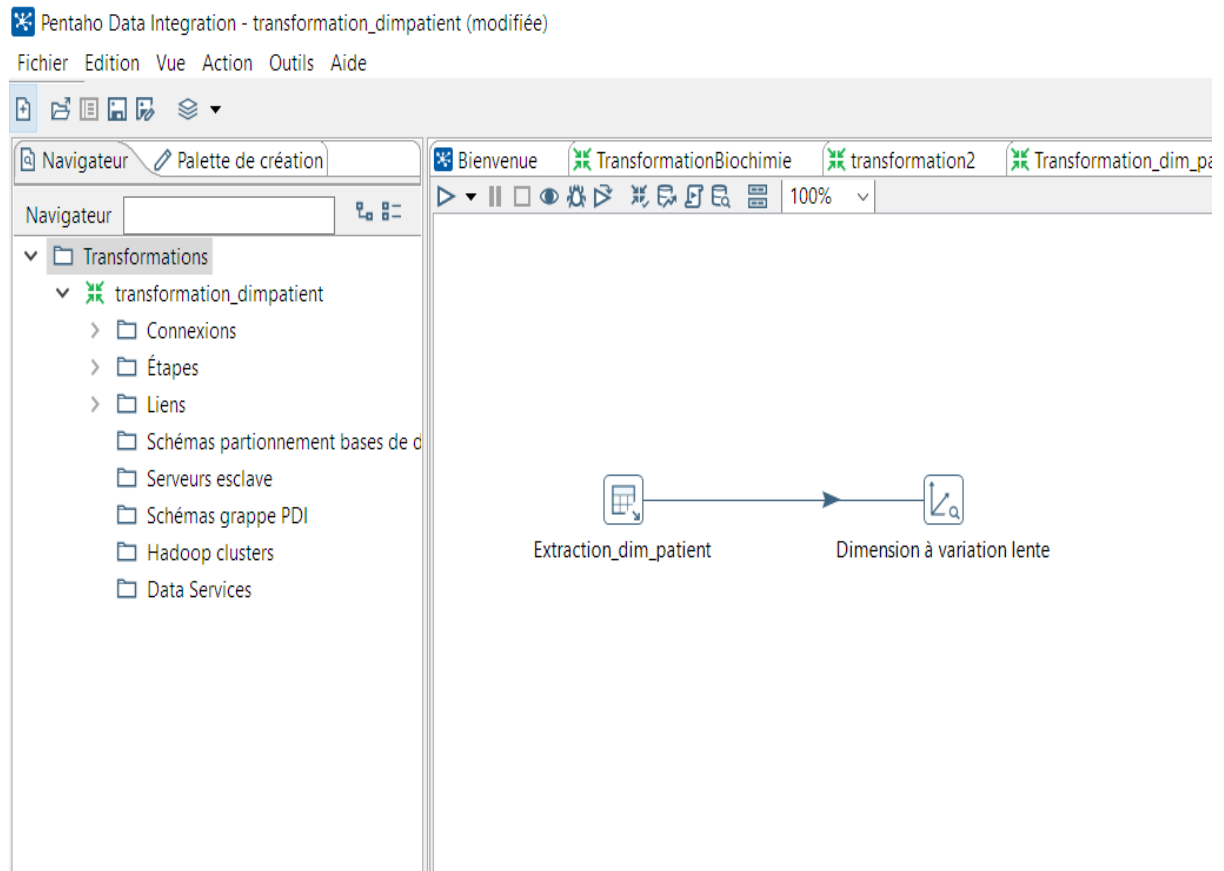


Figure 32 Extraction, transformation et chargement de la dimension patient

6 Restitution des données : exploitation du cube

Après avoir créé et publié le schéma xml du cube, sur le server BI de Pentaho, nous pouvons commencer à exploiter le cube.

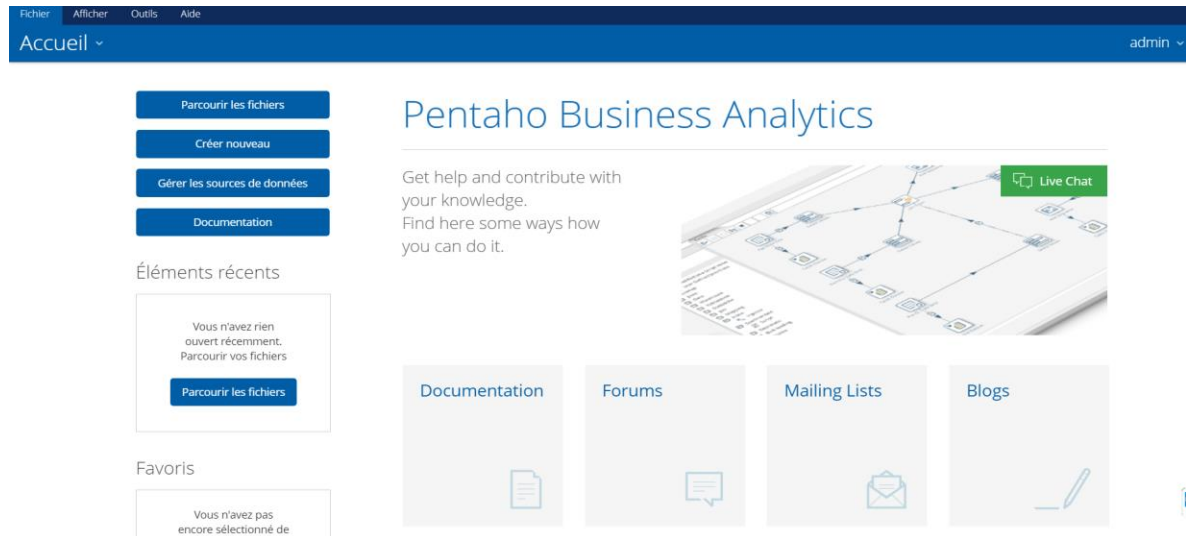


Figure 33 Interface principale de Pentaho

❖ Saiku

Afin d'obtenir une information d'une manière simple pour un utilisateur non informaticien à partir de notre cube nous avons utilisé l'outil graphique « Saiku », en effet il permet de transcrire des actions graphiques en requête MDX.

Voici un exemple d'une requête mdx qui a été appliqué sur notre cube, nous allons ressortir le nombre d'examen par date pour les différents patients.

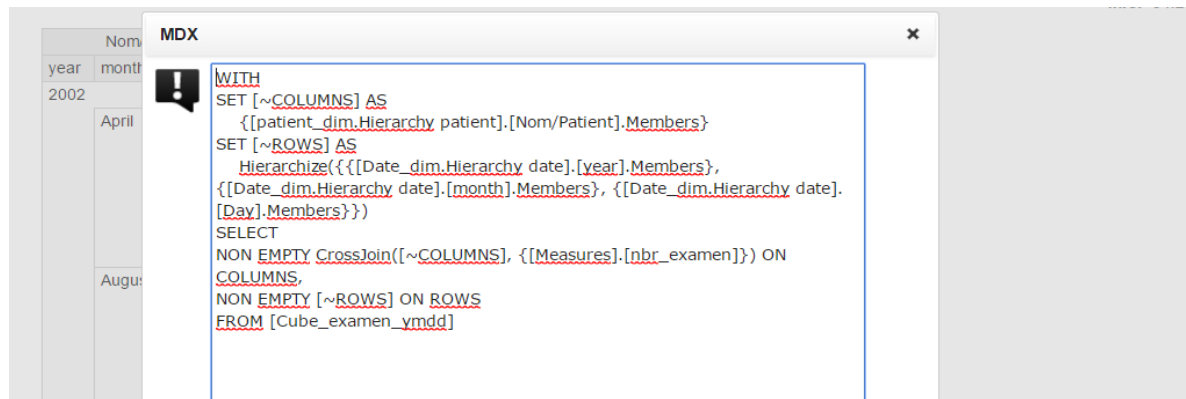


Figure 34 Exemple de requête MDX appliquée sur le cube

CHAPITRE IV : Entrepôt de données

L'outil « Saiku » présente le cube sous forme de tableau croisé. Il permet d'insérer en abscisse ou en ordonné les différentes dimensions afin d'étudier les mesures du cube

La figure suivante affiche le résultat de la requête MDX cité précédemment.

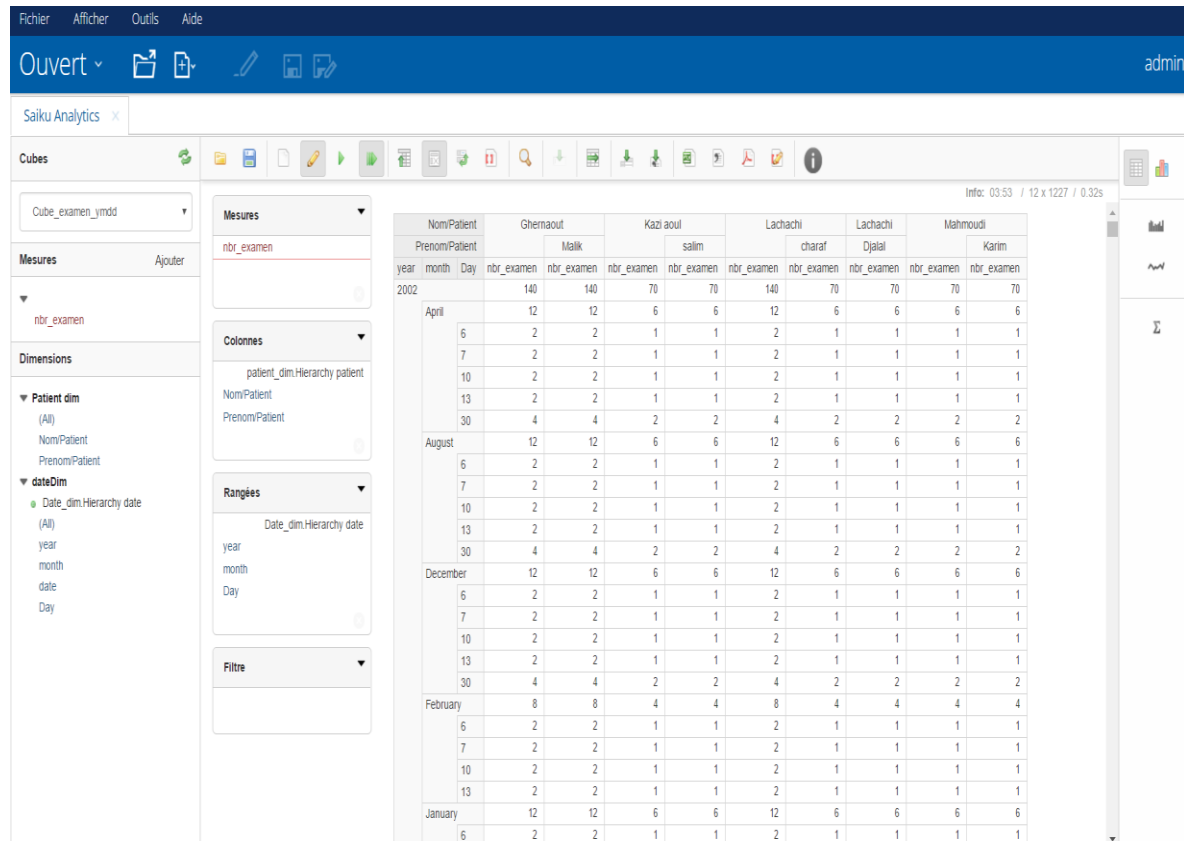


Figure 35 Résultat sous forme de tableau de la requête MDX

Pour même la requêtes MDX une représentation statistique des données est illustrés par les deux figures suivantes :

CHAPITRE IV : Entrepôt de données

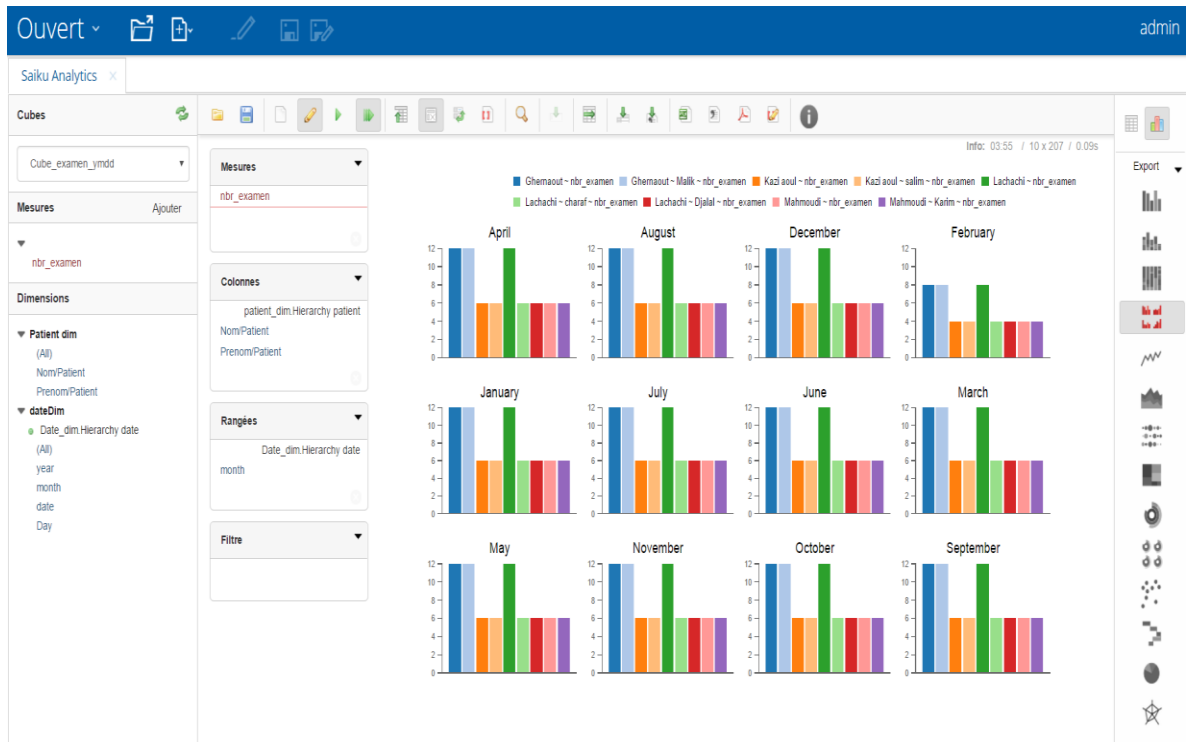


Figure 36 Résultat sous forme de histogrammes de la requête MDX

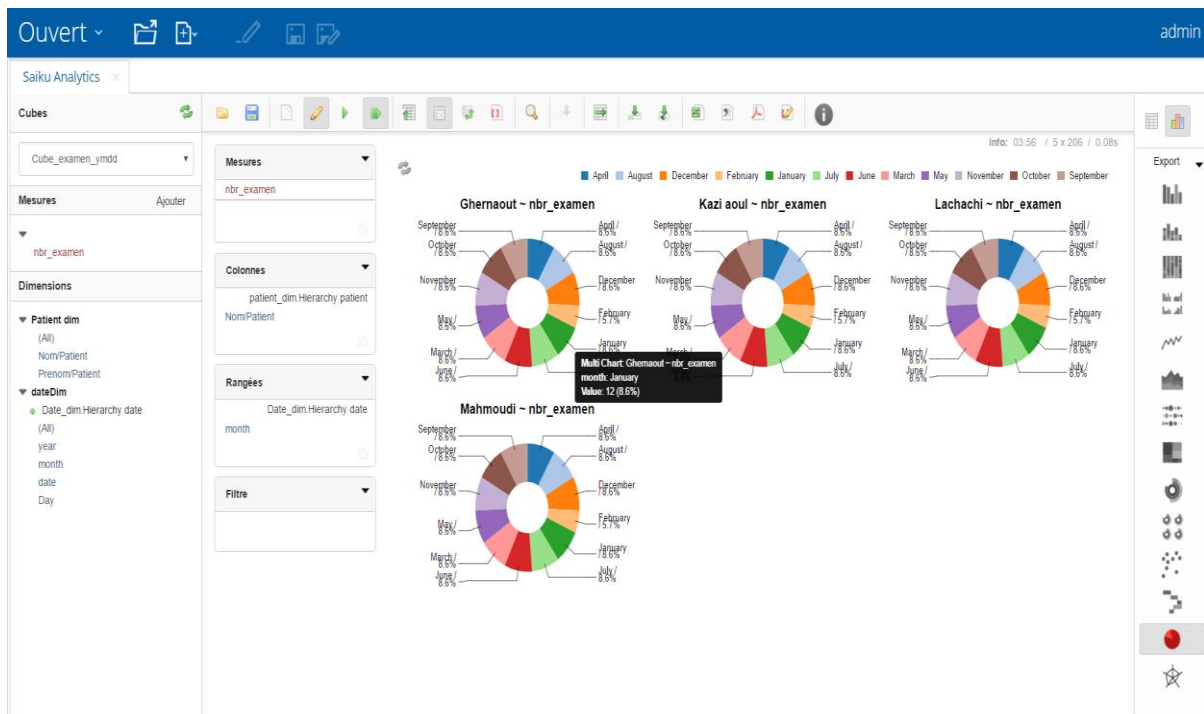


Figure 37 Résultat sous forme de donuts de la requête MDX

7 Conclusion

Dans ce chapitre, nous avons réalisé une partie du projet BI qui a suivi le déploiement de SMARTDIALYSE, cela grâce à l'implémentation du datamart de l'examen de biochimie, avec ce magasin de données nous avons pu avoir des résultats statistiques, pouvant être utilisées pour des prises de décision

Conclusion Générale

Concevoir et réaliser une plateforme web pour le service de néphrologie du C.H.U de Tlemcen s'est fixé comme objectif principale de notre projet de fin d'étude.

Notre travail s'est basé sur le développement d'un système dépendant des technologies J2EE, ceci nous a amené à découvrir une nouvelle plateforme de développement et à enrichir notre savoir et notre expérience.

Le système que nous avons conçu permet d'avoir toutes les informations relatives à un patient dialysé (informations personnels, dossier médicale, examen biochimie, examen spécialisé, visite periodique, ...) accessibles sur la même plateforme.

Aussi un entrepôt de données a été réalisé afin de faciliter la prise de décisions, en prenant en considération les statistiques générées, et tout cela grâce aux données recueillies à partir de Smartdialyse.

Les objectifs fixés pour notre projet de fin d'études ont été atteint à un degré très avancé, d'un côté nous avons présenté une plateforme bénéfique aussi bien pour les médecins que pour les patients. D'un autre côté, ce projet nous a permis de se familiariser avec la programmation web en Java et d'approfondir nos connaissances dans le domaine des systèmes basés sur l'architecture trois tiers.

Au niveau des perspectives de notre travail, nous envisageons d'implémenter SMARTDIALYSE à l'échelle nationale, afin de créer son réseau interne, ce qui facilitera l'échange des dossiers des patients entre services des différentes wilayas, mais aussi facilitera le quotidien du patient et lui permettra d'aller faire ses examens n'importe où dans le territoire national sans se soucier de son dossier médical et personnelle. Notre deuxième perspective majeure est d'étendre l'entrepôt de données afin d'englober tous les examens pour des statistique pointues et des prises de décisions plus fiable.

La réalisation de SMARTDIALYSE a été une expérience unique pour nous autant qu'étudiants impliquées dans le développement de notre chère patrie.

Bibliographie

- [1] «Historique,» CHU de Tlemcen, [En ligne]. Available: <http://www.chu-tlemcen.dz/index.php?id=41&r=Historique>.
- [2] T. Bayard, «Anatomie et rôle des reins,» [En ligne]. Available: <http://www.nephrologie-lyon.com/anatomie-et-role-des-reins.html>.
- [3] R. Marianne Belleza, «Urinary System Anatomy and Physiology,» 09 05 2017. [En ligne]. Available: <https://nurseslabs.com/urinary-system/>. [Accès le 15 05 2017].
- [4] «Physiologie Des Systèmes Intégrés Les Principes Et Fonctions,» [En ligne]. Available: <http://www.lightinghomes.net/id/1515781/physiologie-des-systemes-integres-les-principes-et-fonctions.html>. [Accès le 15 05 2017].
- [5] P.horde, «Insuffisance rénale,» [En ligne]. Available: <http://sante-medecine.journaldesfemmes.com/faq/8807-insuffisance-renale-aigue-et-chronique-symptomes-et-traitement>. [Accès le 15 05 2017].
- [6] [En ligne]. Available: <http://tpehemodialyse.e-monsite.com/pages/presentation/hemodialyse.html>. [Accès le 15 04 2017].
- [7] P.-Y. Martin, «La dialyse péritonéale,» [En ligne]. Available: <http://www.hugge.ch/nephrologie/dialyse-peritoneale-0>. [Accès le 02 04 2017].
- [8] «La dialyse,» [En ligne]. Available: <https://www.clstjean.be/la-dialyse>. [Accès le 15 04 2017].
- [9] «Les reins,» [En ligne]. Available: <http://organes-artificiels.e-monsite.com/pages/les-reins.html>. [Accès le 15 04 2017].

- [10] A. O. T.-M. Benoît Charroux, UML2 pratique de la modélisation, 2nd éd., Paris: Pearson education France, 2009, p. 270.
- [11] P. Roques, UML 2 par la pratique, Paris: Eyrolles, 2008.
- [12] S. Durocher, «www.site.uottawa.ca:4321/~laganier/seg3510/patterns/strategy.ppt,» [En ligne]. [Accès le 2017 04 27].
- [13] Mathieu.G, «<http://design-patterns.fr/introduction-aux-design-patterns>,» 09 05 2013. [En ligne]. [Accès le 27 04 2017].
- [14] J.-M. DOUDOUX, «<http://jmdoudoux.developpez.com/cours/developpons/java/chap-design-patterns.php>,» 19 03 2016. [En ligne]. [Accès le 2017 04 27].
- [15] «commentcamarche.net,» [En ligne]. Available: <http://www.commentcamarche.net/contents/221-reseaux-architecture-client-serveur-a-3-niveaux>. [Accès le 22 04 2017].
- [16] A. BOLE, «Le Serveur d'Application JBoss & les Entreprise JavaBean,» [En ligne]. Available: http://www-igm.univ-mlv.fr/~dr/XPOSE2003/alexandrebole/ejb_3.html. [Accès le 24 04 2017].
- [17] J.-M. DOUDOUX., «Java server faces,» [En ligne]. Available: <https://www.jmdoudoux.fr/java/dej/chap-jsf.htm>. [Accès le 2017 04 28].
- [18] la Free Libre Open Source Lebanese Movement, «<http://cours.cofares.net/programmation/introduction-a-jsf-j2ee/qu-est-ce-jsf>,» [En ligne]. [Accès le 28 04 2017].
- [19] Wikipedia, «<https://fr.wikipedia.org/wiki/MySQL>,» [En ligne]. [Accès le 28 04 2017].

- [20] J. Moussa, «JSF et Facelets,» [En ligne]. Available: <http://djomos.developpez.com/tutoriels/java/jsf/facelets-intro/>.
- [21] C. Fernandes, Les EJB3 (avec struts 2, JSF2, JasperReports 3, Flex 3), 2010.
- [22] M. rouse, «Lemagit BI informatique decisionnelle,» [En ligne]. Available: <http://www.lemagit.fr/definition/BI-informatique-decisionnelle>. [Accès le 20 04 2017].
- [23] «MOVATION BUSINESS INTELLIGENCE (BI),» [En ligne]. Available: <https://motivitysolutions.com/business-intelligence/>.
- [24] G.gardarin, «DATA WAREHOUSE - DATA MINING,» [En ligne]. Available: orap.irisa.fr/ArchivesForums/Forum7/gardarin.ppt.

Liste des figures

Figure 1 Anatomie macroscopique du système urinaire [3]	8
Figure 2 Vue générale du rein [4]	9
Figure 3 : Méthodes de dialyse [6]	10
Figure 4 : Dialyse péritonéale [8]	11
Figure 5 Le Processus d'hémodialyse [9]	12
Figure 6 Digramme de cas d'utilisations.....	18
Figure 7 Séquence Authentification	22
Figure 8 Séquence ajout d'un patient	23
Figure 9 séquence recherche.....	24
Figure 10 Diagramme de classes SMARTDIALYSE	25
Figure 11 Design pattern strategy adapté à SMARTDIALYSE	27
Figure 12 Design pattern façade à SMARTDIALYSE	28
Figure 13 Architecture J2EE globale.....	30
Figure 14 Architecture J2EE avec JSF et EJB.....	32
Figure 15 Exemple d'implémentation des EJB dans SMARTDIALYSE	34
Figure 16 Conteneur web avec JSF	36
Figure 17 Bout de code ajoutpatient.xhtml.....	37
Figure 18 Bout de code patientBean.java	37
Figure 19 Exemple dataTable affichage des patients	39
Figure 20 Exemple polling ajout d'un patient	40
Figure 21 Page d'authentification "SMARTDIALYSE"	42
Figure 22 page d'accueil "SMARTDIALYSE"	43
Figure 23 Page d'ajout d'un patient "SMARTDIALYSE"	44
Figure 24 Tableau de bord des examens et des dossiers d'un patient "SMARTDIALYSE"	45

Figure 25 Interface de gestion des visites periodiques	46
Figure 26 Interface d'ajout d'une visite periodique.....	47
Figure 27 Exemple du rapport de la visite periodique.....	48
Figure 28 Entrepôt de données [23].....	50
Figure 29 Architecture technique de l'entrepôt de données « SMARTDIALYSE"	51
Figure 30 Schéma en étoile de l'examen biochimie	54
Figure 31 Schéma XML du Cube "SMARTDIALYSE"	55
Figure 32 Extraction, transformation et chargement de la dimension patient	56
Figure 33 Interface principale de Pentaho	57
Figure 34 Exemple de requête MDX appliquée sur le cube	57
Figure 35 Résultat sous forme de tableau de la requête MDX	58
Figure 36 Résultat sous forme de histogrammes de la requête MDX	59
Figure 37 Résultat sous forme de donuts de la requête MDX	59

Liste des abréviations

A

AJAX, 43, 44

Asynchronous Javascript and Xml

B

BI, 6, 55, 56, 62, 64, 68

Business Intelligence

C

CHU, 7

Centre Hospitalier Universitaire

CSV, 45

Comma Separated Variable

E

EJB, 31, 36, 37, 38, 39, 40, 41, 69

Enterprise JavaBeans

ETL, 61

Extract Transform Load

H

HOLAP, 57

Hybrid Online Analytical Processing

HTML, 39, 43, 45

HyperText Markup Language

HTTP, 40

HyperText Transfert Protocol

J

J2EE, 5, 34, 35, 36, 37, 39, 53, 69

Java Platform, Enterprise Edition

JSF, 5, 36, 38, 39, 40, 41, 43, 68, 69

JavaServer Faces

M

MDX, 57, 62, 63, 64, 70

Multidimensional Expressions

MOLAP, 57

Multidimensional Online Analytical Processing

MVC, 38, 39

Model-View-Controller

O

ODF, 45

Open Document Format

R

ROLAP, 57

Relational Online Analytical Processing

SGBD, 35, 42, 43

SQL, 42

UML, 19

URL, 60

WYSIWYG, 44

XML, 43, 45, 57, 59, 60, 70

S

Système de Gestion de Base de Données

Structured Query Language

U

Unified Modeling Language

Uniform Resource Locator

W

What You See Is What You Get

X

Extensible Markup Language

Résumé

A travers ce projet, nous voulons améliorer la gestion des patients dialysé au sein du service de néphrologie du CHU de Tlemcen, en mettant en place une plateforme informatique (SMARTDIALYSE) qui vise à faciliter la tâche du corps médical, dans leur mission de garantir le bon fonctionnement du service tout en ayant une coordination avec les patients. En plus de SMARTDIALYSE, la réalisation de l'entrepôt de données, facilitera la prise de décision, et donnera une vision plus claire sur la situation des patients, en donnant la possibilité aux corps médical de visualiser et comparer les données recueillies pour chaque patient.

Mots clés : JSF, J2EE, EJB, MVC, DATAWAREHOUSE, BI

Abstract

The goal of this project is to improve the management of the dialysis patients within the nephrology department of the hospital of Tlemcen, by setting up a computerized platform, called SMARTDIALYSE. This platform aims to facilitate the task of medical crew, in their mission to guarantee the good functioning of the service while having coordination with the patients. In addition to SMARTDIALYSE, the realization of the data warehouse will facilitate decision-making and provide a clearer view of the patients' situation, giving medical staff the opportunity to view and compare the data collected for each patient.

Keywords: JSF, J2EE, EJB, MVC, DATAWAREHOUSE, BI

ملخص

عملنا من خلال هذا المشروع على تحسين إدارة مرضى غسيل الكلى في قسم أمراض الكلى في المستشفى الجامعي لتلمسان من خلال إنشاء نظام آلي والذي يهدف إلى تسهيل مأمورية العاملين في المجال الطبي، في مهمتهم لضمان حسن سير العمل في الخدمة من خلال وجود تنسيق مع المرضى. وإلى جانب SMARTDIALYSE، فإن تحقيق مستودع البيانات لتسهيل اتخاذ القرارات وإعطاء صورة أوضح عن حالة المرضى، مما يجعل من الممكن للطاقم الطبي عرض ومقارنة البيانات التي يتم جمعها لكل مريض.

الكلمات المفتاحية : JSF, J2EE, EJB, MVC, DATAWAREHOUSE, BI