

Mémoire de Fin d'Etudes

Pour l'Obtention du Diplôme de Master en Informatique

Présenté par :

**FORTAS OUSSAMA ILYES
BENACHENHOU LINDA**

Domaine : Mathématiques & Informatique

Spécialité : Aide à la Décision et Systèmes Intelligents

Session1 2021

THEME

**REALISATION D'UNE APPLICATION INTELLIGENTE D'AIDE A LA
DECISION POUR LA PRISE EN CHARGE DES VICTIMES
D'ACCIDENTS.**

Encadré par : Mr. ATMANI Baghdad

Co-encadré par : Mr. BENFRIHA Hichem

Jury

Président: Mr. BOUAMRANE Karim

Examineur: Mr. ABDI Mustapha Kamel

CodeMaster : 5/ADSI/2021

ABSTRACT

This work consists of realizing an intelligent application of decision support precisely in resuscitation for the medical care of children victims of road accidents based on case based reasoning.

Keywords: medical decision support system, case based reasoning, JColibri

Résumé

Ce travail consiste à réaliser une application intelligente d'aide à la décision médicale précisément en réanimation pour la prise en charge des enfants victimes d'accident de la route basée sur le raisonnement à partir de cas (RàPC) ou Case Based Reasoning (CBR) en anglais.

Mots clés : système d'aide à la décision médicale (SADM), raisonnement à partir de cas (RàPC), Jcolibri.

REMERCIEMENTS

Nous tenons à exprimer notre profonde gratitude et nos sincères remerciements à notre encadreur monsieur Atmani et à notre Co-encadreur monsieur Benfriha qui nous ont fait l'honneur de diriger ce travail et leurs conseils furent d'un apport considérable.

Nos remerciements vont également aux enseignants qui nous ont assistés durant tout le cursus universitaire.

Que les membres du jury trouvent ici nos remerciements les plus vifs pour avoir accepté de juger notre travail.

Nos sincères sentiments vont à tous ceux qui, de près ou de loin, ont contribué à la réalisation de ce projet. En particulier nos chères familles et nos amis (es) pour leurs sacrifices, leurs prières et leurs soutiens.

DÉDICACES

Benachenhou Linda

Je voudrais dédier mon travail à ma famille et à mes amis.

Un sentiment particulier de gratitude à mes chers parents qui n'ont jamais cessé de donner leurs encouragements, leur amour et leur soutien.

Je dédie également mon travail à mes deux meilleures amies Leila et Mani qui m'ont soutenues tout au long du processus et à mes tantes qui ont toujours été là pour moi,

Dernière dédicace à mon partenaire et meilleur ami Fortas Oussama Ilyes qui m'a toujours soutenue et encouragée et qui sans lui je n'aurais pas pu réaliser ce travail

Je vous remercie tous et vous promets de ne pas vous décevoir.

DÉDICACES

Fortas Oussama Ilyes

Je voudrais dédier ce travail à mon père, cet effort est consacré à toi, j'aurais souhaité qu'une telle application existe et t'aurai sauvé la vie...

Je le dédie en second lieu à ma mère, qui est beaucoup plus qu'une simple mère, mais un refuge et une idole de perfection et de soutien dans chaque événement de vie. Je vis pour chérir ton nom Mama.

Je remercie grandement ma famille et mes amis proches qui m'ont toujours soutenu. Je remercie également mon ami Zaoui, et mon frère Abdeldjalil qui ont investi leur temps et leurs efforts précieux pour m'aider à construire cette belle œuvre.

Je dédie également ce projet à ma meilleure amie Benachenhou Linda avec qui j'ai eu la chance de partager ce projet, avec qui j'ai partagé tous les défis et qu'avec l'aide du Dieu nous avons vaincus.

Liste des figures

Figure 1 : Cycle de raisonnement à partir de cas.....	9
Figure 2 : L'architecture générale.	17
Figure 3 : Architecture de JColibri [21].....	18
Figure 4 : Architecture de Spring Boot.....	22
Figure 5 : Diagramme de séquence.....	23
Figure 6 : La procédure configure().....	25
Figure 7 : La procédure preCycle()	25
Figure 8 : La procédure Cycle()	26
Figure 9 : La fonction de similarité globale.....	26
Figure 10 : L'exécution de recherche et la sélection de meilleurs cas	27
Figure 11 : L'endpoint /sendTrauma	28
Figure 12 : L'endpoint /retainAcase	28
Figure 13 : La requête d'insertion	29
Figure 14 : L'affichage de la base de cas importée	30
Figure 15 : L'affichage de similarités.....	31
Figure 16 : L'affichage de K meilleurs cas	31
Figure 17 : Affichage du message d'insertion avec son identifiant	31
Figure 18 : Interface d'accueil.....	32
Figure 19 : Interface information sur le patient.....	32
Figure 20 : Interface information sur la respiration	33
Figure 21 : Interface information sur l'hémodynamique.....	33
Figure 22 : Interface bilan des lésions	34
Figure 23 : Interface Solution.....	35
Figure 24 : Interface cas similaires	35
Figure 25 : Interface solutions des cas similaires.....	36
Figure 26 : Interface pour retenir la solution.....	36
Figure 27 : Interface version mobile.....	37
Figure 28 : Interface solution version mobile.....	38

Liste des tableaux

Tableau 1 : Travaux connexes du RAPC dans le domaine médical.....	12
Tableau 2 : Protocole ABCD.....	14
Tableau 3 : La partie problème dans la base de cas.....	16
Tableau 4 : La partie solution dans la base de cas	17
Tableau 5 : Association des mesures de similarité aux descripteurs.....	21

Table des matières

ABSTRACT

Résumé

REMERCIEMENTS

DÉDICACES

DÉDICACES

Liste des figures

Liste des tableaux

Introduction générale	1
Chapitre 1 : Aide à la décision médicale	3
1. Introduction	3
2. Notion d'aide à la décision	3
3. Notion de décision médicale	3
4. Qu'est qu'un système d'aide à la décision médicale ?	4
5. La typologie des systèmes d'aide à la décision médicale	4
5.1. Systèmes d'assistance documentaire	4
5.2. Systèmes de rappels ou d'alertes médicales	4
5.3. Les systèmes consultants	4
6. Les systèmes experts dans l'aide à la décision médicale	5
7. Conclusion	5
Chapitre 2 : Le raisonnement à partir de cas	6
1. Introduction	6
2. Le raisonnement à partir de cas (RàPC)	6
3. Pourquoi le RàPC a eu de succès	6
4. Notion de base	7
4.1. Le cas	7
4.2. La base de cas	7
4.3. Les mesures de similarités	7
4.3.1. La similarité locale	7
4.3.2. La similarité globale	8
5. Le cycle du RàPC	8
5.1. L'élaboration	9
5.2. La recherche	9
5.3. L'adaptation	9
5.4. La révision	10

5.5. La mémorisation	10
6. Domaines d'application	10
7. Travaux Connexes	11
8. Conclusion	12
Chapitre 3 : Conception et Implémentation	13
1. Introduction	13
2. Problématique	13
3. Le protocole ABCD	13
4. Description de la base de cas	14
4.1. La partie problème	14
4.2. La partie solution	16
5. L'approche proposée	17
6. L'architecture générale	17
7. Les outils	18
7.1. JColibri	18
7.1.1. Définition :	18
7.1.2. L'architecture de JColibri	18
• La phase de recherche	18
Les mesure de similarité locales	18
La mesure de similarité globale	21
• La phase d'adaptation	21
• La phase de révision	21
• La phase d'apprentissage	21
7.2. Spring boot	21
7.2.1. Définition	21
7.2.2. L'architecture de SpringBoot	22
8. Web service REST API	22
8.1. Coté client :	22
8.2. Coté serveur :	22
9. Diagramme de séquence	23
10. Les captures	24
10.1. BACKEND	24
10.2. FRONTEND	32
11. Conclusion	38
Conclusion générale	39
Références	40

Introduction générale

De nos jours, l'intelligence artificielle est au cœur de tous les domaines, elle vise à réaliser des machines capables de simuler l'intelligence humaine (raisonnement, apprentissage... etc.).

Le raisonnement à partir de cas (en anglais Case Based Reasoning CBR), fait partie du raisonnement par analogie, et est une forme d'intelligence artificielle qui permet à un système de trouver une solution à un problème P sur la base de plusieurs problèmes P' qui lui sont similaires et qui ont déjà une solution.

On retrouve souvent le raisonnement à partir des cas dans le domaine médical, un domaine idéal pour concevoir un système d'aide à la décision car pour diagnostiquer une maladie, les médecins utilisent leurs connaissances théoriques et aussi leur expérience acquise.

Parfois, le manque d'expérience peut mener à une perte de vie, car le bon diagnostic et le bon traitement sont importants pour la survie d'une personne, mais le temps de prise de la bonne décision l'est encore plus.

Notre projet s'inscrit dans le cadre de la mise en place d'un Système d'Aide à la Décision Médicale (SADM) dans le service de réanimation pour la prise en charge des enfants victimes d'accident de la route qui ont un traumatisme crânien. Ce service reçoit des enfants parfois dans des cas critiques qui demandent la prise en charge immédiate.

Ce projet vise à apporter une aide précieuse aux médecins lorsque les données ne sont pas assez suffisantes et aussi aux nouveaux médecins qui n'ont pas beaucoup d'expérience.

Dès lors qu'un patient est en ambulance, ses informations seront transmises au service de réanimation. Le médecin pourra les consulter et lui proposer le traitement le mieux adapté grâce au système d'aide à la décision médicale, ce qui va permettre d'accroître les chances de survie du patient et aussi de diminuer les douleurs et les séquelles.

En réalisant ce projet nous pourrions améliorer la qualité du diagnostic clinique cela améliorera la qualité des soins et nous offrira également un meilleur rapport coût-efficacité des tests et des thérapies.

Organisation du mémoire :

Ce mémoire est structuré en trois chapitres

Au premier chapitre nous introduisons les systèmes d'aide à la décision, et la typologie des systèmes d'aide à la décision médicale. Au second chapitre nous exposerons les principes de base du raisonnement à partir de cas ainsi que son cycle de fonctionnement. Le troisième chapitre est consacré à l'approche qu'on va proposer pour la conception de l'application, les outils utilisés tels que SpringBoot et JColibri qui facilitent le développement d'applications dans le cadre du raisonnement à partir de cas et l'implémentation de l'application. Enfin une conclusion générale où nous proposerons quelques perspectives pour clôturer ce mémoire.

Chapitre 1 : Aide à la décision médicale

1. Introduction

Dans ce premier chapitre, nous présenterons les systèmes d'aide à la décision dans le domaine médical. De nos jours, avec l'avancement du domaine médical et l'avancement des technologies et techniques, nous nous sommes retrouvés dans le besoin de l'informatique pour faire progresser et améliorer nos services médicaux. Par expérience, on constate de très bons résultats, un meilleur suivi de l'histoire et bien sûr une charge moins lourde pour les médecins praticiens.

Pour mieux comprendre ce domaine, nous expliquerons dans un premier temps ce qu'est la notion d'aide à la décision. Nous nous concentrons sur notre domaine de recherche qui est le domaine médical, Nous définirons ce qu'est exactement une décision médicale, Ensuite, nous expliquerons ce qu'est un système d'aide à la décision dans le domaine médical.

2. Notion d'aide à la décision

Pour introduire cette notion, nous utiliserons la définition proposée par Roy :

“ L’activité de celui qui, prenant appui sur des modèles clairement explicités mais non nécessairement complètement formalisés, aide à obtenir des éléments de réponses aux questions que se pose un intervenant dans un processus de décision ”[1]

Il s'agit donc d'une aide qui permet au décideur d'élaborer une représentation pertinente de la situation, tout en sachant que toute application sur les connaissances et les données à un but d'aide à la décision.

L'aide à la décision est un ensemble de tests techniques qui permettent aux individus d'effectuer la meilleure prise de décision possible. Il est utilisé dans de nombreux domaines tels que les finances, la politique, la gestion de crise et dans le domaine médical.

L'utilisateur bénéficie d'outils qui lui permettent de vérifier et d'analyser les informations afin de pouvoir prendre une décision appropriée à un moment donné, sans avoir besoin d'avoir une grande connaissance de l'informatique, certains types de ces outils incluent des méthodes qui font la possibilité de décider ou de choisir parmi plusieurs solutions.

3. Notion de décision médicale

Dans le domaine médical, la décision est définie par l'acte médical, cet acte s'inscrit dans le cadre d'un diagnostic, d'une planification thérapeutique ou d'une prescription. Le processus de la décision médicale consiste à avoir un diagnostic du problème et à proposer un traitement ou à en changer un. Il existe de

nombreuses applications de la prise de décision dans ce domaine, ces applications ont pour but d'aider le personnel à prendre ses décisions, cela nécessite l'utilisation de divers outils de prise de décision.

Au cours de notre vie de routine, nous rencontrons souvent des problèmes déjà vus et probablement déjà résolus auparavant, dans ce cas, les raisons basées sur des raisons revêtent une importance précieuse dans le domaine de la résolution de problèmes et aussi dans la planification des tâches, lors de leur pratique, les médecins utilisent les connaissances qu'ils possédaient déjà plus l'expérience acquise, non seulement les théories mais aussi les techniques, dans ce cas, il serait utile d'utiliser le Raisonnement à partir de cas du domaine de la décision médicale.

4. Qu'est qu'un système d'aide à la décision médicale ?

Un système d'aide à la décision clinique est un ensemble organisé d'informations et d'outils destinés à aider le personnel médical dans son raisonnement afin d'identifier un diagnostic et de choisir une thérapie adéquate, cela se fait par l'interaction entre l'homme et la machine.

Il s'agit donc essentiellement de logiciels qui aident à offrir aux praticiens des informations utiles aux moments et aux situations nécessaires en donnant une description de la situation et la connaissance de cette situation, traités et filtrés afin d'améliorer la qualité des soins de santé du patient.

5. La typologie des systèmes d'aide à la décision médicale

5.1. Systèmes d'assistance documentaire

Ces systèmes sont faits pour faciliter l'accès aux informations d'importance en peu de temps, ce type de systèmes n'a pas vraiment de méthode de raisonnement. Par exemple les systèmes qui permettent l'accès aux résultats d'analyses d'un laboratoire, les dossiers médicaux des patients, les références médicales et les systèmes qui considèrent les médicaments qui offrent une aide indirecte à la décision.

5.2. Systèmes de rappels ou d'alertes médicales

Ces systèmes surveillent le médecin et lui rappellent de ne pas commettre d'erreurs ou de considérer certains éléments pour sa décision, ils sont plus actifs et plus directement impliqués dans la décision médicale.

Ce type de système aide à la prise de décision en surveillant des données médicales spécifiques et en surveillant afin de réduire le risque d'erreurs médicales[2].

5.3. Les systèmes consultants

Ce type de systèmes utilise l'intelligence artificielle pour offrir des conseils sur les interventions et les diagnostics basés sur les données du patient, ils appliquent un raisonnement aigu qui construit une conception intellectuelle.

Ce type de système se présente sous la forme d'un système informatique qui intègre une base de connaissances médicales et les données des patients par le biais d'un moteur d'inférence qui applique un raisonnement prédéfini.

6. Les systèmes experts dans l'aide à la décision médicale

Les systèmes experts sont des systèmes à base de connaissances visant à simuler le raisonnement des experts engagés au cours d'un processus de prise de décision.

Un système expert se compose de trois modules principaux :

- La base de connaissance est construite à partir de l'expertise d'un spécialiste.

Elle se compose d'une base de règles et d'une base de faits.

- La base de règles : contient les connaissances expertes (règles de l'expert) qui sont représentées généralement par des règles de production s'écrivant, sous la forme : SI (condition) ALORS (action).
- La base de faits est l'ensemble des propositions connues du système à un moment donné. C'est la mémoire de travail du système expert.
- Le moteur d'inférence est un programme qui utilise les règles définies dans la base de connaissances pour résoudre un problème particulier décrit par des faits.
- L'interface pour l'aide à l'acquisition des connaissances fournies par l'expert peut être plus ou moins sophistiquée, l'accent étant souvent mis sur une syntaxe des règles le plus proche possible du langage naturel.

7. Conclusion

Dans ce chapitre nous avons présenté quelques types de système d'aide à la décision médicale dont les systèmes à base de connaissance.

Les systèmes d'aide à la décision médicale se définissent comme étant un processus décisif dont le but est d'apporter aux patients la meilleure qualité de soins.

Le raisonnement à partir de cas s'appuie sur l'acquisition de connaissance pour résoudre de nouveau problème, il est compatible avec les applications auxquelles leur tâche est accomplie par les experts humains comme dans le domaine médical et dont les expériences sont enregistrées dans une base de données ou dans des documents.

Le raisonnement à partir de cas semble être une bonne technique pour mettre en œuvre des systèmes d'aide à la décision et pour la résolution des problèmes de diagnostic.

Chapitre 2 : Le raisonnement à partir de cas

1. Introduction

Le paradigme du raisonnement à partir des cas est une sorte d'intelligence artificielle qui consiste à résoudre de nouveaux problèmes en utilisant des problèmes précédemment résolus. Cela peut être aussi simple que de réappliquer la solution précédente, ou cela peut être aussi difficile que d'appliquer une nouvelle solution.

Dans ce chapitre nous présenterons le raisonnement à partir de cas, quelques notions de base, et son cycle de fonctionnement.

2. Le raisonnement à partir de cas (RàPC)

RàPC est dérivé des travaux de Minsky et Abelson[3].

Janet L Kolodner, auteur de *"An Introduction to case based reasoning"*[4] a introduit le terme "raisonnement à partir de cas" pour la première fois. Son travail est une synthèse des œuvres de l'époque, dans lesquelles elle s'appuie principalement sur les travaux de Minsky et Schank.

Le raisonnement à partir du cas est un moyen de réflexion qui reproduit la méthode de la réflexion humaine, ce sont des cas qui ont été historiquement résolus pour extraire une solution pour de nouveaux problèmes.

Cette réutilisation peut être une simple application de la solution précédente ou bien une application "adaptée" pour avoir une meilleure solution au nouveau problème.

3. Pourquoi le RàPC a eu de succès

Dans les années 1990, L'IA a connu un grand succès industriel, en grande partie grâce à des systèmes experts. Mais malheureusement à cause des difficultés à modéliser les connaissances et à maintenir les systèmes, cet enthousiasme s'est éteint.

Ce type de système nécessite une modélisation complète du système avec lequel on veut raisonner, alors que tout modéliser n'est pas possible.

D'où vient le raisonnement par cas, en sauvegardant la résolution de problèmes au fur et à mesure que nous les rencontrons, nous constaterons qu'au départ le système est moins efficace mais il gagne plus en compétence dans le processus de mise en œuvre, il peut être adapté à de nombreux domaines avec beaucoup moins de difficulté.

4. Notion de base

4.1. Le cas

Le RàPC fonctionne sur le principe des expériences qui sont appelées CAS.

Généralement le cas est donné par une paire de problèmes, ou le problème est la partie du cas constituée d'un ensemble de descripteurs, et la Solution est la partie solution du cas qui correspond à la solution du problème.

On distingue deux types de cas : Le cas cible et le cas source.

Le cas cible est le problème qui doit être résolu, la solution n'a pas encore été trouvée.

Le cas source désigne une expérience déjà passée par la résolution de problèmes.

4.2. La base de cas

Les cas sont stockés dans une base de cas pour trouver la meilleure correspondance avec le problème

4.3. Les mesures de similarités

La similarité entre deux cas est mesurée en fonction de la distance entre les valeurs des mêmes attributs. Cette distance est fréquemment estimée par des mesures euclidiennes et de Hamming. La similarité globale entre deux cas est généralement évaluée par une somme pondérée de la similarité locale de chacun des attributs. Étant donné que tous les attributs d'un cas ne sont pas d'égale importance et que cette importance varie d'une situation à l'autre, un poids est attribué à chaque descripteur dans chaque cas. Ces poids permettent de pondérer la similarité globale entre deux cas en accordant un « vote » plus important aux attributs les plus méritants.

4.3.1. La similarité locale

Le calcul de la similarité locale dépend du type de descripteur et est basé sur la distance.

- Pour les valeurs de descripteurs symboliques (mono-valeurs)[5]

$$sim(a, b) = \begin{cases} 1 & \text{pour } a = b \\ 0 & \text{pour } a \neq b \end{cases}$$

- Pour les valeurs de descripteurs symboliques (multi-valeurs) :

$$sim(a, b) = \frac{card(a) \cap card(b)}{card(a \cup b)}$$

- Pour les valeurs de descripteurs numériques :

$$sim(a, b) = 1 - \frac{|a - b|}{range}$$

Tels que :

a et b sont les descripteurs.

$card$ est la cardinalité de l'ensemble.

$range$ est la valeur absolue de la différence entre la borne supérieure et la borne inférieure de l'ensemble des valeurs.

4.3.2. La similarité globale

Elles sont calculées au niveau des cas ou des objets en agrégeant les similarités locales. Plusieurs similarités globales sont utilisées dans les systèmes de RàPC et aucune d'elles n'est universelle et restent dépendantes du domaine concerné. [6]

- Weighted Block-City

$$sim(A, B) = \sum_{i=1}^n w_i sim_i(a_i, b_i)$$

Tels que

n est le nombre de descripteurs,

w_i est le poids du descripteur i

sim_i est la similarité locale calculée pour le descripteur i

5. Le cycle du RàPC

En 1994, Aamodt et Plaza ont publié un article dans lequel ils proposaient un cycle de raisonnement de cas composé de quatre étapes : la recherche, l'adaptation, la révision et la mémorisation (en anglais retrieve, reuse, revise and retain)[7].

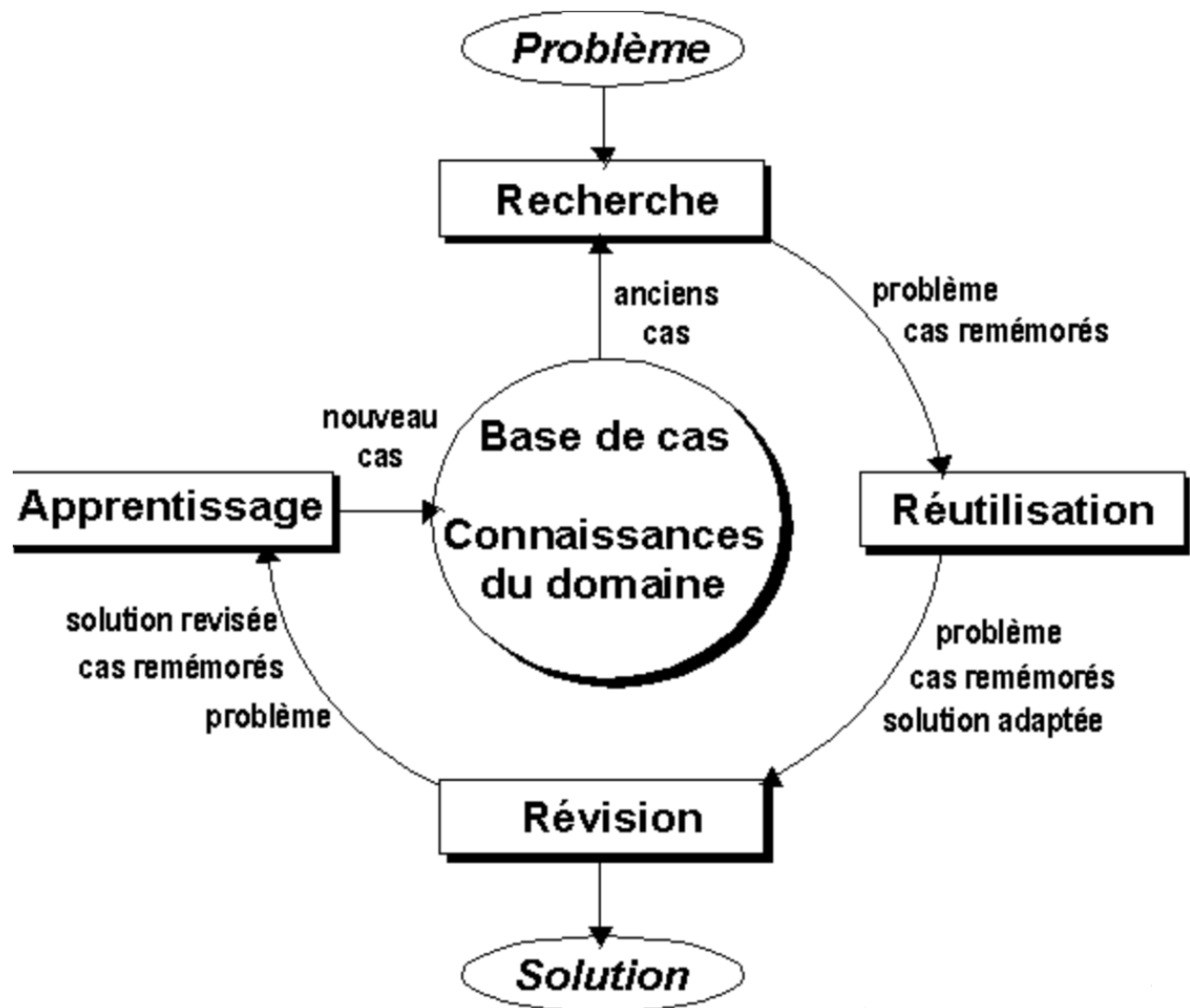


Figure 1 : Cycle de raisonnement à partir de cas

5.1. L'élaboration

Cette étape consiste à passer du problème vaguement décrit dans le monde réel à un problème bien structuré pour être exploitable par le système informatique.

5.2. La recherche

Elle consiste à la recherche dans la base de cas le ou les cas sources les plus proches à partir de la description du cas cible. Parmi les algorithmes de recherche on retrouve l'algorithme de K plus proche voisin qui est le plus habituellement utilisé. Il calcule la similarité entre un cas cible (nouveau problème) et un cas source de la base de cas. K est le nombre de cas sources voisins considérés comme étant proches (autour) du cas cible.

5.3. L'adaptation

Lorsque le meilleur cas sélectionné dans la base de cas ne correspond pas parfaitement avec le nouveau cas, l'ancienne solution doit être adaptée pour s'adapter plus précisément au nouveau problème.

Il existe deux types d'adaptation, l'adaptation structurelle et l'adaptation dérivationnelle.

Dans les méthodes d'adaptation structurelle, le processus d'adaptation est directement appliqué aux solutions stockées dans la base de cas. Les méthodes d'adaptation structurelle peuvent être divisées en trois techniques principales : les méthodes de substitution, les méthodes de transformation et les heuristiques d'adaptation ou méthodes d'adaptation basées sur la critique.

Contrairement aux méthodes d'adaptation structurelle, les méthodes d'adaptation dérivées n'opèrent pas sur les solutions, mais sur la méthode qui a été utilisée pour dériver cette solution. L'objectif est de ré exécuter la même méthode appliquée pour dériver l'ancienne solution, pour recalculer la solution pour le nouveau cas.

5.4. La révision

La révision est l'une des étapes les plus importantes pour un raisonnement à partir de cas. Cela donne au système un moyen d'évaluer ses décisions dans le monde réel, lui permettant de recevoir des retours qui lui permettent d'apprendre du succès ou de l'échec. La révision peut être définie comme le processus d'évaluation de la qualité de la performance de la solution proposée pour le nouveau cas. Le processus d'évaluation peut mettre en évidence la nécessité d'une adaptation supplémentaire généralement appelée réparation de la solution proposée.

Cette étape peut être effectuée en demandant à un humain (l'expert) si la solution est bonne ou non, en simulant les effets de la solution proposée dans le monde réel.

5.5. La mémorisation

Cette étape consiste à stocker dans la base de cas ce qui est utile à retenir pour l'enrichir et permet d'acquérir de nouvelles connaissances qui vont être utilisées pour résoudre des problèmes ultérieurs et augmenter l'expérience du système.

L'apprentissage se fait à partir du succès et aussi de l'échec dans la résolution du problème cible.

6. Domaines d'application

- **Recommandations**

Les systèmes de raisonnement à partir de cas pour la recommandation de produits, ou systèmes de recommandation, ont pour but de suggérer un ensemble de produits à un utilisateur à partir d'une évaluation de ses préférences.

- Planification
CHEF est un système de planification à partir de cas dans le domaine de la cuisine du *Sichuan*. CHEF a pour but de proposer des recettes qui satisfont un ensemble de contraintes données par l'utilisateur[8].
- Médecine
- Automobile

7. Travaux Connexes

Auteur(s)	Sujet	Techniques Utilisés	Année
Bareiss, Porter & Wier[9]	Troubles auditifs	RAPC	1889
Gierl & Stengel-Rutkowski [10]	Syndromes dysmorphiques	RAPC	1994.
Macura & Macura[11]	Images de radiologie	RAPC	1995.
Galobardes et al[12].	Cancer du sein	RAPC	2002.
Ahmed et al[13].	Stress	RAPC et la logique floue	2011
Banerjee & Chowdhury[14]	Anomalies rétinienues	RAPC, Clustering flou et arbres de décision	2015
Chakraborty et al[15].	Choléra	RAPC	2015
Saraiva et al[16].	Quatre types de cancer Gastro-intestinal	RAPC et raisonnement à base de règles	2015
Khussainova, Petrovic, & Jagannathan[17]	Radiothérapie (cancer du cerveau)	RAPC et clustering	2013
Tyagi & Singh[18]	Asthme	RAPC	2015
Yin et al[19]	Mal de crâne	RAPC	2015
Sharaf-El-Deen[20]	Cancer du sein et thyroïde	RAPC et raisonnement à base de règles	2014

8. Conclusion

Le but de ce chapitre est de définir le raisonnement à partir de cas. Plusieurs recherches ont été faites dans ce domaine et plusieurs projets basés sur le raisonnement à partir de cas ont vu le jour.

Dans ce projet, on s'intéresse à l'utilisation du raisonnement à partir de cas dans les systèmes d'aide à la décision médicale ce qui fera l'objet du chapitre suivant.

Chapitre 3 : Conception et Implémentation

1. Introduction

Dans ce chapitre nous présenterons l'approche qu'on a proposée pour notre cas d'étude qui est la prise en charge de l'enfant victime d'un accident de la route, l'architecture générale de notre application et les outils utilisés pour mettre en œuvre cette application.

2. Problématique

Quelle application convient le mieux pour que le médecin soit assisté dans le processus de prise d'une décision médicale au sein du service de réanimation ? Pour répondre à cette problématique nous avons proposé d'appliquer le raisonnement à partir de cas.

3. Le protocole ABCD

La rapidité de la prise en charge de l'enfant victime d'un accident de la route est très importante afin de traiter les traumatismes et augmenter la chance de survie et aussi afin de minimiser les douleurs physiques et psychologiques. Les médecins suivent un protocole de prise en charge qui concerne quatre types d'examen (liberté des voies aériennes, ventilation, hémodynamique et neurologique). Pour chaque examen il y a des mesures à prendre tels que le score de Glasgow, la fréquence respiratoire, la pression artérielle ...etc. et des gestes à effectuer comme l'intubation, la ventilation, l'O₂ thérapie...etc. que nous détaillerons plus bas.

Le tableau ci-dessous récapitule le protocole ABCD.

	Signification	Examen	Geste
A	Airway (liberté des voies Aériennes)	Spo2	O2 thérapie,
B	Breathing : ventilation	Age, sexe, FR, Spo2, cyanose, conscient, Glasgow, râles, encombrement bronchique	O2 thérapie, ventilation,
C	Circulation : Hémodynamique	Age, sexe, PAS, PAD, FC, TRC, T°, Marbrure, T° extrémité, diurèse	SSI, Support-hemo

D	Disability (neurologique)	Glasgow, reflexe photo, conscient, convulsion, vomissement, motrice spon, tonus, mouvement oculaire, fracture, œdème, hémorragie, corps étranger	Sédation, intubation, PIC, osmothérapie
---	------------------------------	--	--

Tableau 2 : Protocole ABCD

4. Description de la base de cas

Dans ce présent paragraphe pour chaque protocole, nous allons décrire plus en détail chaque descripteur.

4.1. La partie problème

Contient tous les descripteurs du patient.

Le descripteur	La signification	Le type
Âge	L'âge de l'enfant en mois	Entier
Poids	Le poids de l'enfant	Double
Sexe	Le sexe qui peut être soit masculin soit féminin	Chaine de caractère
Température	La température de son corps	Double
PAS	La pression artérielle systolique : est la pression maximale, au moment de la « contraction » du cœur	Chaine de caractère
PAD	La pression artérielle diastolique : est la pression minimale, au moment du « relâchement » du cœur	Chaine de caractère
FC	Fréquence cardiaque	Chaine de caractère
T° extrémité	La température des extrémités qui peut prendre la valeur : froide, chaude, normale	Chaine de caractère
Diurèse	Désigner le volume de la sécrétion urinaire (on peut parler aussi de "débit urinaire")	Chaine de caractère
Marbrure	Les marbrures sont des traces cutanées causées notamment	Chaine de caractère

	par une hypoperfusion sanguine	
TRC	Temps de recoloration cutanée pour estimer la pression artérielle	Chaine de caractère
SPO2	Désigne saturation pulsée en O2	Chaine de caractère
FR	Fréquence respiratoire	Chaine de caractère
Cyanose	Est une coloration anormale bleutée de la peau, due à l'oxygénation insuffisante du sang	Chaine de caractère
Râle	Bruit respiratoire anormal entendu par le médecin à l'auscultation des poumons.	Chaine de caractère
Encombrement bronchique	Accumulation de sécrétions dans les bronches	Chaine de caractère
Glasgow	C'est une échelle allant de 3 (coma profond) à 15 (personne parfaitement consciente), et qui s'évalue sur trois critères : Ouverture des yeux ; Réponse verbale ; Réponse motrice.	Chaine de caractère
Tonus	Tonus musculaire Légère contraction permanente du muscle vivant	Chaine de caractère
Conscient	Indique si l'enfant est conscient ou non	Chaine de caractère
Convulsion	La <u>convulsion</u> est la contraction (spasme) violente et involontaire d'un ou plusieurs muscles	Chaine de caractère
Vomissement	Indique si l'enfant a des vomissements ou non	Chaine de caractère
Motrice spontanée	Ensemble des fonctions nerveuses et musculaires permettant les mouvements volontaires ou automatiques du corps.	Chaine de caractère
Mouvement oculaire	Les mouvements oculaires sont les rotations que les globes oculaires (les yeux) effectuent autour de leurs	Chaine de caractère

	centres, et qui modifient la direction du regard.	
Reflexe photo	Le réflexe photomoteur ou réflexe pupillaire est la constriction physiologique de la pupille exposée à la lumière.	Chaine de caractère
Fracture	Une fracture se traduit par une cassure ou une fêlure de l'os	Chaine de caractère
Œdème	L'œdème est un gonflement des tissus mous dû à une augmentation du liquide interstitiel	Chaine de caractère
Corps étranger	Un corps étranger est un objet de taille supérieure au millimètre et qui a pénétré le corps humain	Chaine de caractère
Hémorragie	Une hémorragie correspond à une perte importante de sang, un saignement qui ne s'arrête pas	Chaine de caractère

Tableau 3 : La partie problème dans la base de cas

4.2. La partie solution

Elle correspond à la thérapie qu'il faut suivre et la classe.

Le descripteur	La signification	Le type
O2 thérapie	Masque /sonde nasale/rien	Chaine de caractère
SSI	Sérum salé isotonique	Chaine de caractère
Support hymo		Chaine de caractère
Sédation	Action de calmer, apaisement par un sédatif.	Chaine de caractère
Intubation	Elle consiste à introduire un dispositif tubulaire dans la trachée	Chaine de caractère
Ventilation	La ventilation artificielle consiste à suppléer ou assister la respiration spontanée à l'aide d'un respirateur artificiel	Chaine de caractère
PIC	La pression intracrânienne	Chaine de caractère
Osmothérapie	L'administration de substances osmotiquement actives dans le but de diminuer la pression intracrânienne (PIC) et	Chaine de caractère

	d'améliorer l'hémodynamique cérébrale et/ou périphérique.	
Classe	La classe du cas	Chaine de caractère

Tableau 4 : La partie solution dans la base de cas

5. L'approche proposée

Ainsi, l'approche que nous suggérons est une application Web où l'utilisateur peut introduire le cas qu'il a reçu, lorsqu'il envoie la demande, un contrôleur principal reçoit cette demande et appelle un modèle qui implémente un cycle de raisonnement basé sur le cas.

Le système commencera par créer d'abord une description du cas pour que le système comprenne, puis le système démarre un cycle où le système recherche des cas similaires qui ont déjà été résolus, l'utilisateur adapte ensuite une solution au problème qu'il a introduit. Il révisé la solution qu'il a introduite avant d'adapter finalement la solution au problème. Dans ce cas, la base de cas reçoit le nouveau cas et l'utilisateur est averti que le nouveau cas a été adapté et revient à la page où il présente les nouveaux problèmes.

6. L'architecture générale

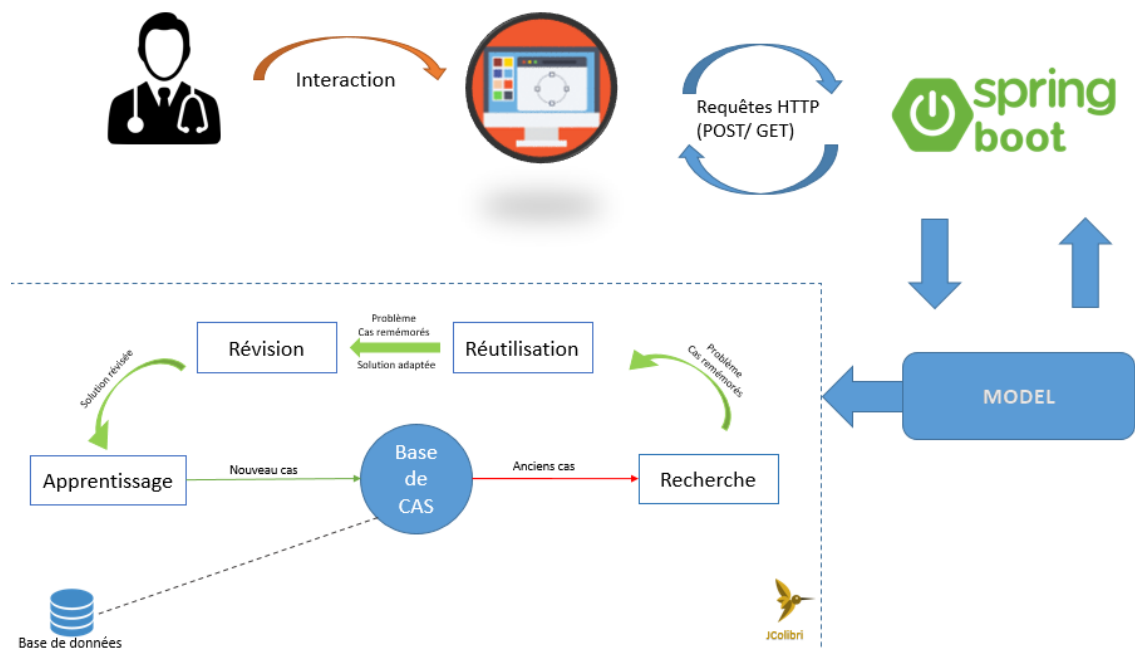


Figure 2 : L'architecture générale.

7. Les outils

7.1. JColibri

7.1.1. Définition :

JColibri est un framework orienté objet, développé en Java, dont le but est de construire (concevoir) un système basé sur le Raisonnement à Partir de Cas (RàPC).

7.1.2. L'architecture de JColibri

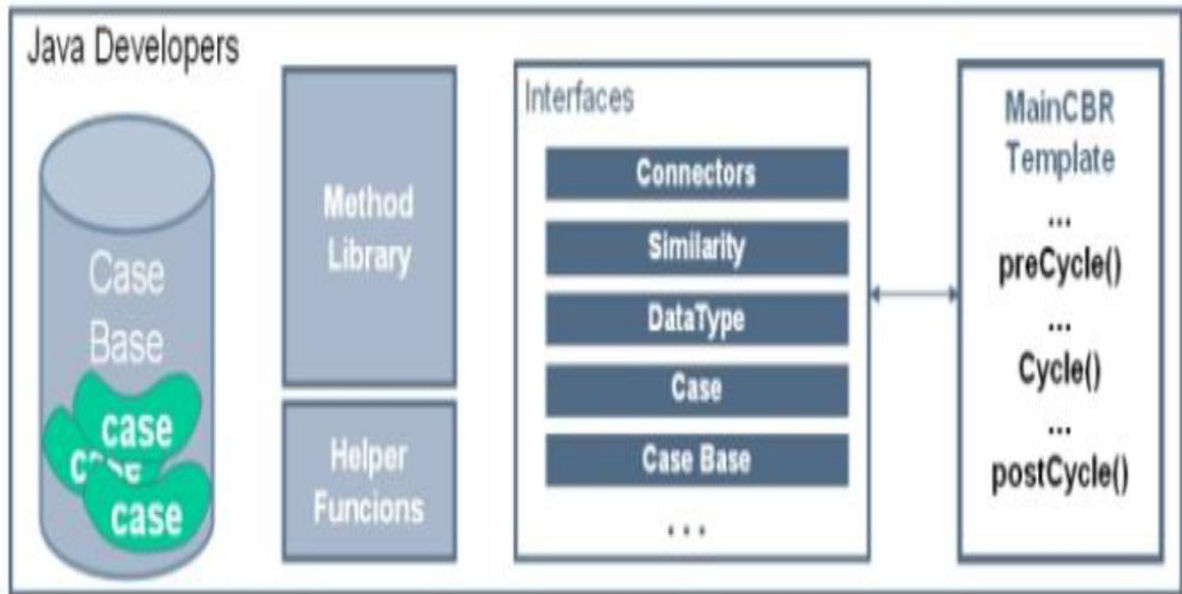


Figure 3 : Architecture de JColibri [21]

- La phase de recherche

Elle consiste à calculer la similarité locale entre le même descripteur pour chaque cas, ensuite elle calcule la similarité globale entre les cas.

Les mesure de similarité locales

Equal : est une fonction qui compare deux attributs qui ont des valeurs fixe, s'ils sont identiques elle retourne 0 sinon elle retourne 1

La distance euclidienne : est une fonction qui calcule la distance entre deux point x et y et est définie par la fonction suivante :

$$D(x,y) = \frac{\sqrt{x^2 - y^2}}{x + y}$$

Et retourne un nombre entre 0 et 1.

Prenant exemple pour le calcul de similarité du descripteur spo2 :

SPO2 peut prendre deux valeurs soit “Normal” soit “hypoxémie”, la fonction utilisée pour le calcul de similarité est égale si le nouveau cas a comme valeur pour le descripteur spo2 normal et un cas de la base de cas a comme valeur normale aussi

Alors equal retourne la valeur 0,

Si le cas dans la base de cas a comme valeur hypoxémie, la fonction equal retourne la valeur 1.

Le tableau ci-dessous présente la fonction de calcul de similarité pour chaque descripteur.

Le protocole	Les descripteurs	Les valeurs du descripteur	La fonction de calcul de similarité utilisé
A	Spo2	Normal, hypoxémie	Equal
B	Âge	Des valeurs entières	Distance euclidienne
	Sexe	F, M	Equal
	FR	Normal, DR-B, DR-H	
	Cyanose	true, false	
	Conscient	true, false	
	Glasgow	<=8,]8,12[, [12,15]	
	Râles	Normal, ronflants, sibilants	
	Encombrement bronchique	True, false	

C	PAS	Normal, hypertension, hypotension	Equal
	PAD	HVS-H, HVS-B, normal	
	FC	Normal, bradycardie, tachycardie	
	TRC	<3, 3, >3	
	Diérèse	Conservé, non conservé	
	Marbrure	True, false	
	T° extrémité	Froide, chaude, normal	
	T°	Double	Distance euclidienne
D	Réflexe	True, false	Equal
	Convulsion	True, false	
	Vomissement	True, false	
	Motrice spontané	True, false	
	Tonus	Hypo, bon, hyper	
	Mouvement oculaire	Oui, non	
	Fracture	True, false	

	Oedème	True, false
	Hémorragie	True, false
	Corps étranger	True, false

Tableau 5 : Association des mesures de similarité aux descripteurs

La mesure de similarité globale

Etant donné une description d'un nouveau cas et la similarité locale pour chaque descripteur, l'algorithme de recherche calcule la similarité globale pour tous les cas en attribuant un poids pour chaque descripteur et récupère les cas les plus similaires au nouveau cas, une des méthodes de récupération la plus courante et la bien connue est l'algorithme du K plus proche voisins (nearest neighbor) ou KNN.

La formule de calcul de similarité globale est :

$$\text{Sim}(C,S) = \sum_{f=1}^n w_f * \text{sim}(C_f,S_f)/\text{sum}(w_f)$$

C : le nouveau cas

S : le cas mémorisé

w : le poids défini par un médecin

n : le nombre de descripteur pour chaque cas

f : l'index du descripteur

sim(C_f,S_f) : la similarité locale pour le descripteur f.

- La phase d'adaptation
L'adaptation par définition rend envisageable de reprendre majoritairement ou absolument la solution qui existe déjà dans la base de cas.
- La phase de révision
Le médecin valide la solution adaptée.
- La phase d'apprentissage
Le nouveau cas est ajouté à la base de cas avec sa description et sa solution.

7.2. Spring boot

7.2.1. Définition

Spring Boot est un Framework open source basé sur Java, qui permet aux utilisateurs de créer des micro-services, un micro service permet aux développeurs de développer et de fournir des services de manière

indépendante [22]. Ce service nous permet de nous débarrasser du processus de configuration complexe.

L'avantage de ce Framework c'est qu'il n'impose plus aucun modèle de programmation unique, il est devenu célèbre au sein du réseau Java puisque pour les développeurs Java, développer et comprendre des applications Spring est relativement facile.

Vous pouvez démarrer vos applications plus rapidement avec une efficacité accrue [10].

7.2.2. L'architecture de SpringBoot

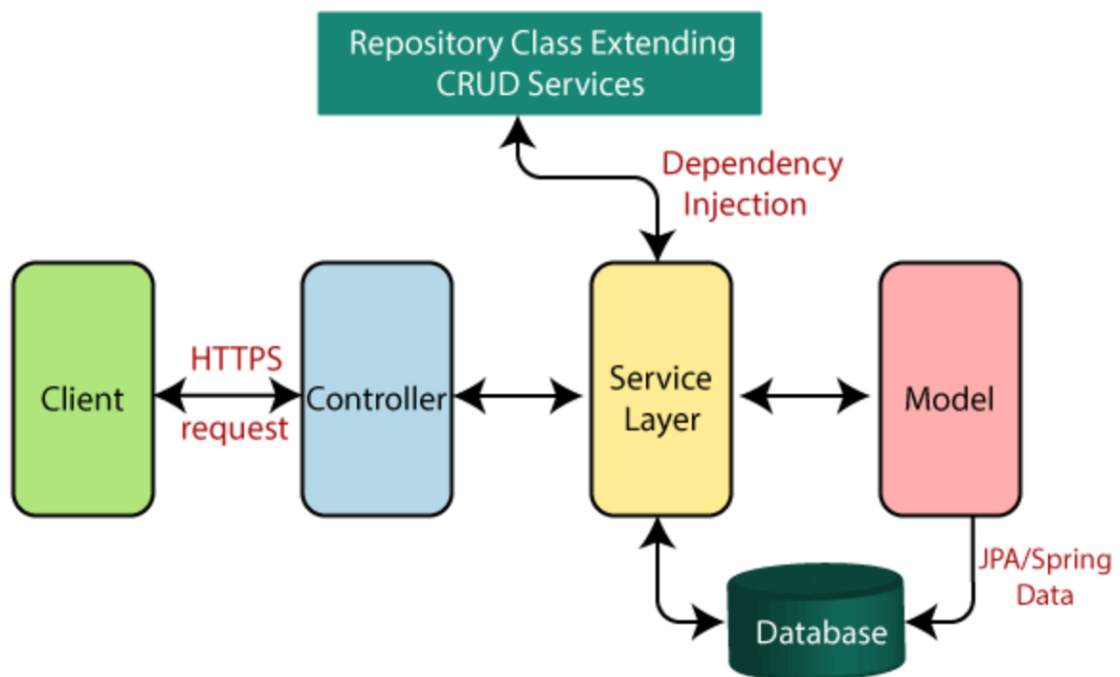


Figure 4 : Architecture de Spring Boot

8. Web service REST API

8.1. Coté client :

1. Prendre connaissance des interfaces que le service Web propose.
2. Construire la requête.
3. Envoyer la requête (HTTP le plus souvent).
4. Récupérer et interpréter les données retournées (JSON, XML ou TEXT).
5. Traiter les données (calculer, afficher).

8.2. Coté serveur :

1. Définir les interfaces publiques.
2. Récupérer les requêtes du client.
3. Traduire la requête et effectuer le traitement.

4. Envoyer la réponse normalisée dans un format standard (JSON, XML ou TEXT).

9. Diagramme de séquence

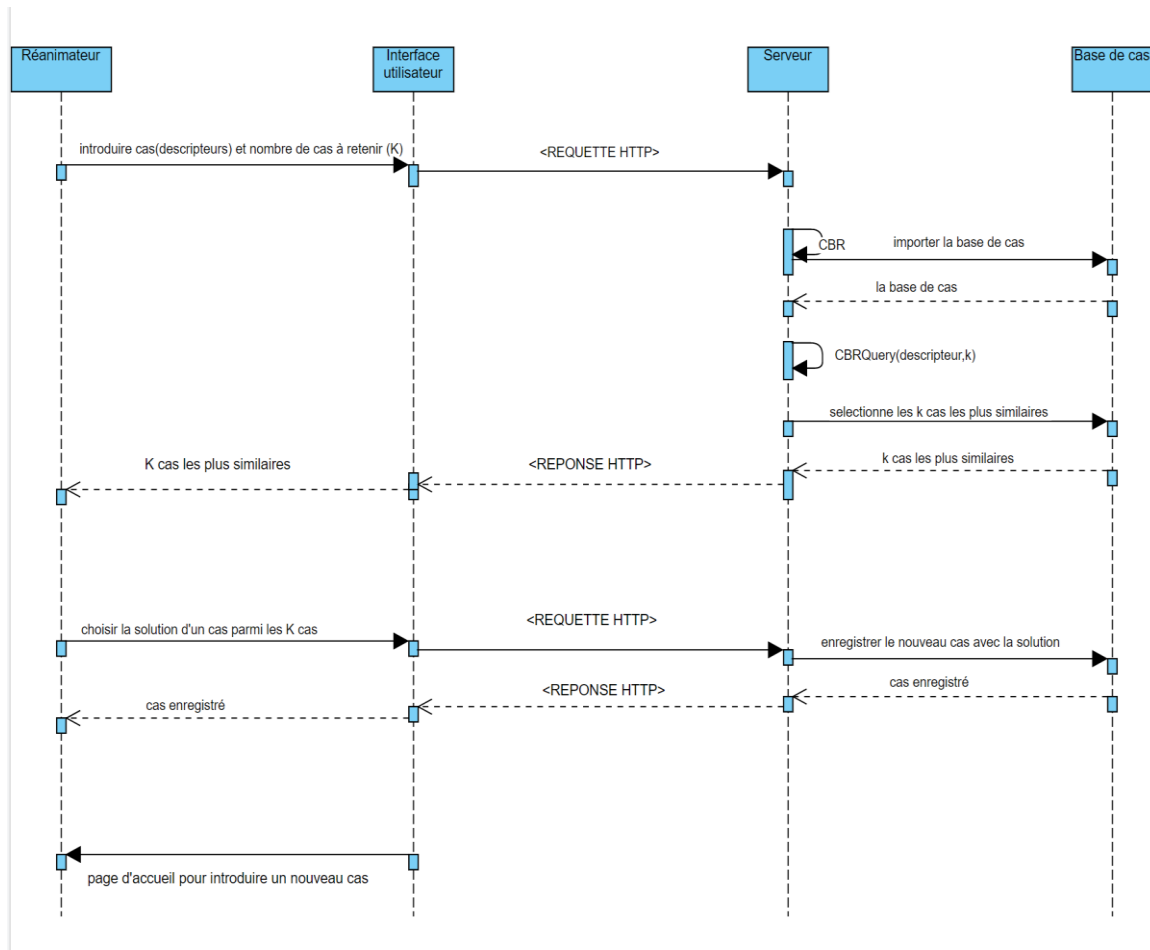


Figure 5 : Diagramme de séquence

1. Le médecin réanimateur introduit le cas en donnant les descripteurs et le nombre de cas à retenir "k"
2. L'interface envoie une requête http au serveur
3. Le serveur reçoit la requête et commence l'exécution du raisonnement à partir de cas
4. Le serveur importe la base de cas
5. Le serveur crée la requête CBR en utilisant les informations envoyées par le réanimateur depuis l'interface
6. Le serveur sélectionne K cas
7. Le serveur renvoie les cas sélectionnés
8. L'interface affiche au médecin les solutions des cas retenus

9. Le médecin choisit une solution pour la retenir et/ou l'adapterL'interfacee envoie une requête http au serveur
10. Le serveur sauvegarde le cas retenu dans la base de cas et dans la base de données
11. Le serveur envoie une notification que la rétention du cas a été exécutée avec succès.
12. L'interface affiche que l'exécution est faite avec succès et recharge la page pour introduire un nouveau cas.

10. Les captures

10.1. BACKEND

Donc, pour notre exécution de code backend, nous suivrons tout le chemin d'exécution de l'application où nous verrons nos points de terminaison et ce que chaque point appelle en tant que fonctions du modèle.

Test1.java

Premièrement on déclare les deux variables importantes qui seront utilisés tout au long de l'exécution du programme

```
Connector _connector;
```

```
CBRCaseBase _caseBase;
```

Ici on définit les fonctions les plus importantes dans notre modèle


```

public void configure() throws ExecutionException {
    try {
        // Create a data base connector
        _connector = (Connector) new DataBaseConnector();

        // Init the ddbb connector with the config file

        _connector.initFromXMLfile(jcolibri.util.FileIO.findFile("com/ouss/reanimati
on/model/databaseconfig.xml"));

        // Create a Lineal case base for in-memory organization
        _caseBase = (CBRCasBase) new LinealCaseBase();

    } catch (Exception e) {
        throw new ExecutionException(e);
    }
}

```

Figure 6 : La procédure configure()

Cette première méthode configure() spécifie que notre connecteur est un connecteur de base de données, elle récupère les informations d'accès aux données à partir d'un fichier xml nommé **databaseconfig.xml**.

Elle précise également que notre base de cas est une base de cas linéaire, cette base sera utilisée dans les prochaines étapes du processus.

```

public CBRCasBase preCycle() throws ExecutionException {
    // Load cases from connector into the case base
    _caseBase.init(_connector);
    // Print the cases
    Collection<CBRCas> cases = _caseBase.getCases();
    for (CBRCas c : cases)
        System.out.println(c);
    mycases = cases;
    return (CBRCasBase) _caseBase;
}

```

Figure 7 : La procédure preCycle()

La méthode preCycle() est la deuxième étape du processus. Elle initie notre base des cas et la remplit avec les cas récupérés de la base de cas, ces cas seront retournés par la fonction à utiliser dans la prochaine étape importante.

```

@Override
public void cycle(CBRQuery cbrQuery, int k) throws ExecutionException {
    // Obtain configuration for KNN

    NNConfig config = new NNConfig();
    jcolibri.cbrcore.Attribute attribute;
    //SimilConfigPanel similConfig;
    jcolibri.method.retrieve.NNretrieval.similarity.LocalSimilarityFunction
function;

    //similConfig = nom;
    attribute = new Attribute("Temperature", TraumaDescription.class);
    config.addMapping(attribute, new Euclidienne());

    attribute = new Attribute("Age", TraumaDescription.class);
    config.addMapping(attribute, new Euclidienne());

    attribute = new Attribute("Poids", TraumaDescription.class);
    config.addMapping(attribute, new Euclidienne());

    attribute = new Attribute("glasgow", TraumaDescription.class);
    config.addMapping(attribute, new Equal());
    config.setWeight(attribute, 0.1);

    attribute = new Attribute("tonus", TraumaDescription.class);
    config.addMapping(attribute, new Equal());

```

Figure 8 : La procédure Cycle()

Nous commençons dans cette procédure par déclarer une configuration, ce qui aide à configurer les attributs et leurs mesures de similarité

On note par exemple que la température et l'âge utilisent la distance Euclidienne, tandis que l'indice de Glasgow et le Tonus utilisent la mesure Equal.

On note également que certains des attributs ont un poids qui leur est attribué, ce qui correspond essentiellement à l'importance de cet attribut.

```

config.setDescriptionSimFunction((GlobalSimilarityFunction) new
Average());

```

Figure 9 : La fonction de similarité globale

Nous avons configuré notre config pour que notre cycle utilise la similarité globale de Moyenne

```

// Execute NN
Collection<RetrievalResult> eval =
NNScoringMethod.evaluateSimilarity(_caseBase.getCases(), cbrQuery,
config);

// Select k cases
Collection<CBRCas> selectedcases = SelectCases.selectTopK(eval,
k);
casestoreturn = selectedcases;

```

Figure 10 : L'exécution de recherche et la sélection de meilleurs cas

Enfin, nous exécutons le `NNscoringmethod` qui calcule essentiellement la similitude entre les cas et la description de notre problème, en utilisant la configuration que nous avons précédemment configurée.

Puis on sélectionne les K voisins les plus proches (on note que K a été spécifié par le client).

Nous collons cette collection de cas récupérés dans la variable **`casestoreturn`** qui sera ensuite accédée par le modèle.

mainController.java

Tout d'abord, nous voyons ici nos principaux endpoints, dont chacun appelle une fonction définie dans la classe Controller

```

@RequestMapping(value = "/sendsTrauma/{k}", consumes = "application/json",
produces = "application/json", method = RequestMethod.POST)
public Collection<CBRCas> getaResponse(@RequestBody TraumaDescription
requet, @PathVariable int k) throws ExecutionException
{System.out.println("/SendsTrauma IS CALLED");
Test1 test1 = new Test1();
try {
// Configure it
test1.configure();
// Run the precycle --> load the cases
test1.preCycle();

```

```

System.out.println("The received request is: " + request);

CBRQuery query = new CBRQuery();
query.setDescription(request);

// Run a cycle with the query
test1.cycle(query, k);

System.out.println("Cycle finished.");

// Run the postcycle
test1.postCycle();

} catch (Exception e) {
    System.out.println(e.getMessage());
    e.printStackTrace();
}
return test1.casestoreturn;
}

```

Figure 11 : L'endpoint /sendsTrauma

Le premier endpoint est **/sendsTrauma** qui reçoit un fichier Json du client contenant dans la description du problème et une variable K dans l'url du nombre de cas à retenir, ce endpoint lance une méthode qui retourne une collection de Cases qui sont dans notre cas, le candidat des cas similaires qui seront montrés dans la deuxième page.

Cela consiste à exécuter les méthodes précédentes (configure, precycle, cycle) et à renvoyer les cas sélectionnés côté client (il les renvoie de la variable **casestoreturn** au format Json)

```

@PostMapping(value = "/retainAcase", consumes = "application/json")
public void retainACase(@RequestBody CaseToRetainDTO myrequest) throws
ExecutionException {
    System.out.println("myrest is " + myrequest);

    TraumaDescription mydescription = (TraumaDescription)
myrequest.getDescription();
    TraumaSolution mysolution = (TraumaSolution) myrequest.getSolution();

    //inserting directly to the database (using queries)
    MaladieInsertRepo myrepo = new MaladieInsertRepo();
    myrepo.caseInsert(myrequest);

    Test1 tss = new Test1();
    tss.configure();
    tss.preCycle();
}

```

Figure 12 : L'endpoint /retainAcase

Le deuxième endpoint est **/retainAcase**, pour cet endpoint nous avons été obligés de créer un objet de transfert de données (CaseToRetainDTO) qui est notre nouveau cas à retenir, il concatène entre la description du problème donnée précédemment et la solution adaptée, cet objet est ajouté à une requête qui l'ajoute à la base de données. Et s'exécute actualise la base de cas pour afficher la base de cas actualisée.

Une autre méthode d'apprentissage était disponible dans ce cas, mais la base de cas que nous avons proposée comportait des problèmes natifs qui rendaient ce type d'apprentissage impossible.

Pour la requête d'insertion

```
String sql = "Insert into maladies (Age, sexe, Poids, Temperature, Pas, pad,
Fc,          tempextr, diurese," +
    " marbure, trc, spo2, fr, cyanose, rales, EncBranchique, glasgow,
tonus, conscient, convulsion," +
    " vomissement, motrice, mouvement, reflexe, fracture, Oedeme, corpsE,
hemorragie, o2ther, ssi, supportHemo," +
    " sedation, intubation, ventilation, pic," +
    " osmotherapie, Id) " +
    "values (?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?,
?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?) ";

PreparedStatement pst = conn.prepareStatement(sql);
TraumaDescription description = mycasetoretain.getDescription();
TraumaSolution solution = mycasetoretain.getSolution();

pst.setDouble(1, description.getAge());
pst.setString(2, description.getSexe());
pst.setDouble(3, description.getPoids());
...
...
pst.setString(29, solution.getO2ther());
pst.setString(30, solution.getSsi());
pst.setString(31, solution.getSupportHemo());

pst.setString(37, String.valueOf(mynewid));

pst.executeUpdate();
System.out.println("Case with Id " +mynewid + " has been added to the
database!");
```

Figure 13 : La requête d'insertion

On prépare une instruction de requête sql et on remplit les valeurs par les valeurs récupérées par l'objet du cas à retenir, on génère un identifiant pour le nouveau cas et on l'ajoute à la base de données.

On note que la phase de recherche a été exécutée dans la partie serveur et l'adaptation du nouveau cas est exécutée au côté client, et l'apprentissage est combiné entre l'utilisateur et le serveur.

Exécution

Lors de l'exécution du code, quelques impressions ont été utilisées pour suivre l'exécution du programme.

Comme dans ce premier qui imprime la base de cas après l'avoir importée.

```
[Description: (100;52.0;M;22.0;36.7;Normal;HVS-H;Normal;Froide;Conservee;true;>3;Normal;DR-B;true;;Ronfflants;true;<=8;Hypo;true;true;false;false;Non;false;t
[Description: (101;50.0;M;20.0;38.9;Hypertension;HVS-H;Bradycardie;Chaude;Conservee;false;<3;Normal;DR-H;true;;Normal;false;<=8;Bon;false;true;false;false;Nc
[Description: (103;81.0;M;20.0;38.0;Hypertension;HVS-H;Tachycardie;Chaude;Non Conservee;true;>3;Hypox?mie;DR-H;false;;Ronfflants;false;<=8;Hyper;false;true;t
[Description: (105;153.0;F;12.0;38.2;Hypotension;HVS-B;Bradycardie;Chaude;Non Conservee;true;>3;Normal;DR-H;true;;Ronfflants;false;<=8;Hypo;false;true;true;fa
[Description: (106;67.0;M;20.0;37.5;Hypertension;Normal;Bradycardie;Chaude;Conservee;false;<3;Normal;Normal;false;;Normal;false;<=8;Bon;false;true;false;fals
[Description: (108;139.0;M;45.0;36.0;Hypotension;HVS-B;Bradycardie;Normal;Conservee;false;>3;Normal;DR-H;false;;Cr?pitants;true;<=8;Bon;true;true;false;false
[Description: (109;156.0;M;40.0;37.0;Normal;Normal;Bradycardie;Normal;Conservee;false;>3;Normal;DR-H;false;;Normal;false;[12,15];Bon;true;false;true;true;Norm
[Description: (111;160.0;M;50.0;36.4;Hypertension;Normal;Bradycardie;Normal;Conservee;false;<3;Normal;DR-H;false;;Normal;false;[12,15];Bon;true;false;false;tr
[Description: (110;111.0;M;25.0;36.5;Hypotension;HVS-B;Bradycardie;Chaude;Conservee;true;>3;Normal;DR-H;true;;Ronfflants;false;<=8;Bon;false;true;false;false;
[Description: (112;74.0;F;20.0;37.2;Hypertension;HVS-H;Bradycardie;Normal;Conservee;false;<3;Normal;DR-H;false;;Normal;false;[12,15];Bon;false;false;false;tr
[Description: (113;140.0;M;45.0;36.5;Normal;Normal;Bradycardie;Chaude;Conservee;false;>3;Normal;DR-H;false;;Ronfflants;false;]8,12[;Bon;true;true;true;false;N
[Description: (115;11.0;F;12.0;36.5;Normal;Normal;Normal;Normal;Conservee;false;<3;Normal;DR-H;false;;Normal;true;[12,15];Bon;true;false;true;false;Normal;tr
[Description: (116;25.0;M;12.0;37.0;Normal;Normal;Bradycardie;Chaude;Conservee;true;>3;Normal;DR-H;true;;Normal;false;<=8;Bon;false;false;true;false;Normal;t
[Description: (118;61.0;M;20.0;38.0;Hypertension;HVS-H;Tachycardie;Normal;Conservee;false;<3;Normal;DR-B;false;;Ronfflants;false;<=8;Bon;false;true;false;fal
[Description: (119;42.0;M;25.0;37.4;Normal;Normal;Bradycardie;Normal;Conservee;false;<3;Normal;DR-B;false;;Normal;false;[12,15];Bon;true;false;true;true;Non;
[Description: (12;18.0;M;8.0;38.8;Hypertension;Normal;Normal;Normal;Non Conservee;false;>3;Normal;Normal;false;;Sibilants;false;<=8;Bon;false;true;false;true
[Description: (121;42.0;M;20.0;39.0;Hypertension;Normal;Normal;Normal;Non Conservee;true;>3;Hypox?mie;DR-H;true;;Sibilants;true;<=8;Hypo;false;true;true;fals
[Description: (122;123.0;M;35.0;39.2;Hypotension;Normal;Bradycardie;Chaude;Non Conservee;true;>3;Normal;Normal;true;;Normal;false;<=8;Hyper;false;true;true;t
[Description: (124;0.0;M;25.0;38.5;Hypertension;HVS-H;Tachycardie;Normal;Conservee;false;<3;Normal;DR-B;false;;Normal;false;<=8;Bon;false;false;true;false;Nc
```

Figure 14 : L'affichage de la base de cas importée

Ces valeurs imprimées sont les similitudes entre les cas (ce sont les valeurs qui seront triées et auront les meilleures K cas sélectionnés)

```
0.1705887719063862
0.140028008402801
0.21821789023599236
0.13192022253539623
0.46695300917967636
0.21821789023599236
0.14151573152725086
0.7019137602506937
0.5423261445466404
0.10383482633023292
0.3550358012483632
0.21821789023599236
0.09812548460529202
0.6749054616751541
0.5859040724295678
0.06380091150886687
0.7071067811865475
```

Figure 15 : L'affichage de similarités

Dans cet affichage suivant, il montre les meilleurs K cas, ils contiennent deux parties, la description du cas et la solution de celui-ci

```
Combined cases
[Description: (100;52.0;M;22.0;36.7;Normal;HVS-H;Normal;Froide;Conservee;true;>3;Normal;DR-B;true;;Ronfflants;true;<=8;Hypo;true;true;false;false;Non;false;t
[Description: (101;50.0;M;20.0;38.9;Hypertension;HVS-H;Bradycardie;Chaude;Conservee;false;<3;Normal;DR-H;true;;Normal;false;<=8;Bon;false;true;false;false;Nc
[Description: (103;81.0;M;20.0;38.0;Hypertension;HVS-H;Tachycardie;Chaude;Non Conservee;true;>3;Hypox?mie;DR-H;false;;Ronfflants;false;<=8;Hyper;false;true;t
```

Figure 16 : L'affichage de K meilleurs cas

Ce message indique au développeur que le cas avec son numéro d'identification a été ajouté à la base de données lors de la phase de mémorisation (cela permet de vérifier si le cas a été correctement ajouté)

```
Case with Id 1 has been added to the database!
```

Figure 17 : Affichage du message d'insertion avec son identifiant

10.2. FRONTEND

La page d'accueil avec un menu latéral :



Figure 18 : Interface d'accueil

Le médecin introduit les informations sur le patient et le résultat constaté lors des examens qui lui ont été effectués

INFORMATION PATIENT

info	info	info
<input type="text" value="age"/>	<input type="checkbox"/> conscient	<input type="text" value="glasgow"/>
<input type="text" value="poids"/>	<input type="checkbox"/> convulsion	<input type="text" value="temperature"/>
<input type="text" value="37°"/>	<input type="checkbox"/> vomissement	<input type="text" value="dieres"/>
<input type="text" value="sexe"/>		

Figure 19 : Interface information sur le patient

L'examen de respiration :

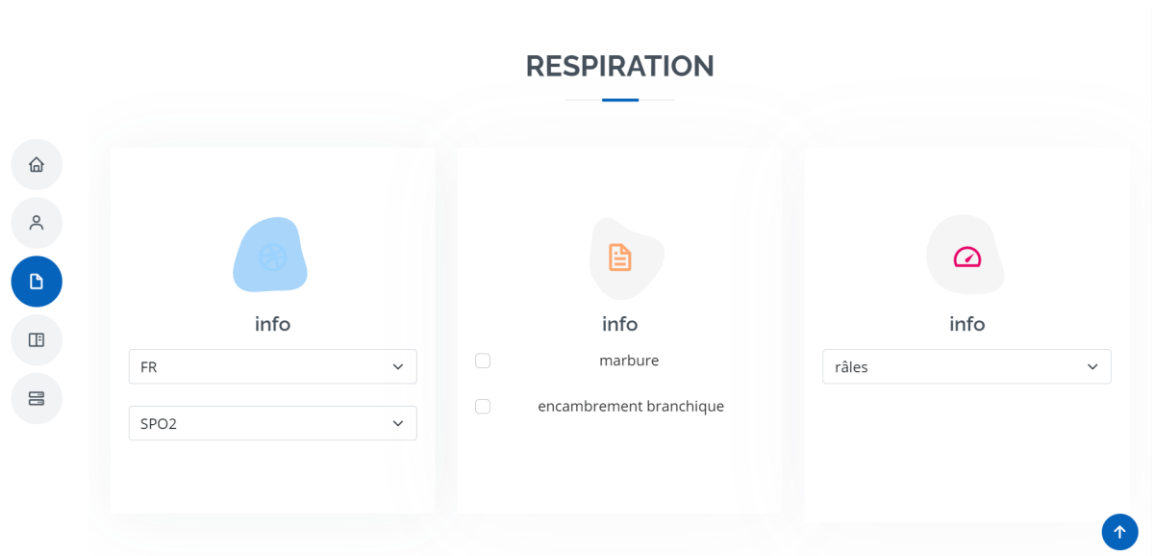


Figure 20 : Interface information sur la respiration

L'examen de l'hémodynamique

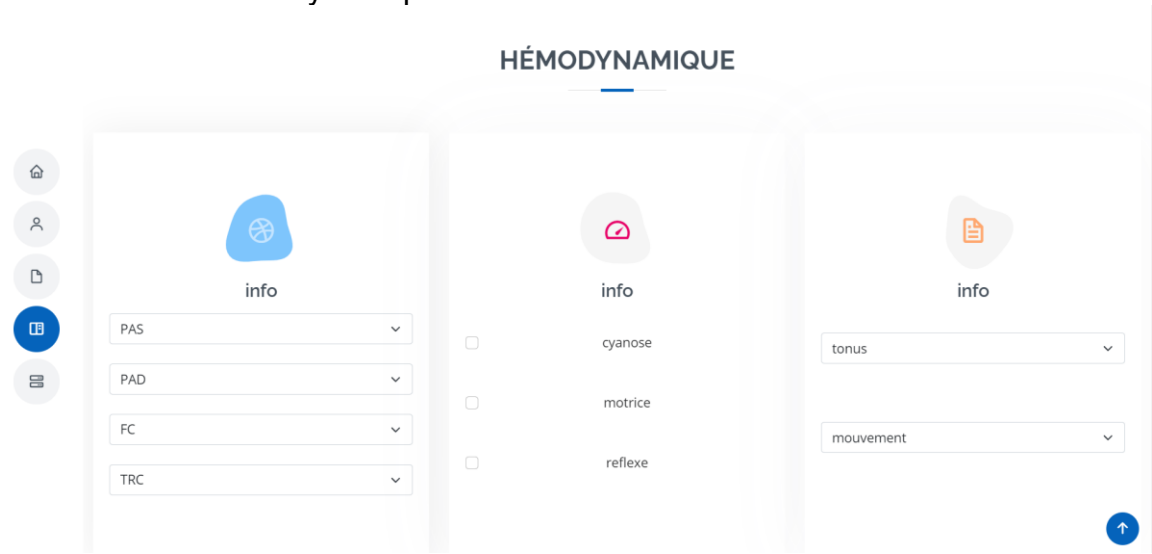


Figure 21 : Interface information sur l'hémodynamique

Le bilan des lésions

The interface is titled "BILAN DES LESIONS". On the left is a vertical sidebar with five icons: a house, a person, a document, a calendar, and a folder. The main area contains four categories of lesions, each with a checkbox and a blue smiley icon: "fracture", "oedeme", "corps étranger", and "hémorragie". Below these is a text input field labeled "nombre de cas a retenir". At the bottom center is a blue button labeled "EXECUTER", and at the bottom right is a blue circle with an upward arrow.

Figure 22 : Interface bilan des lésions

Le médecin choisit le nombre de cas similaires à retenir et envoie la requête au serveur.

le serveur lance l'exécution du cycle du raisonnement par cas en préparant d'abord la description du cas, puis le calcul de similarité entre le cas reçu et les cas existant dans la base de cas, ensuite en prenant en compte le nombre de cas à retenir que le médecin a choisi, le serveur exécute l'algorithme du k plus proche voisins et renvoie la réponse au médecin, les cas similaires s'afficheront sur le côté droit de la page avec des boutons suivant et précédent pour basculer entre les cas retenus tandis que le côté gauche contiendra les informations du patient introduites précédemment pour pouvoir comparer certains descripteurs

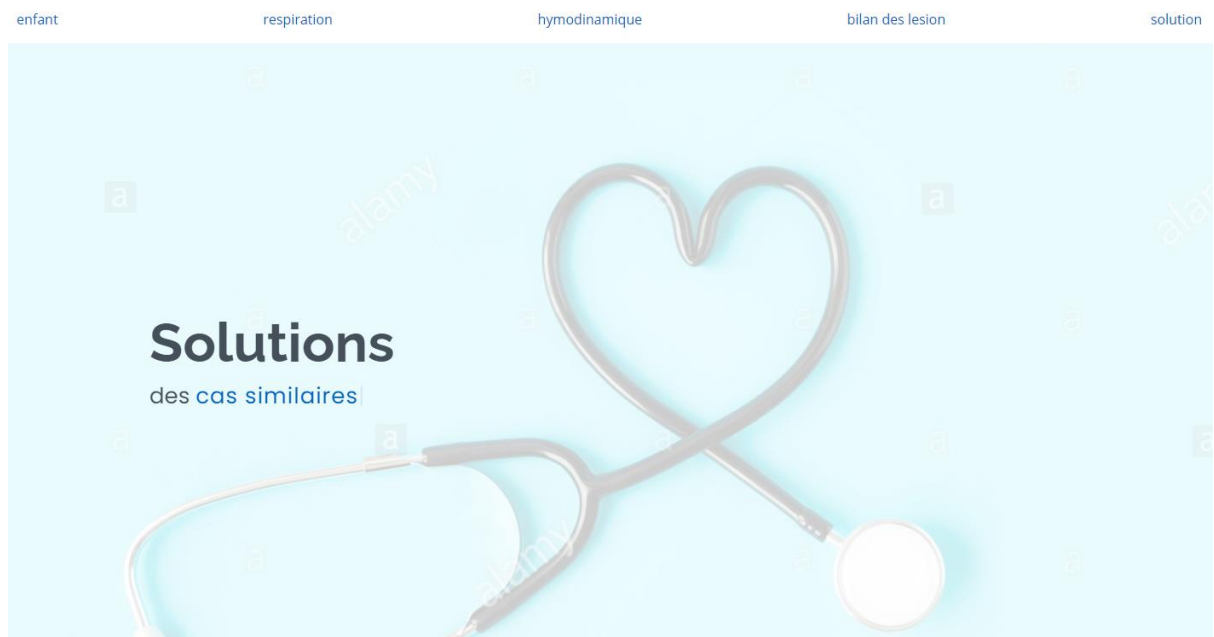


Figure 23 : Interface Solution

INFORMATION PATIENT

info


INFORMATION CAS SIMILAIRE

cas similaire

Figure 24 : Interface cas similaires

Le médecin peut ensuite apporter des modifications et adapter la solution dans cette partie

SOLUTION



solution


o2 therapie

intubation

osmotherapie

ventilation

SOLUTION CAS SIMILAIRE



solution cas similaire

o2 therapie


intubation

osmotherapie

ventilation


Figure 25 : Interface solutions des cas similaires

Puis il clique sur le bouton retenir la solution. Le serveur reçoit le nouveau cas qui contient la partie description et la partie solution et l'enregistre dans la base de cas et la base de données ainsi se fait l'apprentissage, la base de cas sera plus riche et le cycle du raisonnement à partir de cas se termine.



solution

classe



solution cas similaire

classe

Retenir la solution

Figure 26 : Interface pour retenir la solution

Version mobile :

INFORMATION PATIENT



info

age

poids

37°

sexe



Figure 27 : Interface version mobile



Figure 28 : Interface solution version mobile

11. Conclusion

Dans ce chapitre, nous avons en premier lieu présenté notre problématique qui est la prise en charge des enfants victime d'accident de la route. Par la suite, nous avons illustré l'architecture générale et le diagramme de séquence servant à modéliser notre système. Enfin nous avons présenté les outils qui nous ont permis de réussir à créer une application web qui communique avec l'API via des requêtes http, ces requêtes sont traitées par le serveur et appellent le modèle qui est lié à la base de données et à la bibliothèque JColibri et exécute la suite adéquate d'instructions, le serveur notifie à l'interface que l'appel de la fonction a été exécuté et renvoie la réponse qui sera ensuite affichée sur l'interface.

Conclusion générale

Ce projet de fin d'étude de master consiste à concevoir et réaliser une application intelligente d'aide à la décision pour la prise en charge des enfants victime d'accidents de la route, car le service de réanimation reçoit des enfants dans un état critique qui demande la prise en charge immédiate.

Pour réaliser ce travail nous avons présenté premièrement les systèmes d'aide à la décision, en particulier les systèmes d'aide à la décision médicale tel que les systèmes experts qui visent à reproduire le raisonnement humain. En second lieu nous avons présenté le raisonnement à partir de cas car beaucoup de travaux dans les systèmes d'aide à la décision médicale ont proposé des solutions basées sur ce raisonnement. Dans cette phase nous avons approfondi notre connaissance sur la méthodologie du le raisonnement à partir de cas puisqu'elle ne faisait pas partie de notre cursus de Master.

L'approche que nous avons proposée est réalisée en utilisant la plateforme JColibri implémentée en java pour pouvoir exploiter le cycle du raisonnement à partir de cas afin de nous recommander le traitement le mieux adapté pour soigner l'enfant victime d'accident. Et la plateforme Spring boot qui nous a permis de réaliser un RESTful web service pour les médecins réanimateurs. Une interface web a été réalisée en utilisant les langages HTML, CSS, BOOTSTRAP, JavaScript et JQuery, qui répond aux principaux critères ergonomiques à savoir la simplicité et la convivialité de l'interface graphique.

Finalement nous sommes arrivés à réaliser notre application intelligente d'aide à la décision médicale qui répond aux besoins des médecins réanimateurs et aux besoins des patients car cette application va permettre d'assister le médecin dans le processus de prise de décision en améliorant la qualité du diagnostic clinique ce qui va permettre d'améliorer également la qualité des soins et va offrir un meilleur rapport coût-efficacité des tests et des thérapies, et surtout éviter de tomber dans l'erreur.

Des améliorations pourraient aussi être apportées à cette application telles que :

- Donner accès aux médecins à l'aide de l'authentification.
- Sécurisation des données transitant entre le serveur et le client.
- Évaluation du modèle avec les médecins.
- Évaluer différentes mesures de similarité pour voir si cela offre une meilleure qualité de réponses.
- Améliorer les performances et la vitesse du traitement en optimisant le code.

Références

Webographie:

- [1] 'SYSTEME D'AIDE A LA DECISION MEDICALE', *studylibfr.com*.
<https://studylibfr.com/doc/850823/chapitre-i-systeme-d-aide-a-la-decision-medicale> (accessed Jun. 20, 2021).
- [3] 'RaPC_session1'.
https://perso.liris.cnrs.fr/alain.mille/enseignements/Master_PRO/TIA/RAPC/igc_rapc_Folder/RaPC_session1.html (accessed Jun. 29, 2021).
- [16] 'A Hybrid Approach Using Case-Based Reasoning and Rule-Based Reasoning to Support Cancer Diagnosis: A Pilot Study - PubMed'.
<https://pubmed.ncbi.nlm.nih.gov/26262174/> (accessed Jul. 04, 2021).
- [18] 'ACS: Asthma Care Services with the Help of Case Base Reasoning Technique - ScienceDirect'.
<https://www.sciencedirect.com/science/article/pii/S1877050915006456> (accessed Jul. 04, 2021).
- [22] 'Top 7 Exciting Spring Boot Projects & Topics For Beginners [2021]', *upGrad blog*, Jan. 05, 2021. <https://www.upgrad.com/blog/spring-boot-projects-topics-for-beginners/> (accessed Jun. 20, 2021).

Bibliographie:

- [2] B. B. Naouel, 'Contribution à la surveillance pédiatrique : vers un système de prévention des accidents domestiques', 2018.
- [4] J. L. Kolodner, 'An introduction to case-based reasoning', *Artif. Intell. Rev.*, vol. 6, no. 1, pp. 3–34, Mar. 1992, doi: 10.1007/BF00155578.
- [5] B. Fouad and B. Fethi, 'Raisonnement à Partir de Cas En Utilisant le Système jCOLIBRI'.
- [6] Mohamed Karim Haouchine, 'Remémoration guidée par l'adaptation et maintenance des systèmes de diagnostic industriel par l'approche du raisonnement à partir de cas.'
- [7] A. Aamodt and E. Plaza, 'Case-Based Reasoning: Foundational Issues, Methodological Variations, and System Approaches', *AI Commun.*, vol. 7, no. 1, pp. 39–59, 1994, doi: 10.3233/AIC-1994-7104.
- [8] B. Fadi, 'Extraction de connaissances d'adaptation en raisonnement `a partir de cas', p. 112, 2009.
- [9] P. Koton, 'A medical reasoning program that improves with experience', *Comput. Methods Programs Biomed.*, vol. 30, no. 2–3, pp. 177–184, Nov. 1989, doi: 10.1016/0169-2607(89)90070-9.
- [10] L. Gierl and S. Stengel-Rutkowski, 'Integrating consultation and semi-automatic knowledge acquisition in a prototype-based architecture: Experiences with dysmorphic syndromes', *Artif. Intell. Med.*, vol. 6, no. 1, pp. 29–49, Feb. 1994, doi: 10.1016/0933-3657(94)90056-6.
- [11] R. T. Macura and K. J. Macura, 'MacRad: Radiology image resource with a case-based retrieval system', in *Case-Based Reasoning Research and Development*, Berlin, Heidelberg, 1995, pp. 43–54. doi: 10.1007/3-540-60598-3_5.
- [12] E. Golobardes, X. Llorà, M. Salamó, and J. Martí, 'Computer aided diagnosis with case-based reasoning and genetic algorithms', *Knowl.-Based Syst.*, vol. 15, no. 1, pp. 45–52, Jan. 2002, doi: 10.1016/S0950-7051(01)00120-4.

- [13]M. U. Ahmed, S. Begum, P. Funk, N. Xiong, and B. von Scheele, 'A multi-module case-based biofeedback system for stress treatment', *Artif. Intell. Med.*, vol. 51, no. 2, pp. 107–115, Feb. 2011, doi: 10.1016/j.artmed.2010.09.003.
- [14]S. Banerjee and A. R. Chowdhury, 'Case Based Reasoning in the Detection of Retinal Abnormalities Using Decision Trees', *Procedia Comput. Sci.*, vol. 46, pp. 402–408, Jan. 2015, doi: 10.1016/j.procs.2015.02.037.
- [15]S. Chakraborty, C. Pal, S. Chatterjee, B. Chakraborty, and N. Ghoshal, 'Knowledge-Based System Architecture on CBR for Detection of Cholera Disease', in *Intelligent Computing and Applications*, New Delhi, 2015, pp. 155–165. doi: 10.1007/978-81-322-2268-2_17.
- [17]K. Gulmira, '(PDF) Retrieval with Clustering in a Case-Based Reasoning System for Radiotherapy Treatment Planning', Accessed: Jul. 04, 2021. [Online]. Available: https://www.researchgate.net/publication/277253779_Retrieval_with_Clustering_in_a_Case-Based_Reasoning_System_for_Radiotherapy_Treatment_Planning
- [19]Z. Yin, Z. Dong, X. Lu, S. Yu, X. Chen, and H. Duan, 'A clinical decision support system for the diagnosis of probable migraine and probable tension-type headache based on case-based reasoning', *J. Headache Pain*, vol. 16, p. 29, Apr. 2015, doi: 10.1186/s10194-015-0512-x.
- [20]D. A. Sharaf-El-Deen, I. F. Moawad, and M. E. Khalifa, 'A New Hybrid Case-Based Reasoning Approach for Medical Diagnosis Systems', *J. Med. Syst.*, vol. 38, no. 2, p. 9, Jan. 2014, doi: 10.1007/s10916-014-0009-1.
- [21]B. Afif, 'Le Raisonnement à Partir de Cas dans les Systèmes d'Aide à la Décision Médicale Prise en Charge de l'Enfant Victime d'un Accident de la Route', 2017.