UNIVERSITÉ DU BURUNDI
GRENIER DU SAVOIR DU BURUNDI
Dépôt institutionnel officiel
https://repository.ub.edu.bi

Conception et réalisation d’un système de gestion automatisé de la Banque de sang : cas du CNTS

GAKUBA, Joëlle-Fidès

2021-09
UB, Faculté des Sciences de l’Ingénieur(FSI)
https://repository.ub.edu.bi/handle/123456789/720

Document téléchargé depuis le **Grenier du Savoir du Burundi** — https://repository.ub.edu.bi

République du Burundi
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Université du Burundi
Faculté des Sciences de l'Ingénieur



Département de TIC
Mastère en Génie Informatique
Année Académique :

2019-2020

MEMOIRE

Présenté à la

Faculté des Sciences de l'Ingénieur(FSI)

En vue de l'obtention du grade de

MASTER

en

Génie Informatique

Par:

GAKUBA Joëlle-Fidès

Conception et réalisation d'un système de gestion automatisé de la Banque de sang : cas du CNTS

Soutenu le 07/09/2021, devant le jury composé de :

Dr.	NKUNZIMANA Hilaire	Président
Dr.	SAHINGUVU William	Vice-Président
Dr.	NDAYISABA Longin	Secrétaire
Dr.	MUKESHIMANA Michèle	Directeur
Pr.	NDIKUMAGENGE Jérémie	Membre

IDENTIFICATION DES MEMBRES DU JURY

Dr MUKESHIMANA Michèle : Directeur de Thèse

Dr NKUNZIMANA Hilaire : Président du Jury

Dr SAHINGUVU William : Vice-Président

Dr NDAYISABA Longin : Secrétaire

Pr NDIKUMAGENGE Jérémie: Membre

DEDICACES

A mes regrettés parents ;

A mon époux ;

A mes enfants ;

A mon frère et ma soeur ;

A toute ma famille ;

A tous mes amis et camarades de classe ;

A tous ceux qui me donnent des raisons d'être ce que je suis ;

Joëlle-Fidès GAKUBA

REMERCIEMENTS

Ce mémoire est le fruit d'un travail d'équipe de diverses personnes à diverses compétences. Chaque personne y a apporté quelque chose de son domaine de spécialisation, explicitement ou implicitement. Que tous ceux et celles qui, de près ou de loin, ont joué un rôle de quelque nature que ce soit trouvent ici nos profonds sentiments de gratitude.

Nos sincères remerciements sont adressés plus particulièrement à Dr Michèle MUKESHIMANA, Professeur à l'Université du Burundi qui, malgré ses nombreuses obligations, a accepté de diriger ce mémoire. Ses conseils, ses directives et sa disponibilité nous ont permis de mener à bien notre travail.

Nos remerciements sont ensuite adressés aux membres du Jury qui ont accepté de lire ce mémoire et de participer à son appréciation. Que tous les professeurs qui nous ont enseigné, durant tout notre parcours depuis l'école primaire, trouvent ici l'expression de notre profonde gratitude. Vous avez fait de nous ce que nous sommes aujourd'hui.

À tous les ami(e)s de classe, avec qui nous avons partagé les connaissances, merci d'avoir été près de nous. Votre soutien et votre aide nous ont été précieux.

Notre reconnaissance va également à l'endroit des responsables et des employés du CNTS qui nous ont prêté une oreille attentive. Ils ont su répondre à nos questions les plus pertinentes et, partant, nous ont permis de mener à bien notre projet.

Nous remercions enfin, tous ceux qui, d'une façon ou d'une autre, nous ont manifesté un attachement particulier et nous ont témoigné leur amour et sympathie. Nous pensons plus particulièrement à nos proches familles et ami(e)s; qu'ils soient rassurés de notre affectueuse reconnaissance.

LISTE DES SIGLES ET ABBREVIATIONS

1.	CNTS :	Centre National de Transfusion Sanguine
2.	CRTS :	Centre Régional de Transfusion Sanguine
3.	CSS:	Cascading Style Sheet
4.	HTML:	HyperText Markup Language
5.	MERISE:	Méthode d'Etude et de Réalisation Informatique pour les Systèmes d'Entreprise
6.	MSPLS:	Ministère de la Santé Publique et de la Lutte contre le SIDA
7.	MVC :	Model View Controller
8.	OMG:	Object Management Group
9.	OMT:	Object Modeling Technique
10.	OOD:	Object Oriented Design
11.	OOSE:	Oriented Object Software Engineering
12.	PHP:	Personal Home Page
13.	RUP:	Rational Unified Process
14.	SQL:	Structured Query Language
15.	UML:	Unified Modeling Language
16.	UP:	Unified Process
17.	XML:	eXtensible Markup Language

LISTE DES FIGURES

Figure 1: Organigramme du CNTS 5
Figure 2: Diagramme de cas d'utilisation pour le Responsable d'une Institution sanitaire 23
Figure 3: Diagramme de cas d'utilisation pour le Responsable du CNTS et DG 24
Figure 4:Diagramme d'activité pour l'authentification 28
Figure 5: Diagramme d'activité pour faire la demande de sang 29
Figure 6: Diagramme d'activité pour l'enregistrement d'une institution sanitaire 30
Figure 7: Diagramme de séquence pour l'authentification 31
Figure 8: Diagramme de séquence pour la demande de sang 32
Figure 9:Diagramme de séquence pour l'enregistrement d'une institution sanitaire 33
Figure 10: Diagramme de classe 34
Figure 11: Diagramme de déploiement 35
Figure 12: Modèle MVC 48
Figure 13: Page d'accueil de l'application 50
Figure 14: Authentification des utilisateurs 51
Figure 15: Page pour la création d'un compte d'une Institution 51
Figure 16: Page de faire la demande nominative 52
Figure 17: Page pour la demande de banque de sang 53
Figure 18: Page de visualiser des demandes déjà servies 53
Figure 19: Page en .pdf de la fiche de demande nominative 54
Figure 20: Page en.pdf de la fiche de demande de Banque de sang 55
Figure 21: Page des factures des demandes déjà terminées 55
Figure 22: Page pour visualiser les factures 56
Figure 23: Facture en .pdf 56
Figure 24: Page des statistiques 57
Figure 25: Page d'enregistrement d'une institution 57
Figure 26: Page d'attribution des rôles à une institution sanitaire 58
Figure 27: Page pour les entrées dans le stock 59
Figure 28: Page pour le stock disponible 59
Figure 29: Page de livraison des demandes 60
Figure 30: Page de facturation 60
Figure 31: Facture en .pdf 61
Figure 32: Page des notifications 61

Figure 33: Page des statistiques 62
Figure 34: Page de gestion des rôles des utilisateurs 62
Figure 35: Page des rapports 63
Figure 36: Page d'un rapport en .pdf 63

LISTE DES TABLEAUX

Tableau 1: Scenario nominal pour le cas d'utilisation : « s'authentifier » 25
Tableau 2: Scenario nominal pour le cas d'utilisation: « faire demande de sang »..... 26
Tableau 3: Scenario nominal pour le cas d'utilisation : « Gérer les institutions » 27

ABSTRACT

The general mission of the Center is to organize blood transfusion, establish the quality assurance system, ensure quality control and ensure the safety of blood products and their derivatives. From the interviews, documentation and observations made at the CNTS and the health institutions, we noticed that the blood distribution process is carried out manually. We have noticed challenges with blood requests, tracking, deliveries, payment and related reporting. Therefore, we made the choice to implement a web-based computer application of an automated blood bank management system at CNTS. We designed this new system with UML, which is a graphic modeling language based on pictograms. Then we used the mathematical formalism of relational algebra and finally we finished with the implementation and realization of this new information system with the programming language PHP. This system will allow the management of distribution and supply of blood products by creating a database to facilitate the recording and accessibility of data, the monitoring of inputs and outputs and reports related to stock and invoicing.

Keywords: CNTS, blood bank, computer application, relational algebra

RESUME

Le Centre a pour mission générale d'organiser la transfusion Sanguine, d'instaurer le système d'assurance qualité, d'assurer le contrôle de qualité et de veiller à la sécurité des produits sanguins et leurs dérivés. A partir des interviews, documentations et observations faits au CNTS et les institutions sanitaires, nous avons remarqué que le processus de distribution de sang se réalise d'une façon manuelle. Nous avons remarqué des défis lors des demandes de sang, le suivi, les livraisons, le paiement et les rapports liés. De ce fait, nous avons fait le choix d'implémenter une application informatique basée sur le web d'un système de gestion automatisé de la banque de sang au CNTS. Nous avons fait la conception de ce nouveau système avec le langage UML, ce dernier est un langage de modélisation graphique à base de pictogrammes. Ensuite nous avons utilisé le formalisme mathématique de l'algèbre relationnelle et enfin nous avons terminé par l'implémentation et la réalisation de ce nouveau système d'information avec le langage de programmation PHP. Ce système permettra la gestion de distribution et approvisionnement des produits sanguins en créant une base de données pour faciliter l'enregistrement et l'accessibilité des données, le suivi des entrées et sorties et les rapports liés au stock et facturation.

Mots clés : CNTS, banque de sang, application informatique, algèbre relationnelle

TABLE DES MATIERES

IDENTIFICATION DES MEMBRES DU JURY i

DEDICACES ii

REMERCIEMENTS iii

LISTE DES SIGLES ET ABBREVIATIONS iv

LISTE DES FIGURES v

LISTE DES TABLEAUX vii

ABSTRACT viii

RESUME ix

TABLE DES MATIERES x

AVANT-PROPOS xiii

0. INTRODUCTION GENERALE 1

CHAPITRE I. APERCU GENERAL ET CADRE DU PROJET 3

I.1. Présentation du Centre National de Transfusion Sanguine 3

I.1.1. Contexte 3

I.1.2. Mission du CNTS 3

I.2. Présentation du projet de recherche 6

I.2.1 Analyse de l'existant 6

I.2.2. Procédure de distribution des produits sanguins. 6

I.2.3. Problématique 10

I.2.4. Critique de l'existant 10

I.2.5. Solutions proposées 10

I.2.6. Objectif du projet 11

I.2.7. Intérêt du sujet 12

I.2.8. Délimitation du sujet 12

I.2.9. Apports scientifiques et technologiques 13

I.2.10. Méthodologie de recherche et domaine d'application 13

I.2.11. Outil Mathématique utilisée 14

CHAPITRE II. CONCEPTION ET MODELISATION DU NOUVEAU SYSTEME AVEC UML 15

II.1. Présentation du Langage UML 15

II.1.1. Introduction 15

II.1.2. Présentation des diagrammes UML 16

II.1.3. Les diagrammes structurels 16

II.1.4 Les diagrammes de comportements 19

II.2. Modélisation du nouveau Système de gestion de la Banque de sang 21

II.2.1. Démarche de développement 21

II.2.2. Les principes d'UP 21

II.2.3. Diagramme de cas d'utilisation 22

II.2.4. Diagramme d'activité 28

II.2.5 Diagramme de séquence 31

II.2.6. Diagramme de classe 34

II.2.7. Diagramme de déploiement 35

CHAPITRE III. MODELISATION DU NOUVEAU SYSTEME AVEC LE FORMALISM.E MATHEMATIQUE DE L'ALGEBRE RELATIONNELLE 36

III.1. Introduction 36

III.2. Opérateurs ensemblistes 36

III.2.1. Opérateur union (∪) 36

III.2.2. Opérateur intersection (∩) 37

III.2.3. Opérateur différence (-) 37

III.2.4. Opérateur produit cartésien (x) 37

III.3. Opérateurs propres aux Bases de données 37

III.3.1. Renommage $\boldsymbol{\alpha}$ 37

III.3.2. Sélection $\boldsymbol{\sigma}$ 38

III.3.3. Projection $\prod$ 38

III.3.4. Division (/) 38
III.3.5. Jointure ∞ 39
III.4. Requêtes SQL de notre nouveau système et en Algèbre relationnelle 39
III.4.1. Opérations unaires 39
III.4.2. Opérations binaires et n-aires 41
CHAPITRE IV : REALISATION DE L'APPLICATION DE GESTION DE LA BANQUE DE SANG 45
IV.1. Choix d'une application Web 45
IV.2. Choix des technologies à utiliser 47
IV.3. Présentation de l'application de gestion de la Banque de sang 50
IV.3.1. Authentification des utilisateurs 51
IV.3.2. Responsables des institutions sanitaires 52
IV.3.3. Responsable du CNTS 57
IV.3.4. Directeur 62
CONCLUSION ET RECOMMANDATIONS 65
Conclusion 65
Recommandations 66
REFERENCES BIBLIOGRAPHIQUES 67
ANNEXES 69
A. Fiche de réquisition individuelle nominative 69
B. Fiche de réquisition des banques de sang 70
C. Registre des épreuves de compatibilité 70
D. Registre de livraison de sang 71
E. Bordereau d'expédition 71

AVANT-PROPOS

Comme le stipule le sujet de notre travail, Conception et Réalisation d'un système de gestion automatisé de la Banque de Sang : cas du Centre National de Transfusion Sanguine. Ce travail a été effectué à partir des interviews, documentations et observations faits au CNTS et les institutions sanitaires.

Notre travail est organisé en 4 chapitres hormis l'introduction générale. Le premier chapitre porte sur l'aperçu général et cadre du projet. Dans ce chapitre sera analysé également l'existant pour pouvoir proposer une approche d'une solution informatique. Le second chapitre comprend deux parties : la première partie fait le point sur la théorie du langage UML tandis que la seconde est la concrétisation de la première, nous modélisons le nouveau système grâce à ce langage. Le troisième chapitre présente la modélisation de ce nouveau système avec le formalisme mathématique de l'algèbre relationnelle qui va nous permettre une modélisation logique simple et puissante, il va fournir également un ensemble de concepts pour manipuler formellement l'information ainsi modélisée. Le quatrième chapitre est centré sur deux parties ; la première sur les détails techniques qui entrent en jeu lors de la réalisation de l'application et la deuxième partie fait la présentation de l'application et son évaluation pour vérifier si les résultats permettent d'atteindre les objectifs fixés.

Le choix de ce sujet a été soutenu par le fait que le CNTS est une institution la plus sollicitée. Et vue la complexité de certaines opérations, particulièrement celles liées à la gestion des produits sanguins, il s'avère qu'une informatisation ne serait pas de trop. Le CNTS contribue dans l'amélioration des soins de santé et, l'informatisation de ce dernier ne ferait qu'accroître la qualité des services fournis.

0. INTRODUCTION GENERALE

Le temps passe vite et l'ère numérique cède sa place à l'ère de l'information. L'information est devenue un élément crucial au cœur des entreprises, des institutions et dans notre vie quotidienne. Depuis quelques temps, les petites et moyennes entreprises se sont conscientisées de la valeur ajoutée qu'apporte un système bien informatisé. Le besoin croissant des décideurs, d'avoir la bonne information au bon moment, trouve satisfaction dans l'informatique. En effet l'informatisation permet de traiter l'information pour qu'elle soit présentée à l'utilisateur avec un sens qui lui confère toute sa valeur. Mais avant d'informatiser il convient de se poser cette question : « Pourquoi veut-on informatiser ? » Petit Abir[1] de l'Université de Picardie donne une réponse à cette question. Cette réponse identifie clairement les objectifs d'une informatisation et les caractérise de la manière suivante :

1. L'amélioration des services aux clients ;
2. L'amélioration des services aux utilisateurs ;
3. L'amélioration des outils de gestion ;
4. La réponse à la crise du système manuel.
5. La maîtrise des coûts ;

Dans le noble souci de voir tous les services publics et privés jouissant des systèmes de gestion classiques se trouve un jour munis d'un système de gestion automatisé, il importe pour un informaticien de songer à mettre en place des systèmes de gestion automatisés pour stimuler les décideurs à réaliser les avantages de ces derniers et même d'en acheter, afin de rendre la production nationale plus abondante.

Néanmoins, au Burundi comme dans certains autres pays en voie de développement, la plupart des entreprises, pharmacies, magasins, institutions et hôpitaux ne sont pas informatisés. Parmi les institutions non informatisées, le CNTS ne fait pas l'exception. Elle sera donc l'objet de notre travail intitulé **«CONCEPTION ET REALISATION D'UN SYSTEME DE GESTION AUTOMATISE DE LA BANQUE DE SANG : CAS DU CNTS ».**

Le CNTS est une institution la plus sollicitée. Et vue la complexité de certaines opérations, particulièrement celles liées à la gestion des produits sanguins, il s'avère qu'une informatisation ne serait pas de trop. Le CNTS contribue dans l'amélioration des soins de santé et, l'informatisation de ce dernier ne ferait qu'accroître la qualité des services fournis.

La banque de sang est un endroit où les poches de sang collectées lors des événements de don de sang sont stockées en un seul endroit. Le terme «banque de sang» fait référence à une division d'un laboratoire hospitalier où le stockage des produits sanguins a lieu et où des tests appropriés sont effectués pour réduire le risque de contamination liés à la transfusion.

Le système de gestion des banques de sang est un système basé sur le Web qui aide à avoir les informations des poches de sang lors des commandes, des ventes et des livraisons dans la banque de sang. Avec ce système, l'utilisateur de ce système peut faire une commande en ligne et le gestionnaire vérifie si la demande peut être disponible et passe à la livraison des poches de sang selon le type.

À partir de ce système, plusieurs types de rapports peuvent être générés, tels que les rapports sur les stocks de sang, les commandes et les livraisons en fonction des mois et de l'année et des statistiques. Par conséquent, Le système de gestion des banques de sang rendra le stock de la banque de sang plus systématique et plus facile à gérer.

Notre travail est organisé en 4 chapitres hormis l'introduction générale. Le premier chapitre porte sur l'aperçu général et cadre du projet. Dans ce chapitre sera analysé également l'existant pour pouvoir proposer une approche d'une solution informatique. Le second chapitre comprend deux parties : la première fait le point sur la théorie du langage UML tandis que la seconde est la concrétisation de la première, nous modélisons le nouveau système grâce à ce langage. Le troisième chapitre présente la modélisation de ce nouveau système avec le formalisme mathématique de l'algèbre relationnelle qui va nous permettre une modélisation logique simple et puissante, il va fournir également un ensemble de concepts pour manipuler formellement l'information ainsi modélisée. Le quatrième chapitre est centré sur deux parties ; la première sur les détails techniques qui entrent en jeu lors de la réalisation de l'application et la deuxième partie fait la présentation de l'application et son évaluation pour vérifier si les résultats permettent d'atteindre les objectifs fixés.

CHAPITRE I. APERCU GENERAL ET CADRE DU PROJET

I.1. Présentation du Centre National de Transfusion Sanguine

I.1.1. Contexte

Le Centre National de Transfusion Sanguine en abrégé « CNTS » ci-après dénommé « LE CENTRE » érigé en est une administration personnalisée de l'Etat par décret n° 100/073 du 28 Avril 1993 est un service public doté d'une personnalité juridique d'un patrimoine propre et d'une autonomie de gestion. Il est placé sous l'autorité hiérarchique du Ministre ayant la Santé Publique dans ses attributions. Le siège du Centre National de Transfusion Sanguine est établi à BUJUMBURA. Il peut être transféré en tout autre endroit du territoire national par décision du Ministère de tutelle et après avis du Conseil d'Administration. Le Centre a pour mission générale d'organiser la transfusion Sanguine, d'instaurer le système d'assurance qualité, d'assurer le contrôle de qualité et de veiller à la sécurité des produits sanguins et leurs dérivés. Son mandat spécifique est notamment de :

1. faire appliquer les normes de qualité en matière de transfusion sanguine ;
2. organiser la collecte et la distribution du sang et ses dérivés sur le territoire national ;
3. appuyer techniquement les services hospitaliers de transfusion en assurant la formation et le perfectionnement de leur personnel.

Le Centre est relayé par les Centres Régionaux de Transfusion Sanguine en abrégé « CRTS ». Les CRTS sont sous le contrôle administratif, financier et Technique de la Direction du Centre.

I.1.2. Mission du CNTS

La mission de la CNTS est de (d'):

1. Acquérir tout matériel, les équipements et réactifs nécessaires à la transfusion sanguine ;
2. Faire appliquer les normes de qualité en matière de transfusion sanguine ;
3. Appuyer financièrement et techniquement les centres régionaux de transfusion sanguine établis dans le pays;
4. Organiser et coordonner la transfusion sanguine, de garantir le système qualité, d'en assurer le contrôle et de veiller à la sécurité des produits sanguins et de leurs dérivés ;
5. Organiser la collecte, l'analyse biologique, la conservation et la distribution du sang et de ses dérivés sur le territoire national ;
6. Appuyer techniquement les services hospitaliers de transfusion en assurant la logistique en matériel et en réactifs spécifiques à la transfusion sanguine ainsi que la formation et le perfectionnement de leur personnel ;
7. Assurer la formation du partenariat avec les structures ayant le même objectif ;

8. Exercer toute autre activité permettant au Centre la réalisation de sa mission.

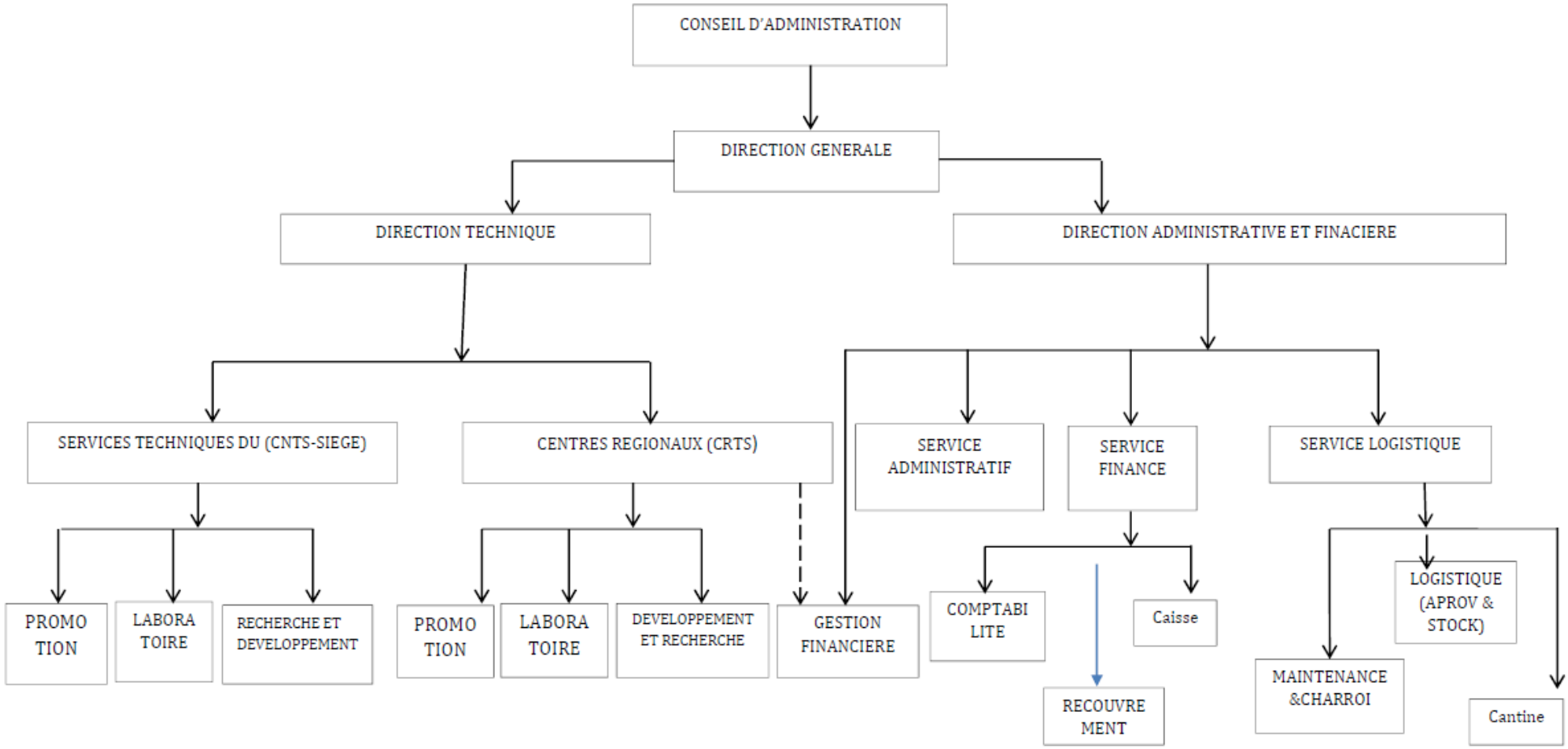


Figure 1: Organigramme du CNTS

I.2. Présentation du projet de recherche

I.2.1 Analyse de l'existant

Les visites et les interviews conduites au CNTS nous ont permis de constater l'existence d'un service de Laboratoire, de Recherche et développement et de distribution des produits sanguins, qui possède des ordinateurs tournant sous Windows avec les logiciels de la suite Microsoft Office. Chacun de ces ordinateurs possèdent une imprimante.

Le CNTS couvre tout le territoire national et particulièrement la Mairie et les provinces BUJUMBURA, MURAMVYA et MWARO. Il est relayé par 14 banques de sang. Le Centre a 4 centres régionaux de transfusion sanguine à savoir GITEGA, NGOZI, BURURI et CIBITOKE. Ces centres régionaux ont comme rôle de faire la distribution comme suit :

Pour le CRTS de NGOZI ; il a un petit local hébergé par l'hôpital de NGOZI, couvre les provinces NGOZI, KAYANZA, MUYINGA et KIRUNDO et est relayé par 2 postes de transfusion sanguine et 12 banques de sang.

Pour le CRTS GITEGA, il a un petit local hébergé par l'hôpital Régionale de GITEGA, couvre les provinces Gitega, RUYIGI, CANKUZO et KARUSI et est relayé par 16 banques de sang.

Pour le CRTS BURURI, il possède son propre bâtiment, couvre les provinces BURURI, MAKAMBA, RUTANA et RUMONGE et est relayé par 10 banques de sang.

Pour le CRTS CIBITOKE, il possède son propre bâtiment dont la moitié est occupé par l'Hôpital CIBITOKE, couvre les provinces CIBITOKE et BUBANZA et est relayé par 6 banques de sang.

La distribution des produits sanguins s'effectue selon les procédures de distribution décrites ci-après.

I.2.2. Procédure de distribution des produits sanguins.

1. **Objectif**

Cette procédure vise à répondre avec satisfaction aux besoins des patients en état d'anémies, d'hémorragies d'origines diverses. Les produits sanguins sont mis en circulation par ordre de dates de prélèvement pour éviter leur destruction pour des raisons de péremption.

2. Domaine d'application

Ladite distribution est réalisée par le personnel affecté au service d'Immuno-hématologie (les techniciens médicaux ou laborantins) appelés à assurer les gardes.

Il faut qu'il y ait une demande de sang soit sous forme :

- De réquisition par une banque de sang signé par le médecin directeur ;
- De réquisition par une demande nominative signée par un médecin.

Ces réquisitions sont faites sur deux formulaires (fiches) différents :

- **Une fiche de réquisition des banques de sang** qui comprend :
    1. La provenance de la réquisition ;
    2. Le nombre de poches demandé ;
    3. Le nom et le titre de celui qui a fait la réquisition ;
    4. Les différents groupes sanguins demandés ;
    5. Le nombre de trousses de transfusion demandé ;
    6. Le nombre de poches de transfert demandé ;
    7. La nature du sang réquisitionné ;
- **Une fiche de réquisition individuelle nominative** qui comprend :
    1. La provenance de la réquisition ;
    2. Le nom de la personne à transfusée ;
    3. L'âge et le sexe de la personne à transfusée ;
    4. Le poids si c'est un enfant ;
    5. Le service demandeur ;
    6. Le motif de transfusion ;
    7. Le nom du médecin demandeur ;
    8. La quantité demandée ;
    9. Si le receveur a déjà été transfusé ou non ;
    10. La demande doit préciser le sang à distribuer (du sang total, du culot globulaire, du plasma, du sang frais).

3. Protocole de distribution des produits sanguins

Les produits sanguins exempts d'infections transmissibles par voie transfusionnelle sont distribués en deux rubriques :

1. Distribution du sang après le test de compatibilité
2. Distribution du sang sans test de compatibilité

3.1. Distribution du sang après le test de compatibilité

Protocole :

- ✓ Déterminer le groupe sanguin du patient ;
- ✓ Mettre sur la paillasse, le produit sanguin de même groupe que celui du patient ;
- ✓ Préparer les dérivés sanguins dont le patient a besoin ;
- ✓ Effectuer le test de compatibilité ;
- ✓ Si le sang est compatible, procéder à l'enregistrement dans **le registre de livraison de sang** lequel comprend les éléments suivants :
    1. La date de la demande de sang
    2. Le numéro de la demande
    3. Le nom de l'hôpital et le service demandeur
    4. L'heure de la demande
    5. La quantité du produit sollicité
    6. Le nombre d'unités du produit servi et leurs numéros d'ordre
    7. L'heure de livraison et de réception du produit
    8. Le nom et la signature du technicien qui a fait l'épreuve de compatibilité
    9. Nom, fonction et signature du transporteur du produit

- **Le registre des épreuves de compatibilité** comprend les éléments suivants :
    1. La date de livraison du produit
    2. L'âge du receveur
    3. Le sexe du receveur
    4. Le numéro de l'épreuve de compatibilité
    5. Le nom et prénom du receveur
    6. L'hôpital et le service demandeur
    7. Le groupe sanguin du receveur
    8. Le numéro d'ordre du produit sanguin servi
    9. Le groupe sanguin du donneur
    10. Le motif de la demande

11. Le nom du médecin demandeur

- Remplir **les bordereaux d'expédition** :

- Le numéro du bordereau
- La date de livraison du produit et le service dont il provient
- Le service de destination
- Le nom et prénom du transporteur
- Le nombre d'unités de produit sanguin, leurs numéros d'ordre, nombre d'unités de trousses de transfusion, le prix unitaire et montant total ainsi que le nombre des poches de transferts
- Les observations
- La signature du technicien qui assure la livraison de ces produits et celle du transporteur.

3.2. Distribution du sang sans test de compatibilité (réquisitions faites par les hôpitaux ayant des banques de sang)

La quantité et le groupe sanguin des produits à servir sont indiqués par le médecin attaché à l'hôpital.

Matériels:

1. Une fiche de demande de sang
2. Un registre de livraison de sang
3. Un bordereau d'expédition
4. Une feuille de désolation

Protocole proprement-dit :

Remplir le registre de livraison de sang et le bordereau d'expédition comme il a été décrit ci-haut.

NB : Si le produit sanguin sollicité n'est pas disponible dans le stock, il faut remplir une feuille de désolation qui doit comprendre:

1. La date de la demande de sang ;
2. Le groupe sanguin du produit sollicité ;
3. La situation du stock et le motif de n'avoir pas honoré la demande ;
4. Le nom du technicien qui a reçu le bon de demande de sang ainsi que sa signature ;
5. L'hôpital et le service demandeur ;
6. Le nom du transporteur, son titre et sa signature.

I.2.3. Problématique

Ayant montré ci-haut le déroulement du processus de distribution de sang par le CNTS envers les institutions sanitaires, nous avons remarqué des problèmes à savoir:

1. Un mauvais service aux institutions sanitaires : la gestion des commandes et des livraisons et la facturation qui sont manuelle conduit souvent à la lenteur du service ;
2. Absence de la bonne information au bon moment : le gérant met à jour le stock une fois la semaine, d'où il en résulte souvent beaucoup d'incohérence entre les informations dans les fichiers et la réalité. Cette situation peut conduire à des nombreux problèmes entre autres la lenteur du service ou la mauvaise prise de décision ;
3. Mauvaise protection des données : d'une part les données stockées dans les fichiers Excel courent un grand risque d'être supprimées soit par un utilisateur malveillant ou accidentellement, d'autre part il est difficile, voire impossible, de restreindre l'accès à certaines données aux utilisateurs dans un fichier Excel ;
4. Manque d'informations statistiques stratégiques permettant aux décideurs du CNTS de prendre de bonnes décisions et suivre l'évolution.

I.2.4. Critique de l'existant

A partir des interviews, documentation et observation faits au CNTS dans notre travail de recherche, nous avons remarqué que le processus de distribution de sang se réalise d'une façon manuelle. Nous avons remarqué certains défis qui sont :

1. Utilisation des registres et des carnets de reçu, ce qui peuvent causer une lenteur, duplication et incohérence des données;
2. Une mauvaise utilisation des équipements informatiques ;
3. Les données non sécurisé ;

I.2.5. Solutions proposées

1. Créer d'une base de données pour faciliter l'enregistrement et l'accessibilité des données ;
2. Suivre les entrées et sorties de façon efficace ;
3. Faire des demandes de sang selon le type pour l'institution sanitaire
4. Calculer automatique des prix poches de sang et imprimer des factures
5. Tenir à jour automatiquement le stock des poches de sang après chaque livraison ou approvisionnement ou vente;

6. Fournir des rapports des livraisons, commandes et approvisionnement des poches de sang selon le type effectuées quotidiennement et mensuellement;
7. Envoyer des notifications en cas de nouvelle commande, sortie, nouvelle institution, nouvel utilisateur, baisse ou expiration des poches de sang selon le type dans le stock.

I.2.6. Objectif du projet

1. Objectif global

Le présent mémoire a pour objectif global la mise en place d'un système de gestion de la banque de sang dans le MSPLS en vue de parier les problèmes de gestion manuelle et manque d'informations utiles en temps opportun qui peuvent occasionner des pertes en vie humaine et va apporter l'amélioration des outils de gestion, des rapports et des statistiques des poches de sang selon les types au CNTS.

2. Objectifs spécifiques

L'application web d'un système de gestion de la Banque de sang développé dans ce mémoire permettra de répondre et atteindre les objectifs spécifiques suivants :

1. Effectuer des commandes en ligne des institutions sanitaires des poches de sang selon le type auprès de la CNTS ;
2. Automatiser la facturation des clients (institutions sanitaires) améliorant ainsi le service fournit à la clientèle et éliminant les risques des erreurs de calculs ;
3. Mettre à jour le stock après chaque livraison ou réception des poches de sang, et partant, la disponibilité de la bonne information au bon moment ;
4. Suivre le niveau du stock et d'avertir le gestionnaire de la baisse des poches de sang dans le stock et la nécessité d'un éventuel renouvellement du stock ;
5. Garantir que les différents utilisateurs accèdent aux données auxquelles ils ont droit d'accéder, limitant ainsi la fuite ou la perte des informations ;
6. Fournir des rapports statistiques journaliers et mensuels sur les livraisons, les commandes et les approvisionnements des institutions sanitaires.

I.2.7. Intérêt du sujet

Le produit de ce travail sera d'une grande utilité pour plusieurs personnes, chacun en ce qui le concerne.

La gestion quotidienne du CNTS verra également son travail réduit car il aura un contrôle beaucoup plus élargi sur les ventes, les livraisons et les commandes effectuées avec moins d'efforts. Des rapports seront à sa disponibilité au moment voulu.

La Direction du CNTS aura accès à des statistiques qui lui permettront de prendre facilement des décisions à partir des données collectées par le logiciel au fur du temps. Il verra la qualité des services fournis augmentée sans parler des gains économiques qui pourront être réalisés grâce à un bon service et une bonne gestion des entrées/sorties.

Pour les institutions sanitaires, le service offert par le CNTS sera nettement amélioré. Les clients seront désormais servis plus rapidement.

Pour **nous,** nous avons l'occasion de confronter la théorie à la pratique. Il faut désormais faire face à la réalité sur le terrain qui est souvent différente ou ne concorde pas avec ce que l'on imaginait.

Pour **les futurs chercheurs, c**e mémoire est rédigé en pensant aux autres chercheurs. Les futurs finalistes feront sûrement à peu près le même parcours que nous. Ils auront sûrement besoin des ressources, des guides et, des informations identiques à celles dont nous avons besoin actuellement. Ils rencontreront probablement des difficultés comme celles que l'on a connues, c'est pour cela qu'on essaiera de mettre à leurs dispositions les informations dont ils pourraient avoir besoin. Nous espérons ainsi pouvoir les aider à s'orienter dans leurs recherches.

I.2.8. Délimitation du sujet

Les projets informatiques affichent une complexité qui ne se remarque dans aucune autre science de l'ingénierie. Ceci est confirmé par les propos de Kelvin Sullivan du Software Engineering Institute que disait : « certains systèmes informatiques sont devenus plus compliqués que les objets physiques les plus sophistiqués de la planète, tels que la Navette spatiale ». Sur base de ces propos, nous avons opté de nous limiter dans le domaine, dans le temps et dans l'espace pour mener à bien notre projet.

La délimitation dans le domaine de notre travail se limite à la gestion de distribution et approvisionnement des produits sanguins.

La délimitation dans le temps de l'application que nous nous proposons de mettre en place dans le cadre de ce mémoire, tient compte des données et des règles de gestion actuelle du CNTS. Nous essayerons de produire une application qui s'adaptera aux changements, sans toutefois garantir que ces derniers trouvent place dans notre application sans aucune modification.

La délimitation dans l'espace de notre travail se limitera à une solution informatique d'un système de gestion de la Banque de sang, cas du CNTS.

I.2.9. Apports scientifiques et technologiques

Ce travail permettra de fournir des informations utiles aux chercheurs développeurs en fournissant un modèle de développement d'une application web moderne en passant en revue les bonnes pratiques et techniques de programmation. Certes, les sujets abordés dans ce mémoire sont vastes et mériteront chacun un livre à part entier. Nous nous contenterons plutôt d'introduire un sujet et de fournir les pistes vers la documentation pour que celui ou celle qui est intéressé puisse continuer ses recherches.

I.2.10. Méthodologie de recherche et domaine d'application

Dans le cadre de ce mémoire nous avons utilisés les méthodes et techniques. Nous avons eu des interviews avec le personnel du CNTS, des institutions sanitaires publiques et privés pour qu'ils puissent nous expliquer les procédures de vente et de gestion appliquées.

Après, nous avons procédé à une observation attentive de leur mode de fonctionnement, analysé les documents dont le CNTS, des institutions sanitaires publiques et privés se servent pour la conservation des données. Cette observation nous a permis de comprendre leur mode de fonctionnement manuel et de nous imaginer une automatisation sans toutefois changer radicalement leurs habitudes de travail.

Enfin, nous avons consultés différents livres, tutoriels, mémoires ainsi que des sites Web. Au cours du développement, les problèmes surviennent vite, nous avons fait recours aux forums de discussion pour poser nos questions.

I.2.11. Outil Mathématique utilisée

Lors de la réalisation de tel système, il faut faire appel non seulement au domaine informatique mais également aux différentes étapes couvrant le domaine mathématique. Nous avons adopté l'Algèbre relationnelle comme outil mathématique à utiliser dans notre système de gestion de banque de sang.

I.3. Conclusion

Dans ce travail, nous avons fait la présentation du Centre National de Transfusion Sanguine auquel nous avons fait notre travail de recherche. Enfin, nous avons fait la présentation du projet de recherche, et de là nous avons montré l'analyse et critique de l'existant, les problématiques et les solutions proposées.

Dans le chapitre suivant, nous allons passer à la conception et modélisation du nouveau système de gestion de la Banque de sang.

CHAPITRE II. CONCEPTION ET MODELISATION DU NOUVEAU SYSTEME AVEC UML

II.1. Présentation du Langage UML

II.1.1. Introduction

Le génie logiciel et la méthodologie s'efforcent de couvrir tous les aspects de la vie du logiciel. Issus de l'expérience des développeurs, concepteurs et chefs de projets, ils sont en constante évolution, parallèlement à l'évolution des techniques informatiques et du savoir-faire des équipes. Comme toutes les tentatives de mise à plat d'une expérience et d'un savoir-faire, les méthodologies ont parfois souffert d'une formalisation excessive, imposant aux développeurs des contraintes parfois contre-productives sur leur façon de travailler.

Avec la mise en commun de l'expérience et la maturation des savoir-faire, des méthodes de travail à la fois plus proches de la pratique réelle des experts et moins contraignantes se sont développées. UML, qui se veut un instrument de capitalisation des savoir-faire puisqu'il propose un langage qui soit commun à tous les experts du logiciel, va dans le sens de cet assouplissement des contraintes méthodologiques [20].

UML est un langage de modélisation graphique à base de pictogrammes. Il est apparu dans le monde du génie logiciel, dans le cadre de la « conception orientée objet ». Il est l'accomplissement de la fusion des précédents langages de modélisation objet. Dans les années 90, un certain nombre de méthodes orientées objets ont émergé, en particulier les méthodes OMT de James RUMBAUGH, BOOCH de Grady BOOCH et OOSE d'Ivar JACOBSON à qui l'on doit les cas d'utilisations [2].

UML est à présent un standard défini par l'Object Management Group (OMG). La dernière version diffusée par l'OMG est UML 2.4.1 depuis août 2011

UML sert principalement à :

1. Décomposer le processus de développement ;
2. Mettre en relation les experts métiers et les analystes ;
3. Coordonner les équipes d'analyse et de conception ;
4. Séparer l'analyse de la réalisation ;
5. Prendre en compte l'évolution de l'analyse et du développement ;
6. Migrer facilement vers une architecture objet d'un point de vue statique et dynamique.

UML est vaste, et plusieurs ouvrages consacrent plusieurs centaines de page pour couvrir tous les notions de ce langage. Sans prétendre être exhaustif dans ce travail, nous estimons qu'il est important de présenter quelques diagrammes que nous avons utilisés.

II.1.2. Présentation des diagrammes UML

UML dans sa version 2 propose treize diagrammes qui peuvent être utilisés pour la description d'un système. Ces diagrammes sont regroupés en deux grands ensembles dont: [2]

Les diagrammes structurels : ayant vocation de représenter l'aspect statique d'un système. Ils permettent d'identifier les objets constituant le programme, leurs attributs, leurs opérations et les méthodes qui leurs sont associés. Ils sont au nombre de six à savoir : le diagramme de Classe, le diagramme d'objet, le diagramme de composant, le diagramme de déploiement, le diagramme de paquetage et le diagramme de structure composite.

Les diagrammes de comportement : ils représentent la partie dynamique d'un système réagissant aux événements et permettant de produire les résultats attendus par les utilisateurs. Sept diagrammes sont proposés par UML :le diagramme des cas d'utilisation, le diagramme d'état-transition, le diagramme d'activités, le diagramme de séquence, le diagramme de communication, le diagramme global d'interaction, et le diagramme de temps.

II.1.3. Les diagrammes structurels

1. Diagramme de classe

Le diagramme de classe constitue l'un des pivots essentiels de la modélisation avec UML. En effet, ce diagramme permet de donner la représentation statique du système à développer. Cette représentation est centrée sur les concepts de classe et d'association. Chaque classe se décrit par les données et les traitements dont elle est responsable pour elle-même et vis-à-vis des autres classes. Les traitements sont matérialisés par des opérations. [2]

La description du diagramme de classe est fondée sur le concept d'objet, le concept de classe comprenant les attributs et les opérations et les différents types d'association entre classes.

Un objet est un concept, une abstraction ou une chose qui a un sens dans le contexte du système à modéliser. Chaque objet a une identité et peut être distingué des autres sans considérer à priori les valeurs de ses propriétés. Il est caractérisé par les valeurs de ses propriétés qui lui confèrent des états significatifs suivant les instants considérés.

Une classe décrit un groupe d'objets ayant les mêmes propriétés (attributs), un même comportement (opérations), et une sémantique commune (domaine de définition).Un objet est une instance d'une classe. La classe représente l'abstraction de ses objets. Au niveau de l'implémentation, c'est-à-dire au cours de l'exécution d'un programme, l'identificateur d'un objet correspond une adresse mémoire.

Une classe est représentée par un rectangle comportant plusieurs compartiments : Les trois compartiments de base sont la désignation de la classe, la description des attributs, la description des opérations.

Deux autres compartiments peuvent être aussi indiqués comme la description des responsabilités de la classe et la description des exceptions traitées par la classe.

Pour la visibilité, chaque attribut ou opération d'une classe peut être de type public, protégé, privé ou paquetage. Les symboles + (public), # (protégé), - (privé) et ~ (paquetage) sont indiqués devant chaque attribut ou opération pour signifier le type de visibilité autorisé pour les autres classes.

Un attribut est une propriété élémentaire d'une classe. Pour chaque objet d'une classe, l'attribut prend une valeur (sauf le cas d'attributs multi valués).Le nom de la classe peut être qualifié par un « stéréotype ». La description complète des attributs d'une classe comporte un certain nombre de caractéristiques qui doivent respecter le formalisme suivant :

Visibilité/Nom attribut : type [= valeur initiale {propriétés}]

Un attribut peut avoir des valeurs multiples. Dans ce cas, cette caractéristique est indiquée après le nom de l'attribut (ex. : prénom pour une personne qui peut avoir trois prénoms).[3]

Une opération est une fonction applicable aux objets d'une classe. Une opération permet de décrire le comportement d'un objet. Une méthode est l'implémentation d'une opération. La signature d'une méthode correspond au nom de la méthode et la liste des paramètres en entrée.

La description complète des opérations d'une classe doit respecter le formalisme suivant :

Visibilité Nom d'opération (paramètres) [:[type résultat] {propriétés}]

Un lien est une connexion physique ou conceptuelle entre instances de classes donc entre objets. Une association décrit un groupe de liens ayant une même structure et une même sémantique. Un lien est une instance d'une association. Chaque association peut être identifiée par son nom. Une association entre classes représente les liens qui existent entre les instances de ces classes.

La multiplicité indique un domaine de valeurs pour préciser le nombre d'instance d'une classe vis-à-vis d'une autre classe pour une association donnée. La multiplicité peut aussi être utilisée pour d'autres usages comme par exemple un attribut multi valué. Le domaine de valeurs est décrit selon plusieurs formes : l'intervalle fermé (2, 3 ... 15), les valeurs exactes (3, 5, 8), et la valeur indéterminée notée * (1..*).

La navigabilité indique si l'association fonctionne de manière unidirectionnelle ou bidirectionnelle, elle est matérialisée par une ou deux extrémités fléchées. La non-navigabilité se représente par un « X ». Par défaut, on admet qu'une navigabilité non définie correspond à une navigabilité implicite.

D'autres propriétés particulières comme contraintes sont proposées dans UML pour préciser la sémantique d'une association. Pour une association de multiplicité supérieure à 1, les liens peuvent être : ordonnés ou non ordonnés (valeur par défaut).

La généralisation est la relation entre une classe et deux autres classes ou plus partageant un sous-ensemble commun d'attributs et/ou d'opérations.

La classe qui est affinée s'appelle superclasse, les classes affinées s'appellent sous-classes. L'opération qui consiste à créer une superclasse à partir de classes s'appelle généralisation. Inversement la spécialisation consiste à créer des sous-classes à partir d'une classe.

2. Diagramme de déploiement

Le diagramme de déploiement permet de représenter l'architecture physique supportant l'exploitation du système. Cette architecture comprend des nœuds correspondant aux supports physiques (serveurs, routeurs...) ainsi que la répartition des artefacts logiciels (bibliothèques, exécutables...) sur ces nœuds.

Un nœud correspond à une ressource matérielle de traitement sur laquelle des artefacts seront mis en œuvre pour l'exploitation du système. Les nœuds peuvent être interconnectés pour former un réseau d'éléments physiques. Un nœud ou une instance de nœud se représente par un cube ou parallélépipède.

Un artefact est la spécification d'un élément physique qui est utilisé ou produit par le processus de développement du logiciel ou par le déploiement du système. C'est donc un élément concret comme, par exemple, un fichier, un exécutable ou une table d'une base de données.

Une spécification de déploiement peut être associée à chaque artefact. Elle permet de préciser les conditions de déploiement de l'artefact sur le nœud sur lequel il va être implanté.

II.1.4 Les diagrammes de comportements

1. Diagramme de cas d'utilisation

Les cas d'utilisation constituent un moyen de recueillir et de décrire les besoins des acteurs du système. Ils peuvent être aussi utilisés ensuite comme moyen d'organisation du développement du logiciel, notamment pour la structuration et le déroulement des tests du logiciel**.** Un cas d'utilisation permet de décrire l'interaction entre les acteurs (utilisateurs du cas) et le système. La description de l'interaction est réalisée suivant le point de vue de l'utilisateur. La représentation d'un cas d'utilisation met en jeu trois concepts : l'acteur, le cas d'utilisation et l'interaction entre l'acteur et le cas d'utilisation.

Un acteur est un utilisateur type qui a toujours le même comportement vis-à-vis d'un cas d'utilisation. Ainsi les utilisateurs d'un système appartiennent à une ou plusieurs classes d'acteurs selon les rôles qu'ils tiennent par rapport au système. Un acteur peut aussi être un système externe avec lequel le cas d'utilisation va interagir. Un acteur peut se représenter symboliquement par un « bonhomme » et être identifié par son nom.

Un cas d'utilisation correspond à un certain nombre d'actions que le système devra exécuter en réponse à un besoin d'un acteur et doit produire un résultat observable pour un ou plusieurs acteurs ou parties prenantes du système.

Une interaction permet de décrire les échanges entre un acteur et un cas d'utilisation.

Afin d'optimiser la formalisation des besoins en ayant recours notamment à la réutilisation de cas d'utilisation, trois relations peuvent être décrites entre cas d'utilisation : une relation d'inclusion (« include »), une relation d'extension (« extend ») et une relation de généralisation.

2. Le diagramme d'activité

Le diagramme d'activité présente un certain nombre de points communs avec le diagramme d'état-transition puisqu'il concerne le comportement interne des opérations ou des cas d'utilisation. Cependant le comportement visé ici s'applique aux flots de contrôle et aux flots de données propres à un ensemble d'activités et non plus relativement à une seule classe.

Les concepts du diagramme d'activité sont :

- transition ;
-  nœud initial (état initial) ;
-  nœud final (état final) ;
- nœud de fin flot (état de sortie) ;
- nœud de décision (choix).

Les concepts spécifiques au diagramme d'activité sont le nœud de bifurcation, le nœud de jonction, le nœud de test-décision, le nœud de fusion-test, le pin d'entrée et de sortie, le nœud d'objet et la partition.[3]

3. Diagrammes de séquence

L'objectif du diagramme de séquence est de représenter les interactions entre objets en indiquant la chronologie des échanges. Cette représentation peut se réaliser par cas d'utilisation en considérant les différents scénarios associés.

Une ligne de vie représente l'ensemble des opérations exécutées par un objet. Un message reçu par un objet déclenche l'exécution d'une opération. Le retour d'information peut être implicite (cas général) ou explicite à l'aide d'un message de retour.

Dans un diagramme de séquence, deux types de messages (synchrone et asynchrone).

Un message synchrone, l'émetteur reste en attente de la réponse à son message avant de poursuivre ses actions. La flèche avec extrémité pleine symbolise ce type de message. Le message retour peut ne pas être représenté car il est inclus dans la fin d'exécution de l'opération de l'objet destinataire du message.

Un message asynchrone, l'émetteur n'attend pas la réponse à son message, il poursuit l'exécution de ses opérations. C'est une flèche avec une extrémité non pleine qui symbolise ce type de message.

UML possède plus de diagramme que celles présenté ci-haut. Nous ne pouvons pas les présenter tous dans ce mémoire, de peur que cela soit long et que beaucoup de livre explique déjà toutes ses notions de façon claire avec des exemples à l'appui. Pour plus d'information, nous conseillons la consultation des livres UML, Analyse et Conception[4] et UML 2 pour les développeurs[5] qui ont été notre principale source d'inspiration dans cette partie.

II.2. Modélisation du nouveau Système de gestion de la Banque de sang

Dans cette partie, nous mettons en pratique la théorie UML annoncé plus haut, en modélisant notre nouveau système. Il existe plusieurs logiciels permettant de modéliser en UML. Nous avons choisi starUML et Visual Paradigm.

II.2.1. Démarche de développement

UML n'est qu'un langage de modélisation et non une méthode. Nous n'avons, jusqu'aujourd'hui, pas une démarche unifiée mettant en œuvre UML pour construire les modèles ou conduire un projet. Cependant les auteurs d'UML, ont décrit, dans un ouvrage « *Le Processus unifié de développement logiciel* »[4], le processus unifié (Unified Process) qui doit être associé à UML.

Il existe deux principaux processus de développement objet qui ont été associés à UML à savoir l'UP (Unified Process) et le RUP (Rational Unified Process). Dans le cadre de ce mémoire, nous n'allons pas donner une présentation détaillée d'UP. Cependant, il nous est paru intéressant de dégager les idées fondatrices d'U.P dans le cadre d'une présentation générale.

II.2.2. Les principes d'UP

Le processus de développement UP, associé à UML, met en œuvre les principes suivants :

Pour le processus guidé par les cas d'utilisation, l'orientation forte donnée par UP est de montrer que le système à construire se définit d'abord avec les utilisateurs. Les cas d'utilisation permettent d'exprimer les interactions du système avec les utilisateurs, donc de capturer les besoins. Une seconde orientation est de montrer comment les cas d'utilisation constituent un vecteur structurant pour le développement et les tests du système. Ainsi le développement peut se décomposer par cas d'utilisation et la réception du logiciel sera, elle aussi, articulée par cas d'utilisation.

Pour le processus itératif et incrémental, **c**e type de démarche étant relativement connu dans l'approche objet, il paraît naturel que U.P préconise l'utilisation du principe de développement par itérations successives. Concrètement, la réalisation de maquette et prototype constitue la réponse pratique à ce principe. Le développement progressif, par incrément, est aussi recommandé en s'appuyant sur la décomposition du système en cas d'utilisation. Les avantages du développement itératif se caractérisent comme suit :les risques sont évalués et traités au fur et à mesure des itérations, les premières itérations permettent d'avoir un feed-back des utilisateurs, les tests et l'intégration se font de manière continue, les avancées sont évaluées au

fur et à mesure de l'implémentation.

Pour le processus centré sur l'architecture, les auteurs d'UP mettent en avant la préoccupation de l'architecture du système dès le début des travaux d'analyse et de conception. Il est important de définir le plutôt possible, même à grandes mailles, l'architecture type qui sera retenue pour le développement, l'implémentation et ensuite le déploiement du système. Le vecteur des cas d'utilisation peut aussi être utilisé pour la description de l'architecture.

Pour le processus orienté par la réduction des risques, l'analyse des risques doit être présentée à tous les stades de développement d'un système. Il est important de bien évaluer les risques des développements afin d'aider à la bonne prise de décision. Du fait de l'application du processus itératif, UP contribue à la diminution des risques au fur et à mesure du déroulement des itérations successives.

Ces principes sont à la base du processus unifié décrit par les auteurs d'UML. Ce sont ces mêmes principes qui nous ont guidés dans le processus de conception de notre application.

II.2.3. Diagramme de cas d'utilisation

Conformément aux processus de développement Unifié U.P, la première étape dans l'élaboration de nos différents diagrammes UML commence par l'élaboration du diagramme de cas d'utilisation. Ce dernier va nous guider dans l'élaboration des autres diagrammes.

Les différents acteurs intervenant dans notre système sont :

1. Le responsable de l'Institution sanitaire : il (elle) va faire les commandes selon le type de demande que son institution a l'autorisation de faire et voir l'historique des commandes déjà effectuées ;
2. Le responsable du CNTS : Il a le rôle de faire l'approvisionnement des produits sanguins, gérer les commandes faites par les hôpitaux/centre de santé, gestion des utilisateurs de ces derniers et la facturation.
3. Le Directeur du CNTS : Il va visualiser les rapports selon les commandes et approvisionnements des produits sanguins, gérer tous les utilisateurs et il a tous les droits de paramétrage du système.

Nous allons faire la représentation du diagramme de cas d'utilisation en 2 figures qui sont :

- Acteur : Responsable d'une institution sanitaire

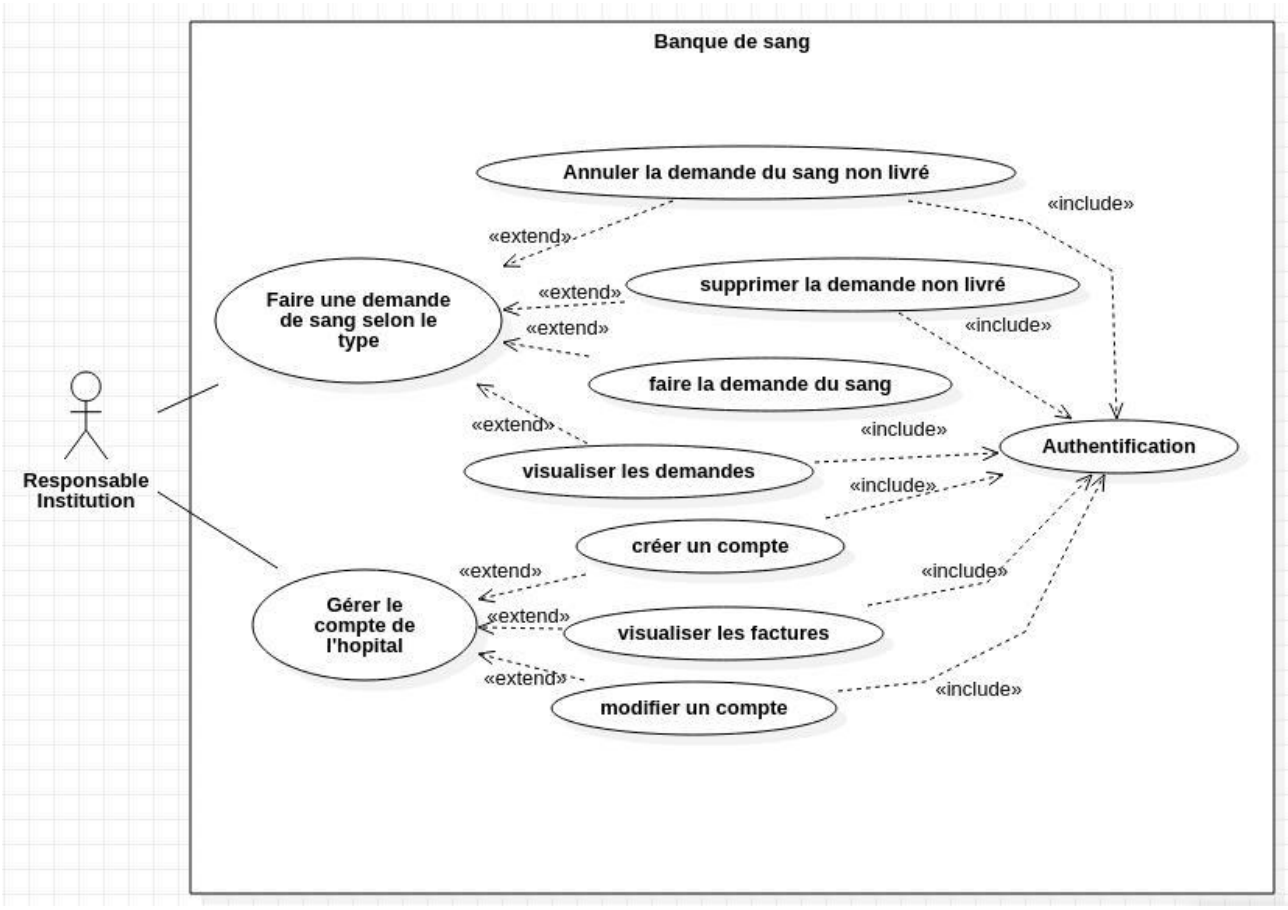


Figure 2: Diagramme de cas d'utilisation pour le Responsable d'une Institution sanitaire

Cette figure ci-dessous représente le diagramme de cas d'utilisation pour le responsable d'une institution sanitaire qui a deux principaux rôles qui est de gérer le compte de l'institution sanitaire auquel il appartient et de faire une demande de sang selon le type de demande.

- Acteurs : Responsables du CNTS et le DG

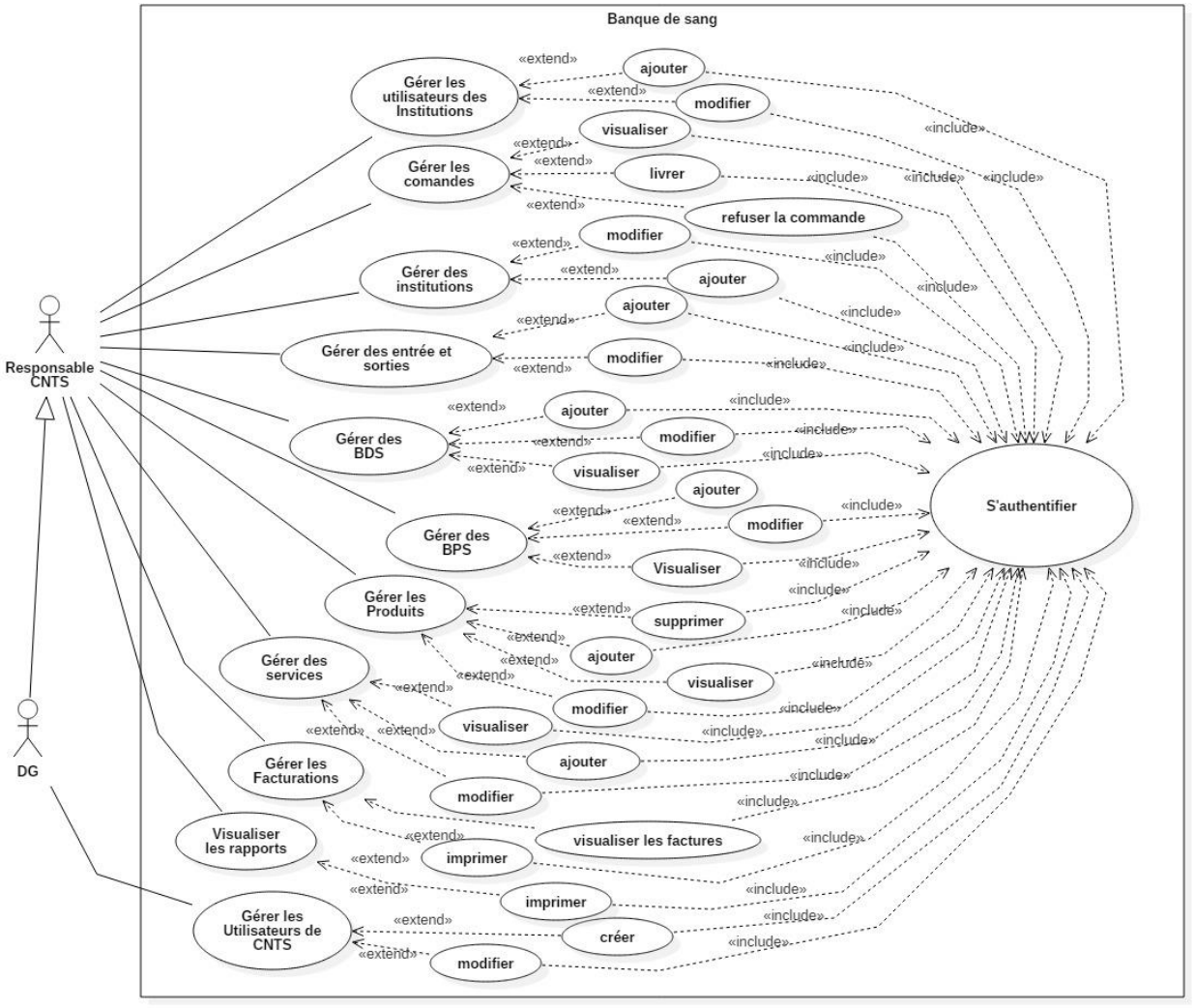


Figure 3: Diagramme de cas d'utilisation pour le Responsable du CNTS et DG

Cette figure ci-dessous représente le diagramme de cas d'utilisation pour le responsable du CNTS qui a le rôle de gérer les utilisateurs des institutions, les commandes, les institutions, les entrées, les sorties, les BPS, les BDS, les produits, les services, les facturations, et visualiser les rapports ; et le DG qui a le rôle de gérer les utilisateurs du CNTS et faire tous ce que le responsable du CNTS fait.

Les scénarios

- **Cas d'utilisation : s'authentifier**

  A. Sommaire

  - **Titre :** identification
  - **Résumé :** Le serveur étant allumé, un utilisateur appelle la page d'accueil et clique sur connexion et le système la lui affiche.
  - **Acteur :**Responsable Institution/Responsable CNTS/ Directeur CNTS

  B. Description des scenarios

  - **Pré-condition :** le système est lancé, l'utilisateur possède les bons identifiants
  - **Scenario Nominal**

Acteur	**Système**
1. L'utilisateur démarre l'application	2. Le système affiche la page d'accueil
3. L'utilisateur clique sur connexion	4. Le système lui affiche la page d'identification.
5. L'utilisateur saisit son identifiant et son mot de passe	6. Le système affiche l'interface selon le profil choisi

Tableau 1: Scenario nominal pour le cas d'utilisation :« s'authentifier »

  - **Enchainement alternatif**

    Si le nom d'utilisateur et le mot de passe ne sont pas corrects, l'enchainement alternatif redémarre à l'étape 4 du scenario nominal.

- **Cas d'utilisation : Faire une demande de sang**

  A. Sommaire

  - **Titre :** Faire une demande de sang selon le type
  - **Résumé :** Faire la demande de sang nominative ou demande de banque de sang, compléter le formulaire et valider.

- **Acteur :** Responsable Institution

B. Description des scenarios

- **Pré-condition :** L'utilisateur avec le profil d'un responsable d'une institution, bien identifié.
- **Scenario Nominal**

Acteur	**Système**
1. L'utilisateur choisit le type de demande	2. Le système affiche les différentes commandes
3. L'utilisateur clique sur nouvelle commande	4. Le système lui affiche le formulaire
5. L'utilisateur complète le formulaire et valide	6. Le système affiche les détails de la commande

Tableau 2: Scenario nominal pour le cas d'utilisation: « faire demande de sang »

- **Enchainement alternatif**

  Si l'utilisateur ne complète pas bien les données du formulaire, l'enchainement alternatif recommence sur l'étape 4 du scenario nominal.

- **Cas d'utilisation : Gérer les institutions**

A. Sommaire

- **Titre :**Gérer les institutions sanitaires
- **Résumé :**Enregistrer les institutions sanitaires selon leur BPS, BDS et leur attribuer le type de demande de sang.
- **Acteur :** Responsable CNTS

B. Description des scenarios

- **Pré-condition :** L'utilisateur avec le profil d'un responsable du CNTS, bien identifié.
- **Scenario Nominal**

Acteur	**Système**
1. L'utilisateur choisit dans paramètre	2. Le système affiche les différents paramètres.
3. L'utilisateur clique sur Institution	4. Le système lui affiche la liste de l'institution enregistrée
5. L'utilisateur clique sur ajouter	6. Le système affiche le formulaire
7. L'utilisateur complète le formulaire et valide	8. Le système lui affiche la page où l'institution est enregistrée

Tableau 3: Scenario nominal pour le cas d'utilisation : « Gérer les institutions »

- **Enchainement alternatif**

  Si l'utilisateur ne complète pas bien les données du formulaire, l'enchainement alternatif recommence sur l'étape 6 du scenario nominal.

II.2.4. Diagramme d'activité

Les diagrammes de cas d'utilisation ayant été établis, les diagrammes d'activités nous ont permis de décrire beaucoup plus en détails le flux des activités pour chaque cas d'utilisation.

Nous allons vous présenter 3 cas :

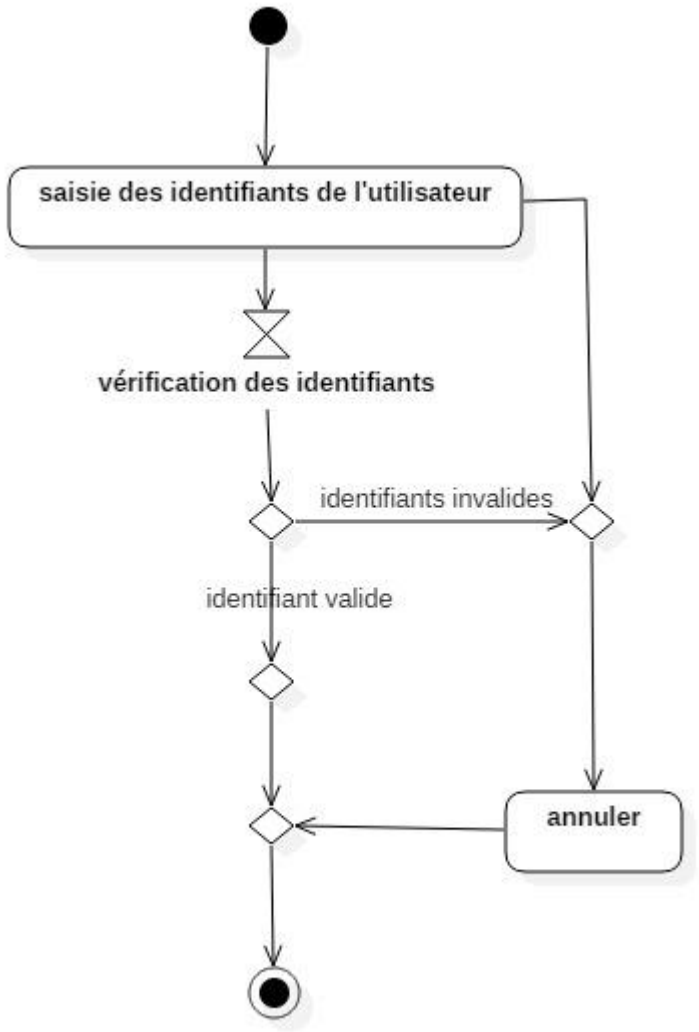


Figure 4:Diagramme d'activité pour l'authentification

Cette figure ci-dessus représente les différents flux d'activités pour qu'un utilisateur puisse s'authentifier. L'utilisateur complète ses identifiants et si c'est valide, il est redirigé sur une page selon son profil.

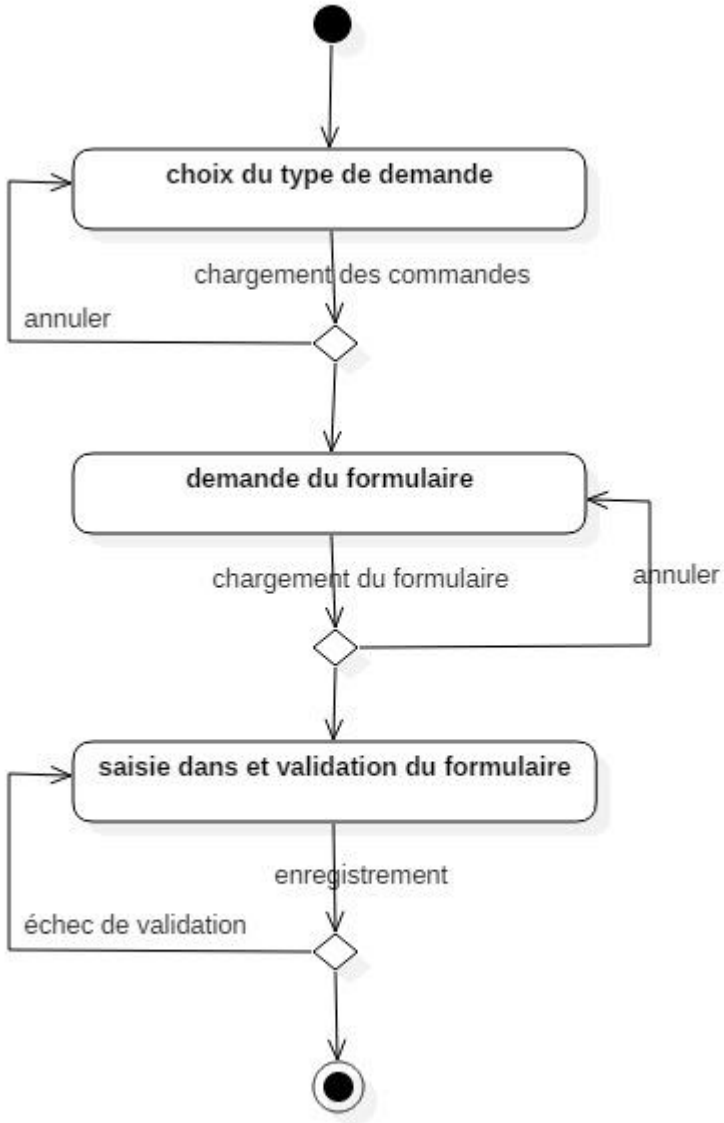


Figure 5: Diagramme d'activité pour faire la demande de sang

Cette figure ci-dessus représente les différents flux d'activités qu'une institution sanitaire fait pour passer la demande de sang selon le type de demande qui lui est octroyé par le CNTS.

Après avoir fait le choix du type de demande, le responsable complète le formulaire, puis il valide et la commande est envoyée.

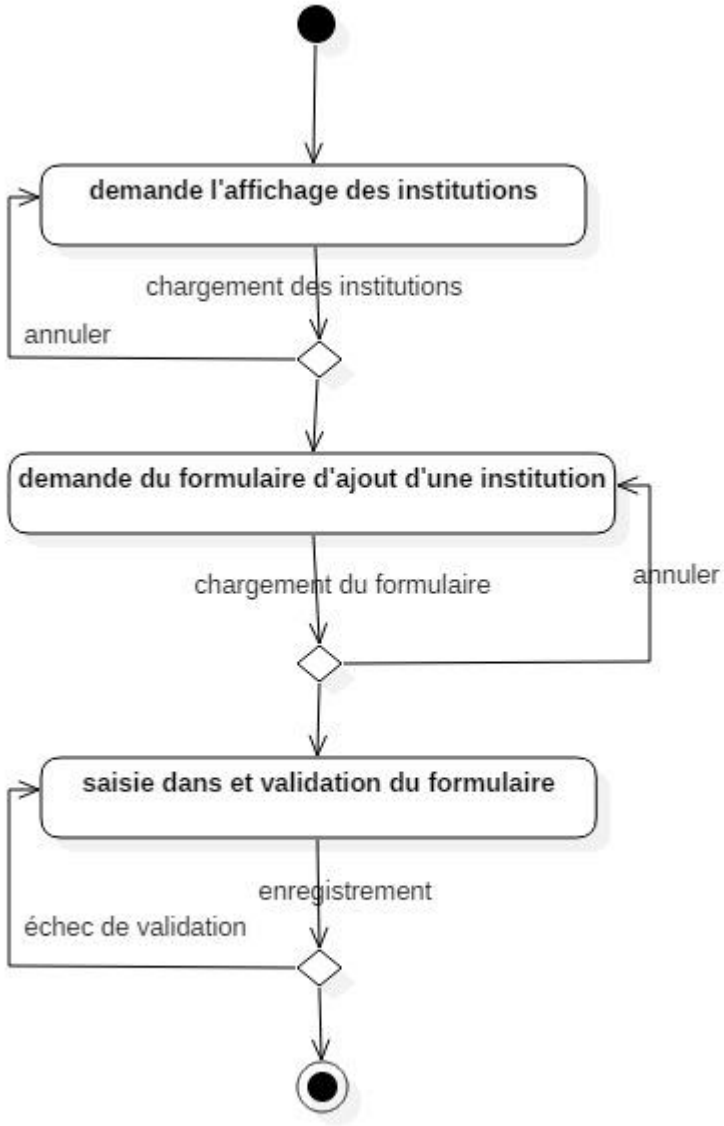


Figure 6: Diagramme d'activité pour l'enregistrement d'une institution sanitaire

Cette figure ci-dessus représente les différents flux d'activités que le CNTS fait pour enregistrer une institution sanitaire. Le Responsable du CNTS affiche les institutions et clique sur le formulaire d'ajout de l'institution, puis complète le formulaire et valide.

II.2.5 Diagramme de séquence

Nous allons vous présenter 3 cas :

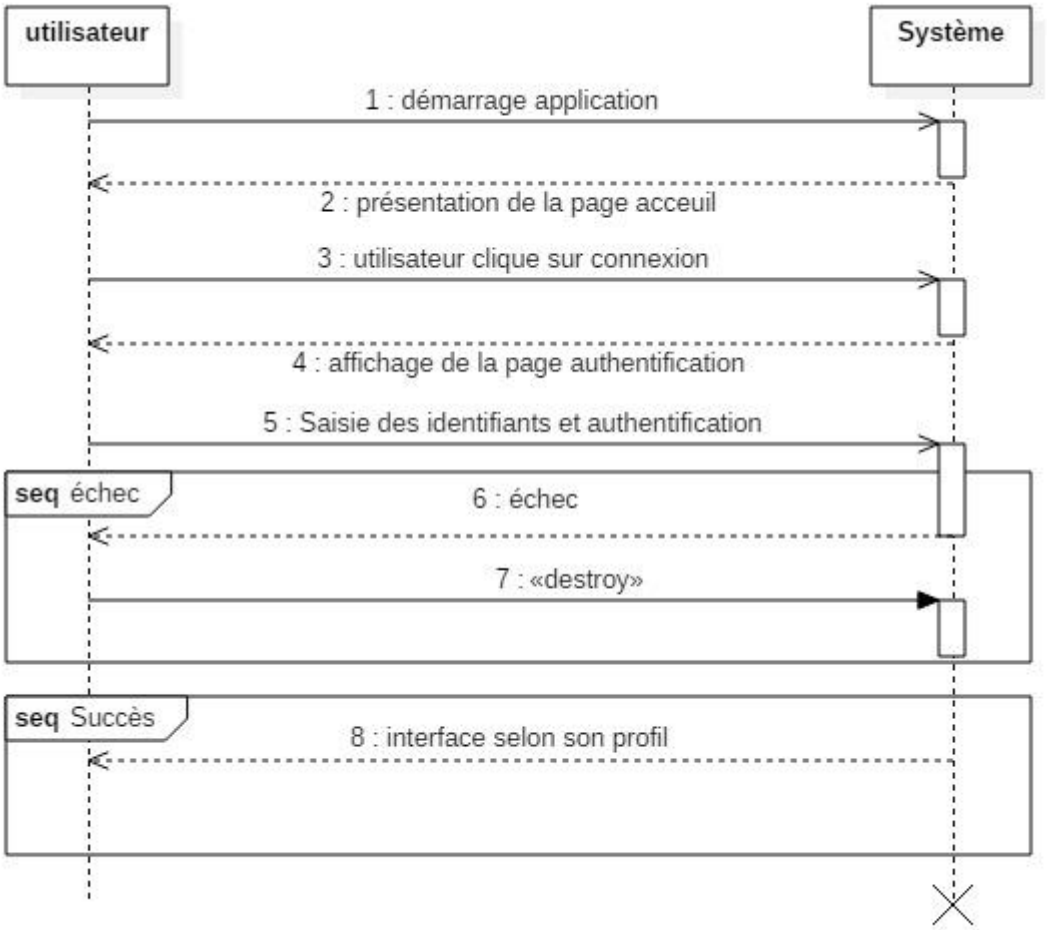


Figure 7: Diagramme de séquence pour l'authentification

Cette figure ci-dessus représente les différentes interactions entre l'utilisateur et le système pour que l'authentification se passe. L'utilisateur met ses identifiants et le système lui redirige vers une interface qui est liée à son profil.

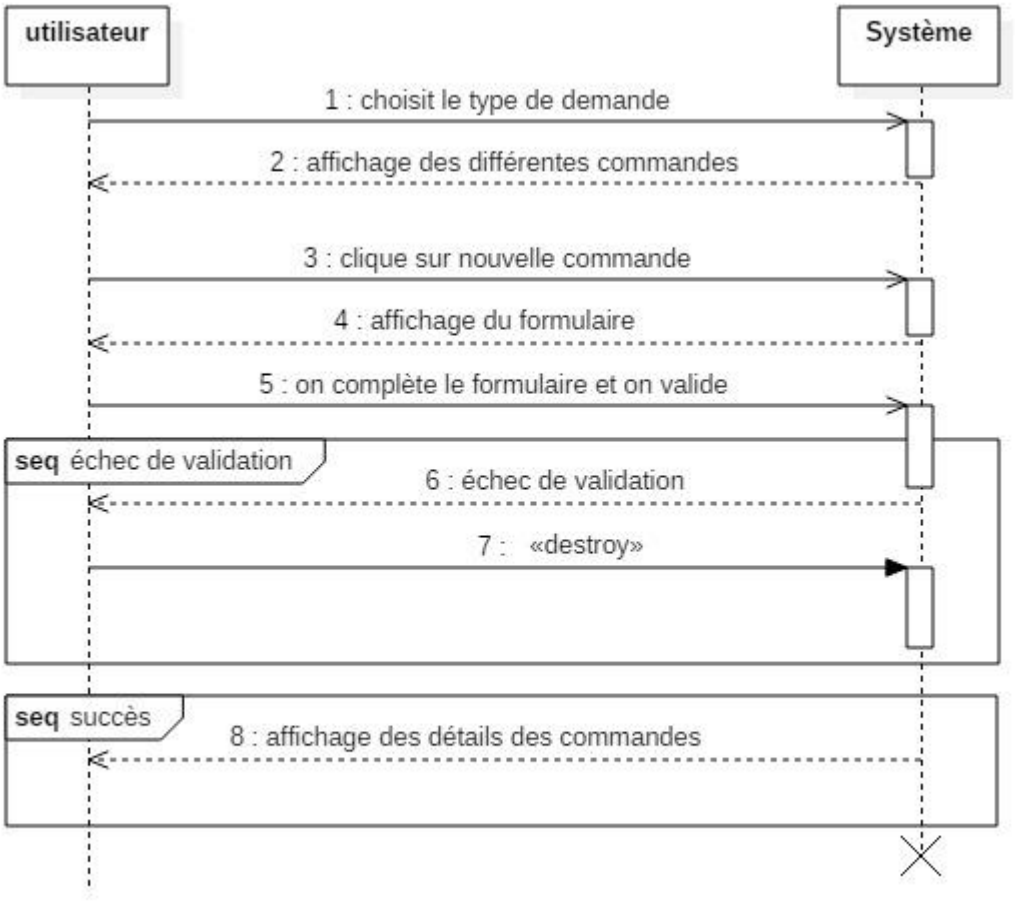


Figure 8: Diagramme de séquence pour la demande de sang

Cette figure ci-dessus représente les différentes interactions entre l'utilisateur et le système pour la demande de sang selon le type de demande. Cette séquence est pour le responsable d'une institution sanitaire qui fait le choix du type demande, complète le formulaire puis valide et sa demande est envoyée s'il a bien complété le formulaire.

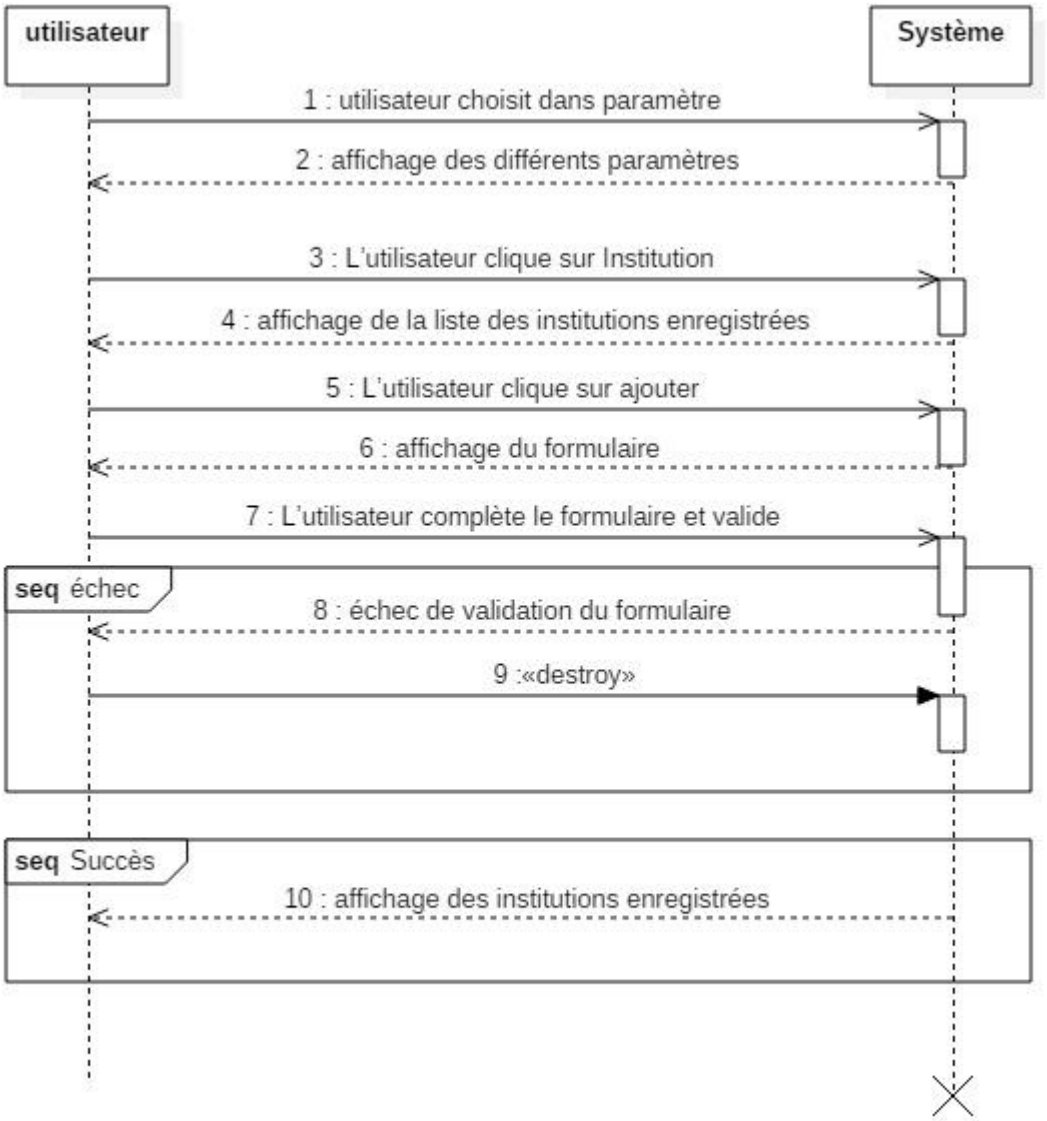


Figure 9:Diagramme de séquence pour l'enregistrement d'une institution sanitaire

Cette figure ci-dessus représente les différentes interactions entre l'utilisateur et le système pour l'enregistrement d'une institution sanitaire. La séquence ci-dessus est pour le Responsable du CNTS qui se dirige dans Paramètres, affiche les institutions puis ajouter une institution, puis complète le formulaire et valide.

II.2.6. Diagramme de classe

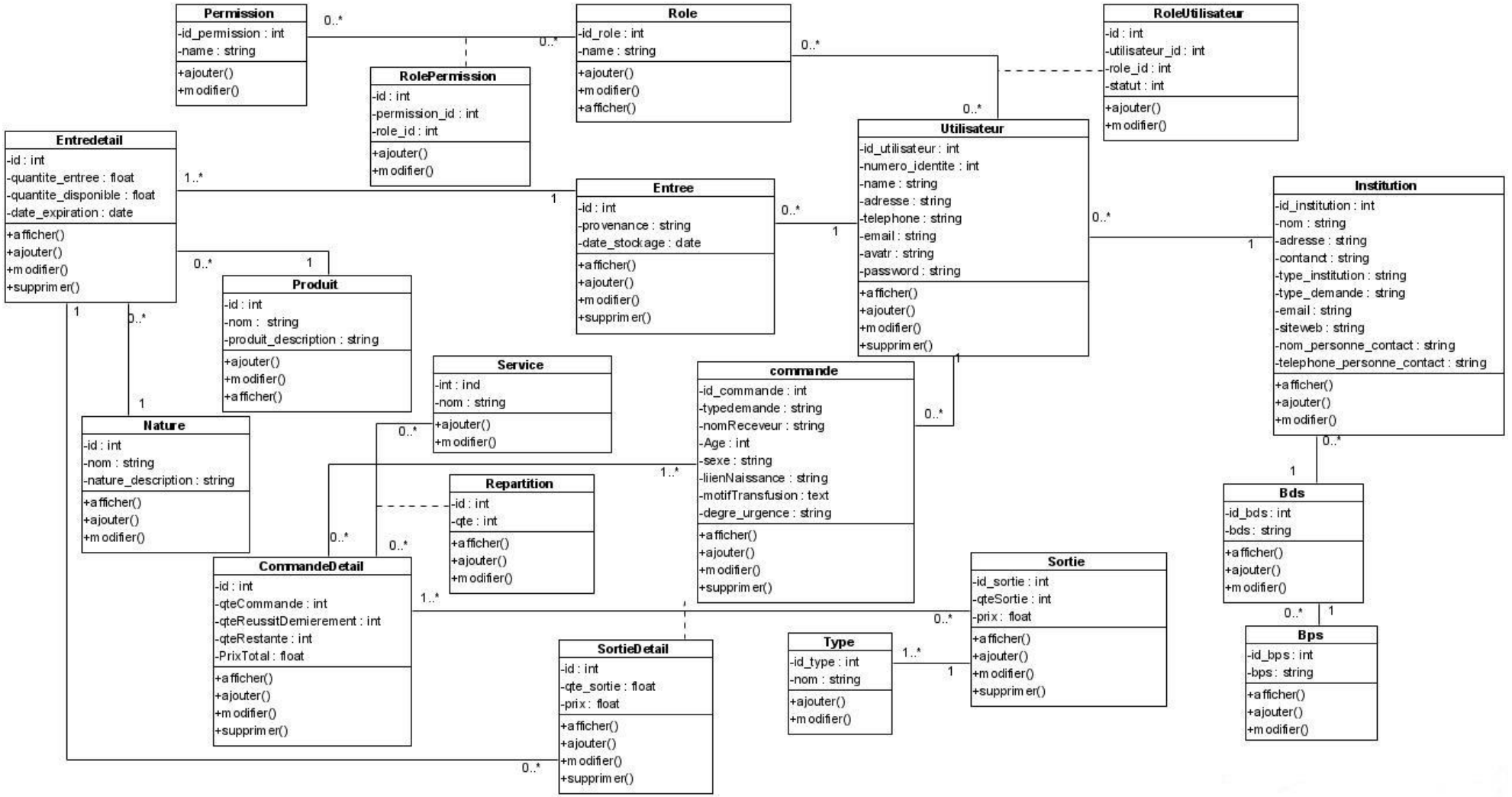


Figure 10: Diagramme de classe

II.2.7. Diagramme de déploiement

Enfin, nous terminons notre modélisation avec le diagramme de déploiement qui décrit l'architecture physique de notre application.

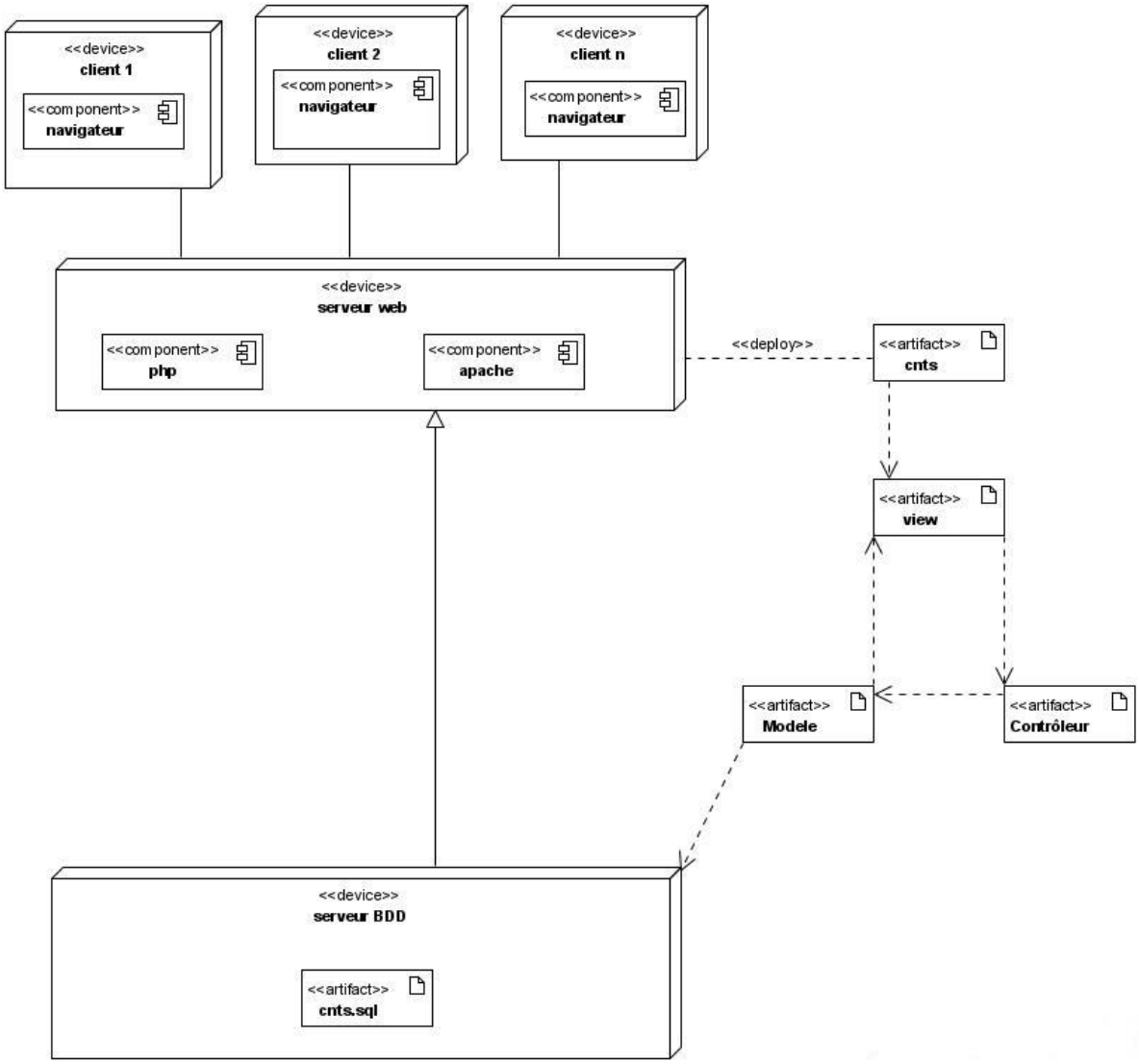


Figure 11: Diagramme de déploiement

II.3. Conclusion

Dans ce chapitre, nous avons fait la conception et modélisation du nouveau système où en premier lieu nous avons parlé de la théorie du langage UML tandis que en second lieu, nous modélisons le nouveau système grâce à ce langage.

Dans le chapitre, nous allons modéliser le nouveau système avec le formalisme mathématique de l'Algèbre Relationnelle.

CHAPITRE III. MODELISATION DU NOUVEAU SYSTEME AVEC LE FORMALISM.E MATHEMATIQUE DE L'ALGEBRE RELATIONNELLE

III.1. Introduction

L'algèbre relationnelle est une collection d'opérateurs permettant de réaliser des opérations sur des relations. Elle permet par exemple de sélectionner certains enregistrements d'une relation satisfaisant une condition ou encore de regrouper des enregistrements de relations différentes.

Le résultat de l'application d'un opérateur sur une ou deux relations est une nouvelle relation qu'on appelle fermeture. Elle implique notamment qu'il n'y a pas de doublons dans le résultat et permet l'écriture d'expressions de calcul.

L'algèbre relationnelle utilise les opérateurs classiques de manipulation des ensembles qui sont : [6] l'union, l'intersection, la différence et le produit cartésien

Il introduit aussi des opérateurs propres aux bases de données qui sont : la sélection, la projection, la jointure et la division.

Ces opérateurs sont soit unaires soit binaires. Les opérateurs unaires impliquent un seul opérande: sélection (noté σ), projection (∏), renommage(α). Les opérateurs binaires impliquent deux opérandes: produit cartésien (X), jointures (∞), union (∪), intersection (∩), différence (-), division (/).

III.2. Opérateurs ensemblistes

Les opérateurs ensemblistes correspondent aux opérateurs habituels de la théorie des ensembles. Ces opérateurs sont l'union, l'intersection, la différence et le produit cartésien.[6]

III.2.1. Opérateur union (∪)

Deux relations sont dites union-compatibles si elles ont le même schéma de relation, c'est-à-dire qu'elles ont le même nombre d'attributs et que ceux-ci ont le même domaine.L'union des deux relations R **υ**S produit une nouvelle relation de schéma identique à R et à S possédant les enregistrements appartenant à R ou à S ou aux deux relations.

- Syntaxe : R∪ **S**
- Sémantique : réunit dans une même relation les tuples de R et ceux de S (sans doublons)

III.2.2. Opérateur intersection (∩)

L'intersection des deux relations R∩S produit une nouvelle relation de schéma identique à R et à S possédant les enregistrements appartenant conjointement à R et à S.

- Syntaxe : R ∩S
- Sémantique : sélectionne les tuples qui sont à la fois dans R et S

III.2.3. Opérateur différence (-)

La différence des deux relations R-S produit une nouvelle relation de schéma identique à R ou à S possédant les enregistrements présents dans R mais pas dans S.

- Syntaxe : R -S
- Sémantique : sélectionne les tuples de R qui ne sont pas dans S

III.2.4. Opérateur produit cartésien (x)

Le produit cartésien des deux relations R x S produit une nouvelle relation de schéma Z égal à l'union des schémas R et S et possédant comme enregistrements, la concaténation des enregistrements de R avec ceux de S.

- Syntaxe : R x S
- Sémantique : chaque tuple de R est combiné avec chaque tuple de S

III.3. Opérateurs propres aux Bases de données

III.3.1. Renommage α

Le renommage ou l'affectation permet de renommer les attributs d'une relation pour résoudre des problèmes de compatibilité entre noms d'attributs de deux relations opérandes d'une opération binaire.

- Syntaxe : α[ancien_nom : nouveau_nom] R
- Sémantique : les tuples de R avec un nouveau nom de l'attribut

III.3.2. Sélection σ

La sélection appelée encore restriction est un opérateur unaire qui prend en entrée une relation r de schéma R et produit en sortie une nouvelle relation de même schéma R ayant comme enregistrements ceux de r satisfaisant la condition de sélection. Le but étant de sélectionner un ensemble de tuples d'une relation, en fonction d'un critère de sélection (prédicat ou expression logique de prédicats).La condition de sélection utilise les opérateurs de comparaison (=, <, <=, >, >=, !=), les connecteurs logiques (et, ou, non) et les parenthèses.

- Syntaxe: σ[p] R
  p: prédicat de sélection (condition de sélection)
- Sémantique : crée une nouvelle relation de population l'ensemble des tuples de R qui satisfont le prédicat p

III.3.3. Projection ∏

La projection est un opérateur unaire qui prend en entrée une relation r de schéma R (A1; A2; ...; An) et produit en sortie une nouvelle relation de schéma (A1; A2; .. ;Ai; Aj) inclus dans R ayant comme enregistrements ceux de r restreints à ce sous-schéma (A1; A2; .. ;Ai; Aj).Le but de la projection étant de ne retenir que certains attributs dans une relation (c'est-à-dire un ensemble de colonnes). A l'issue d'une projection, la relation résultante peut contenir des doublons.

- Syntaxe: ∏ [attributs] R
  (attributs: liste l'ensemble des attributs de R à conserver dans le résultat.)
- Notation fonctionnelle : R{liste d'attributs}
- Sémantique : crée une nouvelle relation de population l'ensemble des tuples de Rréduits aux seuls attributs de la liste spécifiée

III.3.4. Division (/)

Le résultat de la division d'une relation R(X, Y) par une relation S (Y) est une relation Q(X)définie par :

1) le schéma de Q est constitué de tous les attributs de R n'appartenant pas à S.

2) les tuples qj de Q tels que, quels que soit les tuples si de S, le tuple (qj,si) est un tuplede R (c'est-à-dire QXS⊆R).

- Syntaxe: R / S

- Sémantique : crée une nouvelle relation de population des tuples dont la concaténation avec tous les n-uplets de S appartiennent à R.

III.3.5. Jointure ∞

Soient r et s deux relations de schémas respectifs R et S. La jointure de R et S, selon une condition que doivent vérifier les valeurs des tuples, est l'ensemble des tuplesdu produit cartésien R x S satisfaisant cette condition. Donc on peut la considérer comme un produit cartésien suivi d'une sélection. La relation résultant de la jointure possède comme schéma l'union des deux schémas R et S et comme enregistrements la concaténation des enregistrements de R avec ceux de S qui répondent à la condition de sélection. La condition de sélection utilise les opérateurs de comparaison (=;<; <=; >; >=;!=), les connecteurs logiques (et, ou, non) et les parenthèses. Lorsque le critère de sélection est l'égalité, on parle d'équijointure sinon on parle de Thétajointure.

- Syntaxe : R ∞[p]S
  p: prédicat de sélection (condition de jointure)
- Sémantique : combine certains tuples qui répondent à une condition

III.4. Requêtes SQL de notre nouveau système et en Algèbre relationnelle

Vu le nombre suffisant des requêtes que nous avons utilisé, nous allons vous présenter certaines requêtes pour les 2 types d'opérations :

III.4.1. Opérations unaires

1. **Projection :**
   - **Description :**
     - ✓ Projection de tous les natures de la table natures
     - ✓ Projection des quantités entrées, quantités disponible et date d'expiration du stock (entreedetails).
   - **Forme Algébrique :**
     - ✓ $\prod_{(nature)} natures$
     - ✓ $\prod_{(\text{quantite}-\text{entree},\text{quantite}-\text{disponible},\text{date}-\text{expiration})} \text{entreedetails}$
   - **Requête SQL :**
     - ✓ SELECT DISTINCT nature FROM natures

nature
Sang total
Culot globulaire
Plasma frais congelé
concentré des plaquettes

- ✓ SELECT DISTINCTquantite_entree,quantite_disponible,date_expiration FROM entreedetails ;

quantite_entree	quantite_disponible	date_expiration
20.00	20.00	2021-05-30
55.00	55.00	2021-06-05
120.00	120.00	2021-06-04
120.00	75.00	2021-10-21
130.00	130.00	2021-10-12
200.00	200.00	2021-10-12
340.00	340.00	2021-10-12
120.00	120.00	2021-10-12
140.00	140.00	2021-10-12
160.00	160.00	2021-10-12

2. **Sélection :**
   - **Description :**
     - ✓ Sélection des BDS qui ont comme BPS BUJUMBURA MAIRIE qui a comme identifiant numéro1
     - ✓ Sélection des institutions qui ont comme type institution Hôpital.
   - **Forme Algébrique :**
     - ✓ $\sigma_{(\text{bps}-\text{id}=1)} bds$
     - ✓ $\sigma_{(\text{type}-\text{institution}=\text{"Hopital"})} institutions$
     - ✓
   - **Requête SQL :**
     - ✓ SELECT * FROM bds WHERE bps_id=1;

id	bps_id	bds	bds_status	created_at	updated_at
1	1	BUJUMBURA CENTRE	1	2021-05-09 17:56:24	2021-05-10 09:38:14
2	1	BUJUMBURA NORD	1	2021-05-10 09:38:41	2021-05-10 09:38:41
3	1	BUJUMBURA SUD	1	2021-05-10 09:38:55	2021-05-10 09:38:55

- ✓ SELECT * FROM ‘institutions’ WHERE type_institution=”Hopital”;

id	bds_id	institution_status	type_institution	type_demande	institution	adresse	telephone
1	3	1	Hopital	Requisition	kira	bujumbura	69123476
2	1	1	Hopital	Requisition	CPLR	ROHERO	23907790
3	2	1	Hopital	Requisition	HMK	KAMENGE	234567

III.4.2. Opérations binaires et n-aires

3. Union :

- **Description :** Sélection des quantités entrées, quantités disponible et date d’expiration des produits de nature sang total qui a comme identifiant numéro 1 et qui ont comme provenance ULT qui a comme identifiant numéro 2.
- **Forme Algébrique :**
  - ✓ $\prod_{(\mathbf{quantite-entree,quantite-disponible,date-expiration})}(\boldsymbol{\sigma}_{(\mathbf{nature-id=1})}\ \mathbf{entreedetails})$

$$\cup$$

$$\prod_{(\mathbf{quantite-entree,quantite-disponible,date-expiration})}(\boldsymbol{\sigma}_{(\mathbf{entree-id=2})}\ \mathbf{entreedetails})$$

- **RequêteSQL :**
  - ✓ SELECT distinct quantite_entree,quantite_disponible, date_expiration FROM entreedetails WHERE nature_id=1
    UNION
    SELECT quantite_entree,quantite_disponible, date_expiration FROM entreedetails WHERE entree_id=2;

quantite_entree	quantite_disponible	date_expiration
55.00	55.00	2021-06-05
120.00	75.00	2021-10-21
130.00	130.00	2021-10-12
200.00	200.00	2021-10-12
340.00	340.00	2021-10-12
120.00	120.00	2021-10-12
140.00	140.00	2021-10-12
160.00	160.00	2021-10-12

4. **Intersection :**
   - **Description :**Sélection des quantités entrées ; quantités disponible des produits qui commencent par A et qui comme nature identifié par le numéro 1.
   - **Forme Algébrique :**

$$\prod_{(prouit-id,nature-id,quantite-entree,quantite-disponible)} (\sigma_{(nature-id=1)}\ entreedetails)$$

$$\cap$$

$$\prod_{(produit)} (\sigma_{(produitlikeA\%etproduits.id=entreedetails.produit-id)}\ produits$$

   - **Requête SQL :**
     - ✓ SELECT distinct produit_id,nature_id,quantite_entree,quantite_disponible, produits.produit FROM entreedetails INNER JOIN produits WHERE entreedetails.nature_id=1 AND produits.produit LIKE "A%" AND produits.id=entreedetails.produit_id;

produit_id	nature_id	quantite_entree	quantite_disponible	produit
1	1	220.00	218.00	A+
3	1	250.00	250.00	AB+
5	1	180.00	180.00	A-
7	1	160.00	160.00	AB-

5. **Produit cartésien**

- ✓ **Description :** Sélection de tous les BPS et BDS
- ✓ **Forme Algébrique :** $\boldsymbol{bds} \times \boldsymbol{bps}$
- ✓ **Requête SQL :**

  Select * from bds,bps ;

id	bps	bps_status	created_at	updated_at	id	bps_id	bds	bds_status	created_at	updated_at
1	BUJUMBURA MAIRIE	1	2021-05-09 17:56:08	2021-05-10 09:36:03	1	1	BUJUMBURA CENTRE	1	2021-05-09 17:56:24	2021-05-10 09:38:14
2	BUBANZA	1	2021-05-14 07:15:39	2021-05-14 07:15:39	1	1	BUJUMBURA CENTRE	1	2021-05-09 17:56:24	2021-05-10 09:38:14
3	BUJUMBURA	1	2021-05-14 07:17:49	2021-05-14 07:17:49	1	1	BUJUMBURA CENTRE	1	2021-05-09 17:56:24	2021-05-10 09:38:14
4	GITEGA	1	2021-05-14 08:23:52	2021-05-14 08:23:52	1	1	BUJUMBURA CENTRE	1	2021-05-09 17:56:24	2021-05-10 09:38:14
5	NGOZI	1	2021-05-14 10:07:19	2021-05-14 10:07:19	1	1	BUJUMBURA CENTRE	1	2021-05-09 17:56:24	2021-05-10 09:38:14
6	KAYANZA	1	2021-05-14 10:07:26	2021-05-14 10:07:26	1	1	BUJUMBURA CENTRE	1	2021-05-09 17:56:24	2021-05-10 09:38:14
1	BUJUMBURA MAIRIE	1	2021-05-09 17:56:08	2021-05-10 09:36:03	2	1	BUJUMBURA NORD	1	2021-05-10 09:38:41	2021-05-10 09:38:41
2	BUBANZA	1	2021-05-14 07:15:39	2021-05-14 07:15:39	2	1	BUJUMBURA NORD	1	2021-05-10 09:38:41	2021-05-10 09:38:41
3	BUJUMBURA	1	2021-05-14 07:17:49	2021-05-14 07:17:49	2	1	BUJUMBURA NORD	1	2021-05-10 09:38:41	2021-05-10 09:38:41
4	GITEGA	1	2021-05-14 08:23:52	2021-05-14 08:23:52	2	1	BUJUMBURA NORD	1	2021-05-10 09:38:41	2021-05-10 09:38:41
5	NGOZI	1	2021-05-14 10:07:19	2021-05-14 10:07:19	2	1	BUJUMBURA NORD	1	2021-05-10 09:38:41	2021-05-10 09:38:41
6	KAYANZA	1	2021-05-14 10:07:26	2021-05-14 10:07:26	2	1	BUJUMBURA NORD	1	2021-05-10 09:38:41	2021-05-10 09:38:41
1	BUJUMBURA MAIRIE	1	2021-05-09 17:56:08	2021-05-10 09:36:03	3	1	BUJUMBURA SUD	1	2021-05-10 09:38:55	2021-05-10 09:38:55
2	BUBANZA	1	2021-05-14 07:15:39	2021-05-14 07:15:39	3	1	BUJUMBURA SUD	1	2021-05-10 09:38:55	2021-05-10 09:38:55
3	BUJUMBURA	1	2021-05-14 07:17:49	2021-05-14 07:17:49	3	1	BUJUMBURA SUD	1	2021-05-10 09:38:55	2021-05-10 09:38:55
4	GITEGA	1	2021-05-14 08:23:52	2021-05-14 08:23:52	3	1	BUJUMBURA SUD	1	2021-05-10 09:38:55	2021-05-10 09:38:55
5	NGOZI	1	2021-05-14 10:07:19	2021-05-14 10:07:19	3	1	BUJUMBURA SUD	1	2021-05-10 09:38:55	2021-05-10 09:38:55
6	KAYANZA	1	2021-05-14 10:07:26	2021-05-14 10:07:26	3	1	BUJUMBURA SUD	1	2021-05-10 09:38:55	2021-05-10 09:38:55
1	BUJUMBURA MAIRIE	1	2021-05-09 17:56:08	2021-05-10 09:36:03	4	2	BUBANZA	1	2021-05-14 07:16:16	2021-05-14 07:16:16
2	BUBANZA	1	2021-05-14 07:15:39	2021-05-14 07:15:39	4	2	BUBANZA	1	2021-05-14 07:16:16	2021-05-14 07:16:16
3	BUJUMBURA	1	2021-05-14 07:17:49	2021-05-14 07:17:49	4	2	BUBANZA	1	2021-05-14 07:16:16	2021-05-14 07:16:16
4	GITEGA	1	2021-05-14 08:23:52	2021-05-14 08:23:52	4	2	BUBANZA	1	2021-05-14 07:16:16	2021-05-14 07:16:16
5	NGOZI	1	2021-05-14 10:07:19	2021-05-14 10:07:19	4	2	BUBANZA	1	2021-05-14 07:16:16	2021-05-14 07:16:16
6	KAYANZA	1	2021-05-14 10:07:26	2021-05-14 10:07:26	4	2	BUBANZA	1	2021-05-14 07:16:16	2021-05-14 07:16:16
1	BUJUMBURA MAIRIE	1	2021-05-09 17:56:08	2021-05-10 09:36:03	5	2	MPANDA	1	2021-05-14 07:16:27	2021-05-14 07:16:27

6. **Thétajointure**

- ✓ **Description :** sélection de tous lesquantitésentrées, quantités disponibles,date d'expiration, identifiants des natures et identifiant de la table entreedetails et la table natures pour les attributs identifiants et nature avec une condition que les identifiants de la table entreedetails soient inferieur aux identfiants de la table natures
- ✓ **Forme Algébrique :**

$$\prod_{(\boldsymbol{quantite-entree,quantite-disponible,date-expiration,nature-id,id})} \boldsymbol{entreedetails}$$

$$^{\infty(\boldsymbol{entreedetails.id}>natures.id \text{ et } natures.id=entreedetails.nature-id)}$$

$$\prod_{(\boldsymbol{id,nature})} (\boldsymbol{natures})$$

- ✓ **Requête SQL :**

  SELECT distinct entreedetails.quantite_entree,entreedetails.quantite_disponible,entreedetails.date_expiration,entreedetails.nature_id,entreedetails.id,natures.id,natures.nature

FROM entreedetails,natures WHERE entreedetails.id>natures.id AND natures.id=entreedetails.nature_id;

quantite_entree	quantite_disponible	date_expiration	nature_id	id	id	nature
200.00	200.00	2021-10-22	1	2	1	Sang total
250.00	250.00	2021-10-22	1	3	1	Sang total
250.00	250.00	2021-10-22	1	4	1	Sang total
180.00	180.00	2021-10-22	1	5	1	Sang total
170.00	170.00	2021-10-22	1	6	1	Sang total
160.00	160.00	2021-10-22	1	7	1	Sang total
170.00	170.00	2021-10-22	1	8	1	Sang total
80.00	80.00	2021-10-22	2	9	2	Culot globulaire
70.00	70.00	2021-12-23	3	10	3	Plasma frais congelé
110.00	110.00	2021-12-24	6	11	6	Concentré des plaquettes

Conclusion

Dans ce chapitre, nous avons parlé du formalisme mathématique de l'algèbre relationnelle où nous avons parlé en premier lieu des opérations ensemblistes et les opérations propres aux Bases de données et en deuxième partie, nous avons fait des illustrations des requêtes SQL de notre nouveau système en Algèbre relationnelle.

Dans le chapitre suivant, nous allons présenter la réalisation de notre application.

CHAPITRE IV : REALISATION DE L'APPLICATION DE GESTION DE LA BANQUE DE SANG

Dans ce chapitre, nous allons montrer pourquoi le choix d'une application web, ensuite les technologies utilisés dans notre application et enfin la présentation des interfaces de notre système.

IV.1. Choix d'une application Web

Le choix de réaliser une application web n'a pas été le fruit du hasard. En effet, les applications web sont beaucoup plus appréciées de nos jours car elles offrent un nombre considérable d'avantages par rapport aux applications traditionnelles Desktop. Cela étant, les applications web présentent aussi des inconvénients.

Analysons d'abord les avantages des applications web avant de mettre en évidence leurs inconvénients.

1. **Portabilité et Accessibilité**: le principal avantage des applications web est qu'elles sont portables. Peu importe le langage avec lequel l'application est conçue, ou encore moins le système d'exploitation sur lequel tourne le serveur, l'application est accessible à partir de n'importe quel client ayant seulement un navigateur web et un accès réseau internet ou intranet. Ce modèle simplifie beaucoup le développement car le développeur n'a pas besoin de se soucier des contraintes liées au système d'exploitation des utilisateurs clients [7]
2. **Un support natif du réseau**: l'application que nous nous proposons de développer devra être un outil de collaboration entre les différents employés du CNTS et des différents hôpitaux. Chacun travaille sur une machine différente mais tous doivent travailler sur les mêmes données. Cela suscite l'intérêt de travailler en réseau. Techniquement, réaliser une application desktop travaillant en réseau nécessite beaucoup de travail qu'une application web car on doit développer d'une part la partie serveur et, d'autre part, la partie client. Pour notre cas le choix est donc clair, c'est l'application web qu'il nous faut [7].
3. **Une installation facile** : la configuration des applications web se fait une fois pour toute sur le serveur. Une fois que l'application est déployée, n'importe qui ayant un navigateur peut accéder et utiliser l'application sans installer de logiciel supplémentaire. Chaque système d'exploitation (Windows, Linux, Mac OS) intègre nativement un navigateur, ce qui fait que, pour utiliser l'application, tout ce dont on a besoin c'est d'être connecté sur le réseau.

4. **La stabilité**: le client accède à l'application via son navigateur. Il n'y a pas d'accès direct au serveur, et vice versa, le serveur n'accède pas directement aux ressources matérielles ou logiciel du client sans autorisation implicite de celui-ci. Il n'y a, par conséquent, pas de risque qu'un utilisateur fasse une mauvaise manipulation (intentionnellement ou non) qui risque de faire planter le serveur [7].

5. **Une mise à jour et maintenance faciles** : le coût de maintenance des applications a toujours été significatif. Pour les applications desktop, on doit appliquer les mises à jour pour tous les clients ce qui devient vite une lourde tâche dans une grande entreprise ayant beaucoup d'ordinateurs. Pour les applications web, quand on effectue une mise à jour, il suffit de l'appliquer au serveur pour que tous les clients en bénéficient [7].

Les avantages des applications web sont nombreux mais les désavantages ne manquent pas : ils sont surtout liés aux contraintes techniques dues au développement des applications web. Nous pouvons citer à titre d'exemple [7] :

- **Une mauvaise expérience des utilisateurs** : les applications Desktop traditionnelles offrent beaucoup plus de confort à l'utilisateur en fournissant des widget (menus, des glisser/déplacer, des menus contextuels, etc.) que les applications web fournissent difficilement. Il existe cependant des techniques pour pallier à ce problème.
- **L'Incompatibilité des Navigateurs** : les applications Web utilisent des navigateurs web pour afficher les informations aux utilisateurs. Il se peut que les utilisateurs aient un navigateur qui n'est pas à jour et provoquent le disfonctionnement de certaines fonctionnalités. Pour cette raison, lors du développement, on doit s'assurer du test de l'application sur différents navigateurs.

Les applications web offrant visiblement beaucoup d'avantages que d'inconvénients, notre choix était vite fait de choisir une application web de gestion de la Banque de sang au CNTS qui va interagir avec les institutions sanitaires.

Dans la suite, nous exposons les outils, technologies et techniques qui nous ont permis de produire l'application web que nous voulons réaliser.

IV.2. Choix des technologies à utiliser

À ce stade du projet, le problème a été identifié et analysé en profondeur, nous avons défini une modélisation appropriée aux besoins de l'application, il reste à concrétiser notre modélisation en la transformant en une application utilisable par les utilisateurs finaux. Pour cela, des choix judicieux des technologies doivent être effectués pour aboutir au produit final. Le choix des technologies à utiliser constitue une étape importante pour mener à bien un projet informatique. En effet, il s'agit de choisir les outils et les technologies qui vont nous permettre de concrétiser nos idées et de réaliser nos objectifs.

Parmi un large choix de technologies informatique disponible sur le marché à l'heure actuel, Jeremy Parr et Rod Shaw[8] admettent également que le choix peut être parfois contraignant. Les besoins spécifiques de l'application, le matériel sur lequel elle tourne, le budget alloué au projet, les connaissances de l'équipe de développement, tels sont les critères qui orientent le choix de ces technologies. Dans la suite nous essayons d'expliquer les raisons qui nous ont poussés à choisir telle ou telle technologie que nous avons utilisées pour réaliser notre application. Le premier critère de sélection nous était déjà imposé avant de commencer le développement. Faute de moyen, toute solution payante était déjà exclu de nos options, nous avons donc utilisé uniquement des technologies gratuites et open source.

Les différentes technologies que nous avons utilisées sont :

1. PATTERN MVC :

Ce modèle est un concept d'architecture qui propose une séparation en trois entités des données, des traitements et de l'interface:[9]

La vue représente ce que l'utilisateur a sous les yeux. La vue peut donc être une application graphique Swing, AWT pour Java, Form pour C# . . . ; une page web ; un terminal Linux ou une console Windows ; etc. La vue reçoit du modèle les données qu'il doit présenter à l'utilisateur par l'intermédiaire d'un contrôleur qui doit d'abord valider les données. Les mêmes données peuvent avoir plusieurs vue pour différents clients (Ex : Navigateur et Smartphones pour les applications web).

Le modèle représente les données traitées par l'application généralement stockées dans une base de données. Le modèle notifie les vues et le contrôleur associé lorsqu'il y a eu un changement de son état. Cette notification permet à la vue de mettre à jour l'interface avec les nouvelles données et le contrôleur de valider ces données.

Un contrôleur assure des échanges entre la vue et le modèle notamment grâce à des composants métiers. Il permet de transformer les commandes provenant de l'interface utilisateur (view) pour les envoyer au modèle ou au sens contraire convertir les données provenant du modèle vers l'interface utilisateur UI. Il sert de pont (interface) de communication entre le modèle et la vue.

Schématiquement ce modèle peut être représenté comme suit :

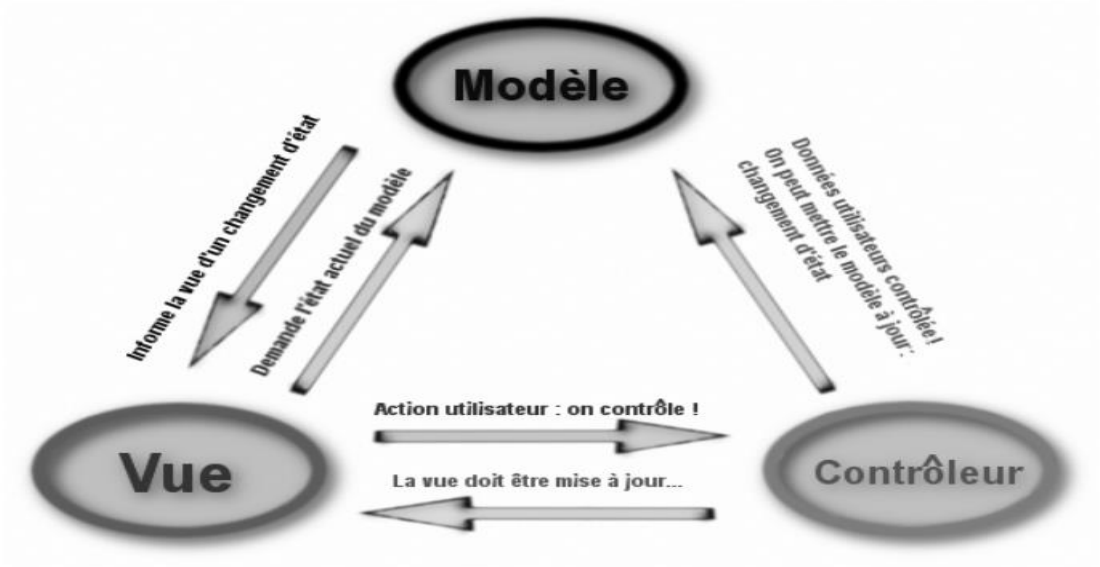


Figure 12: Modèle MVC

2. **PHP** :

Hypertext Preprocessor, plus connu sous son sigle PHP, est un langage de programmation libre, principalement utilisé pour produire des pages Web dynamiques via un serveur HTTP, mais pouvant également fonctionner comme n'importe quel langage interprété de façon locale. PHP est un langage impératif orienté objet.[10]

3. **JAVASCRIPT :**

JavaScript est un langage de programmation de scripts principalement employé dans les pages web interactives et à ce titre est une partie essentielle des applications web. Avec les technologies HTML et CSS, JavaScript est parfois considéré comme l'une des technologies cœur du World Wide Web.[11]

4. (X)HTML :

HTML (HyperText Markup Language) a fait son apparition dès 1991 lors du lancement du Web. Il est le langage de description d'une page Web. Ce langage s'appuie sur un ensemble de balises standards interprétées par le navigateur afin de définir le contenu et la mise en forme de la page. C'est donc en HTML qu'on écrit ce qu'on souhaite que la page affiche : du texte, des liens, des images.... Le XHTML, quant à lui, est une évolution du précédent langage conformément aux contraintes du XML et impose le respect de certaines règles pour qu'une page soit bien formée (noms des balises en minuscule, attributs des balises obligatoirement encadrés par des guillemets, fermeture obligatoire de toutes les balises, …).[12]

5. CSS :

Depuis 1996, le W3C recommande de séparer le contenu de l'application (le fond et la forme). Ceci permet d'utiliser le même style sur plusieurs pages ou de proposer plusieurs styles (apparence) au même contenu. Le CSS permet de décrire précisément la mise en forme d'une page HTML, XHTML ou XML. Ce langage est venu compléter le HTML en 1996. Il est ainsi possible de séparer clairement le contenu d'une page Web (matérialisé par les balises (X) HTML) et sa forme (décrite par la feuille de style CSS). Il gère en général l'apparence de la page web (agencement, positionnement, décoration, couleur, taille du texte...).[12]

6. LARAVEL :

Laravel est un framework web open-source écrit en PHP respectant le principe modèle-vue-contrôleur et entièrement développé en programmation orientée objet. Laravel est distribué sous licence MIT, avec ses sources hébergées sur GitHub.[13]

7. BOOTSTRAP :

Bootstrap est une collection d'outils utiles à la création du design de sites et d'applications web. C'est un ensemble qui contient des codes HTML et CSS, des formulaires, boutons, outils de navigation et autres éléments interactifs, ainsi que des extensions JavaScript en option.[14]

8. MYSQL :

MySQL Database Service est un service de base de données entièrement géré pour déployer des applications natives du cloud en utilisant la base de données open source la plus populaire au monde.[15]

9. Visual Code :

Visual Studio Code est un éditeur de code extensible développé par Microsoft pour Windows, Linux et macOS. Les fonctionnalités incluent la prise en charge du débogage, la mise en évidence de la syntaxe, la complétion intelligente du code, les snippets, la refactorisation du code et Git intégré.[16]

10. XAMPP :

XAMPP est un ensemble de logiciels permettant de mettre en place un serveur Web local, un serveur FTP et un serveur de messagerie électronique. Il s'agit d'une distribution de logiciels libres offrant une bonne souplesse d'utilisation, réputée pour son installation simple et rapide.[17]

IV.3. Présentation de l'application de gestion de la Banque de sang

Le grand moment est venu pour présenter la concrétisation des analyses et des théories énoncées plus haut. Le produit final auquel nous avons abouti à la fin de ce travail est présenté sommairement dans les captures d'écrans ci-dessous.

L'application est divisée en trois parties principales (chaque partie par rôle d'utilisateur) permettant ainsi de fournir à chaque utilisateur le strict nécessaire dont il a besoin pour accomplir son travail. Ainsi les responsables des institutions, les responsables du CNTS et le DG n'accèdent pas aux mêmes fonctionnalités.

Le rôle d'un utilisateur est déterminé par le système d'authentification selon son profil.



Figure 13: Page d'accueil de l'application

IV.3.1. Authentification des utilisateurs

Pour chaque activité de l'application, l'utilisateur doit d'abord s'authentifier grâce à un formulaire sur la figure suivante.

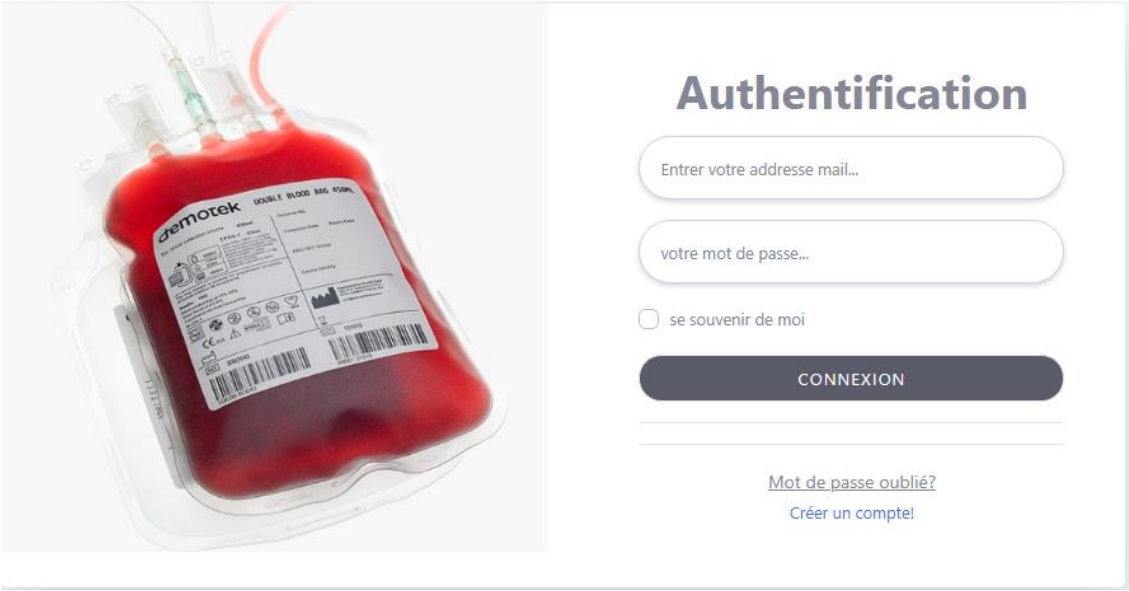


Figure 14: Authentification des utilisateurs

Après authentification, on est redirigé vers la bonne page selon votre profil.

Si l'institution sanitaire n'a pas de compte, il peut ajouter l'institution ou créer le compte de l'institution qui existe.

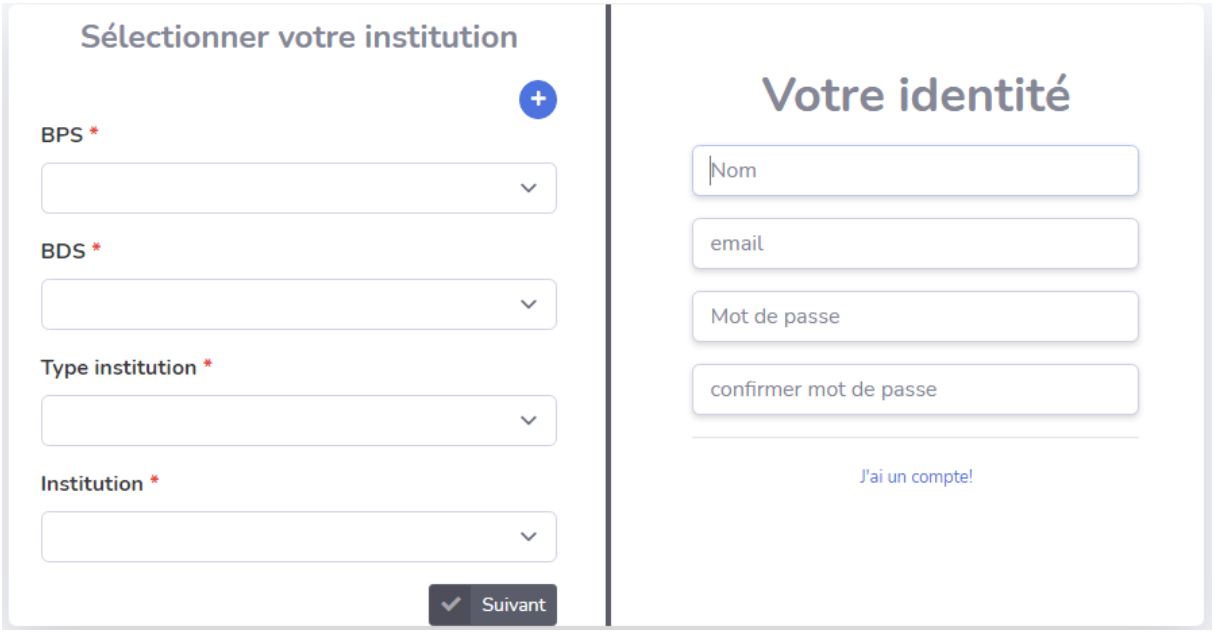


Figure 15: Page pour la création d'un compte d'une Institution

IV.3.2. Responsables des institutions sanitaires

Le Responsable d'une institution sanitaire gère les commandes selon le type de demande, la facturation et les rapports liés au stock et factures.

L'institution ayant déjà mis ses identifiants, il peut choisir de faire la demande nominative qui est liéeà un patient selon le motif qui sera cité par le Docteur.

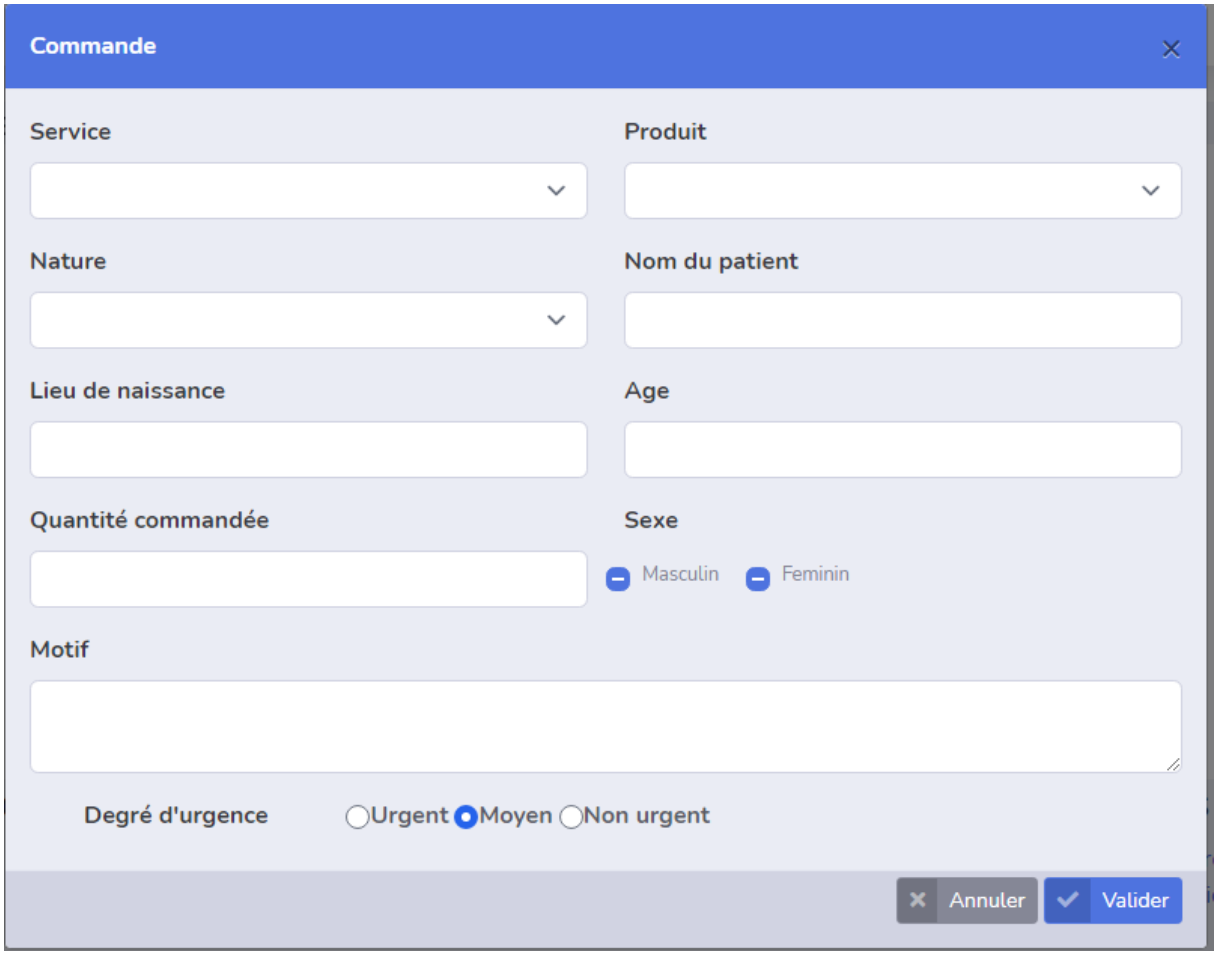


Figure 16: Page de faire la demande nominative

L'institution sanitaire peut aussi faire une demande de banque de sang pour les hôpitaux qui ont accès à faire cette demande. Elle va compléter le formulaire suivant en choisissant le produit, la nature, la quantité de la demande nécessaire.

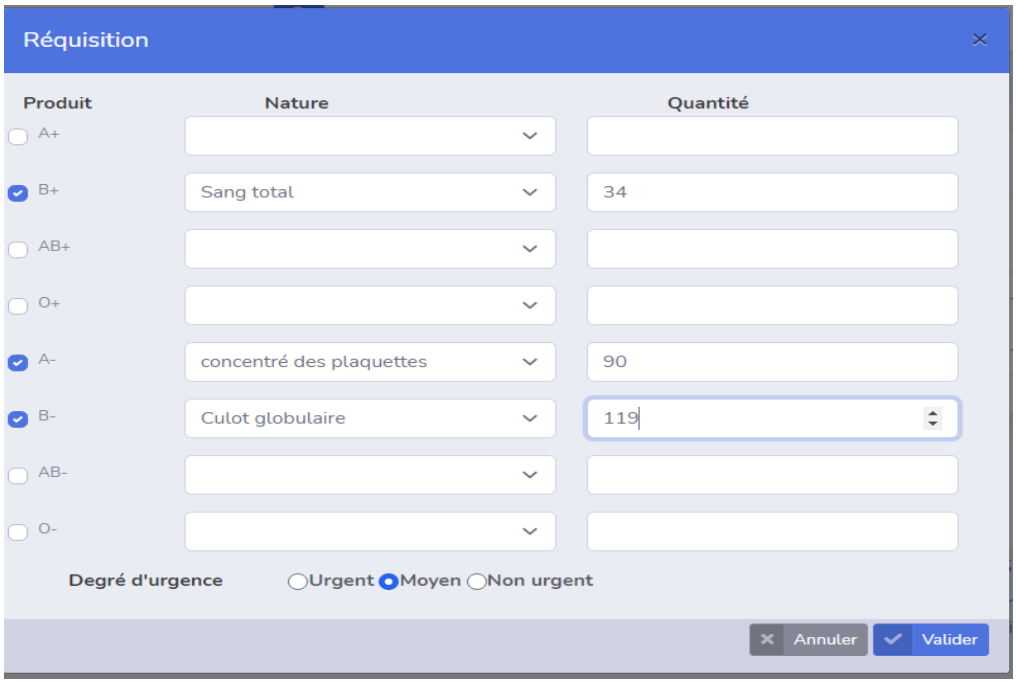


Figure 17: Page pour la demande de banque de sang

L'institution sanitaire peut voir la fiche des demandes déjà servies selon la figure suivante.

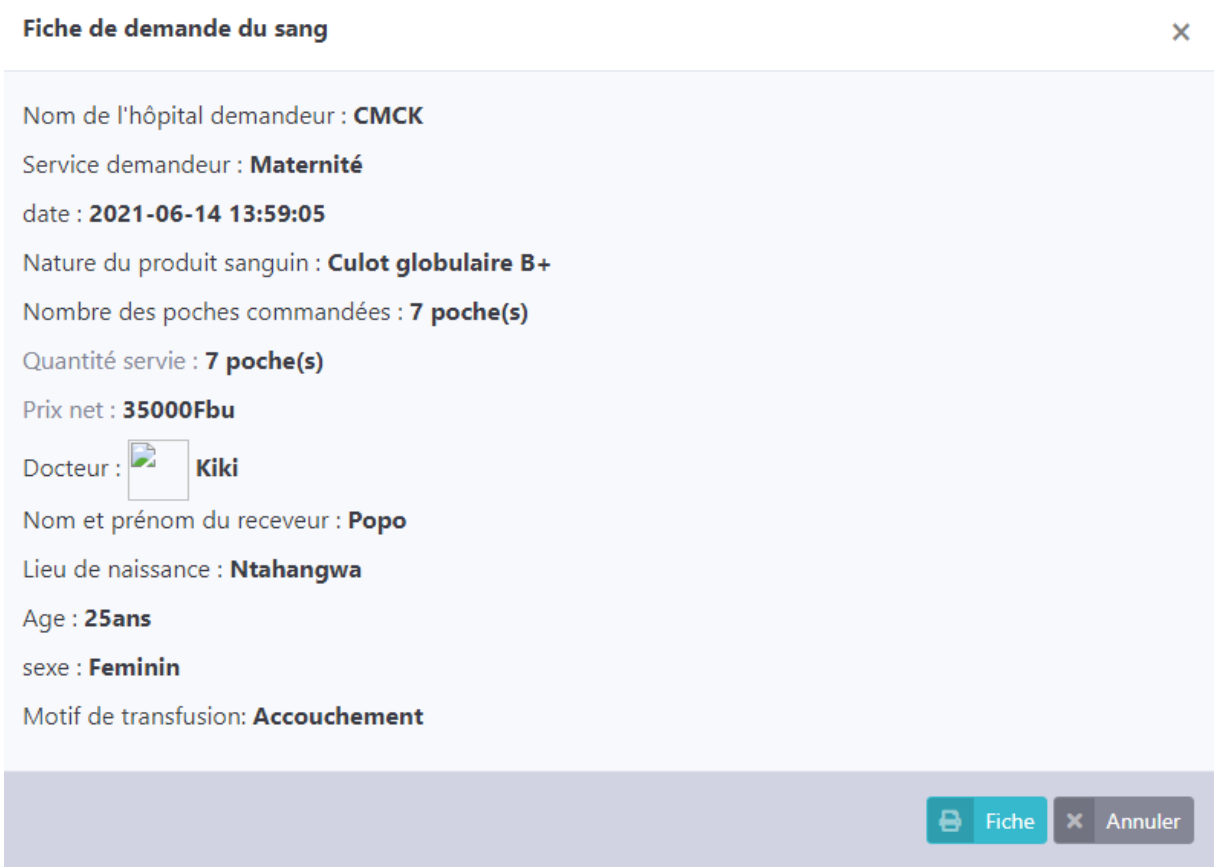


Figure 18: Page de visualiser des demandes déjà servies

L'institution a accès à visualiser et imprimer une fiche de demande du sang nominative et d'une fiche de demande de banque de sang.

CNTS
Centre National de Transfusion Sanguine

FICHE DE DEMANDE DU SANG

Bps : BUJUMBURA MAIRIE
Bds : Bujumbura sud
Institution : CMCK

Service demandeur	Maternité
Date	14-06-2021
Nature du produit sanguin	Culot globulaire
Nombre des poches commandées	7
Quantité servie	7
Prix	35000Fbu
Docteur	Kiki
Nom et prénom du receveur	Popo
Age du receveur	25
Lieu de naissance	Ntahangwa
Sexe du receveur	Feminin
Motif de transfusion	Accouchement

Fait à le / /

Signature

Figure 19: Page en .pdf de la fiche de demande nominative

REPUBLIQUE DU BURUNDI

MINISTERE DE LA SANTE PUBLIQUE ET DE LA LUTTE CONTRE LE SIDA

CENTRE NATIONAL DE TRANSFISION SANGUINE

BPS : BUJUMBURA MAIRIE
BDS : Bujumbura centre
HOPITAL : CPLR

FICHE DE REQUISITION DES PRODUITS SANGUINS

Produit	Dernière Qté	Dernière utilisation						Qté restante	Qté Commandée	Qté Servie
		Pédiatrie	Maternité	Médecine interne	Bloc opératoire	Néonatologie	Autres services			
A+	0							0	9	9
B+	0							0	11	11
AB+	0							0	6	6

Chef du service laboratoire **Directeur de l'hopital**

Figure 20: Page en.pdf de la fiche de demande de Banque de sang

Sur la partie facturation ;si la commande n'est pas servie totalement, l'institution peut visualiser sa facture dans la partie Facturation en cours et si la commande est servie totalement, elle pourra trouver la facture dans la partie Facturation en dette et si les factures sont payées, l'institution pourra trouver ses factures dans la partie Facture clôturée.

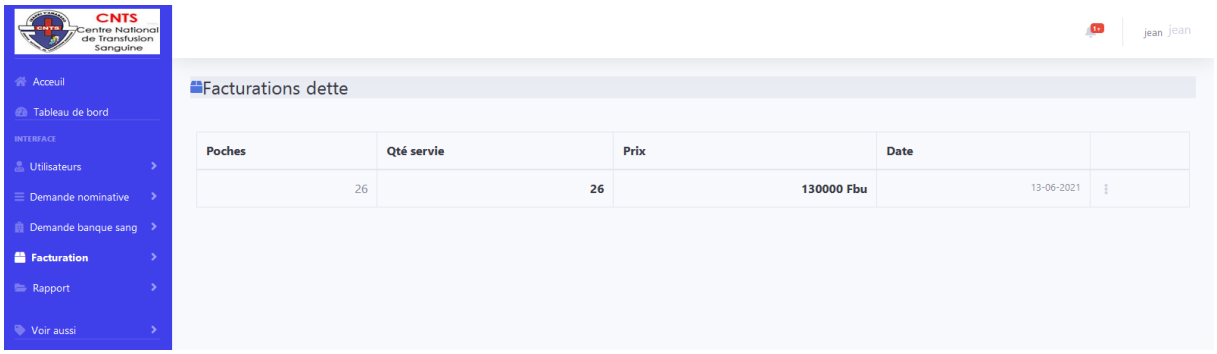


Figure 21: Page des factures des demandes déjà terminées

L'institution sanitaire peut aussi visualiser sa facture et l'état des commandes selon les types des produits.



Figure 22: Page pour visualiser les factures

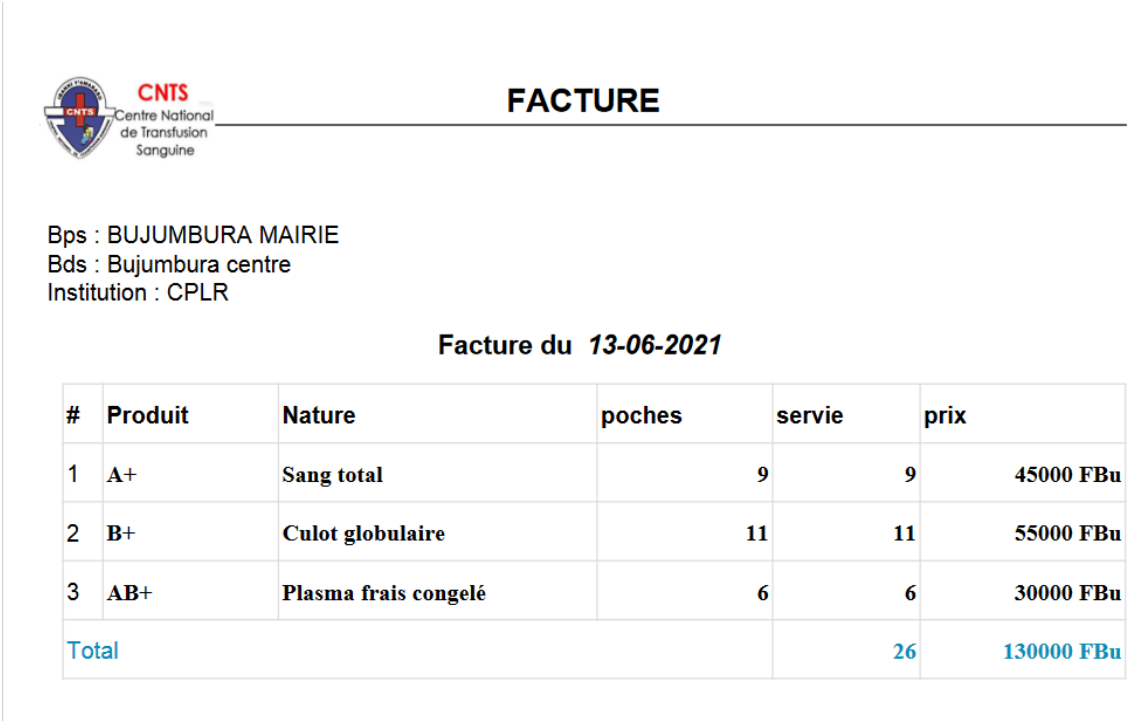


Figure 23: Facture en .pdf

L'institution sanitaire peut aussi voir des statistiques liées aux commandes, montant en cours de paiement, facturation en dette et des demandes faites par année.

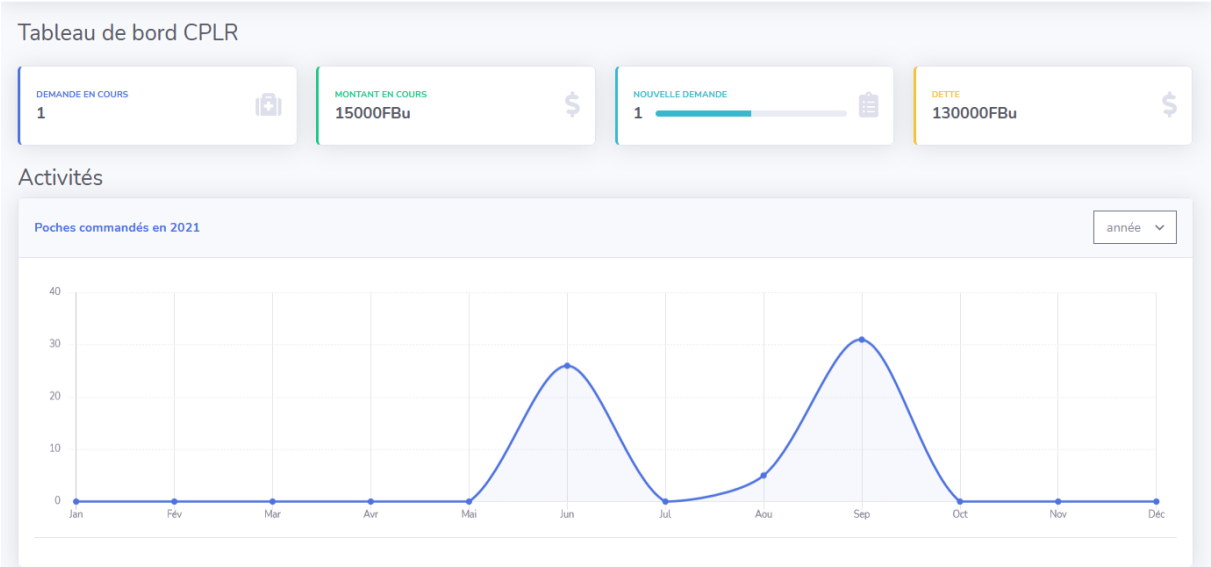


Figure 24: Page des statistiques

IV.3.3. Responsable du CNTS

Le responsable du CNTS fait la gestion des institutions, la gestion du stock,la gestion des commandes, l'approvisionnement des produits, la facturation et la gestion des rapports liés au stock et factures.

Le responsable du CNTS ayant mis ses identifiants, il va d'abord enregistrer une institution en suivant le formulaire suivant :

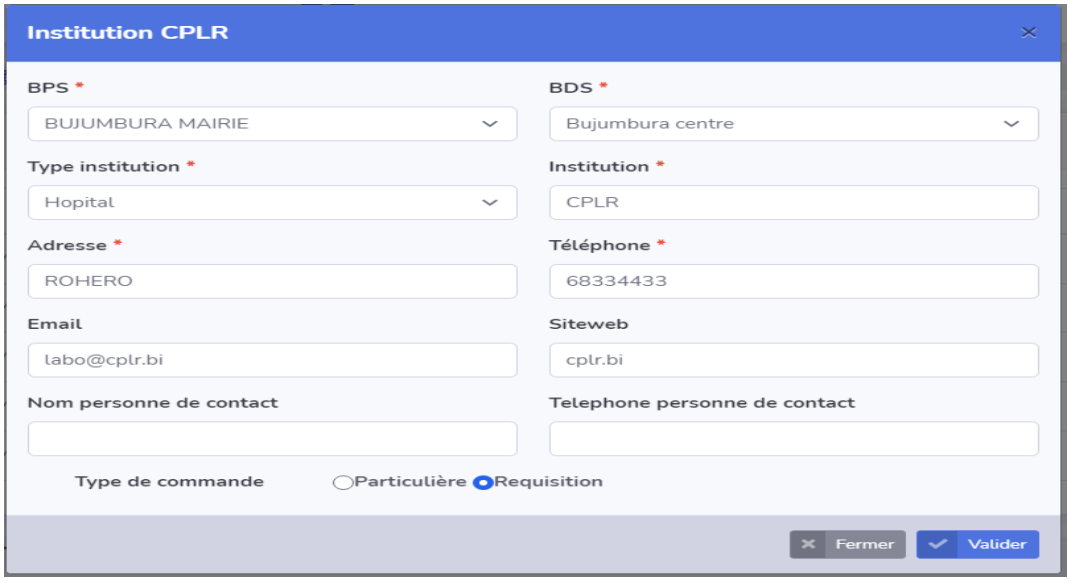


Figure 25: Page d'enregistrement d'une institution

Après, il va donner le rôle à une institution selon la page suivante:

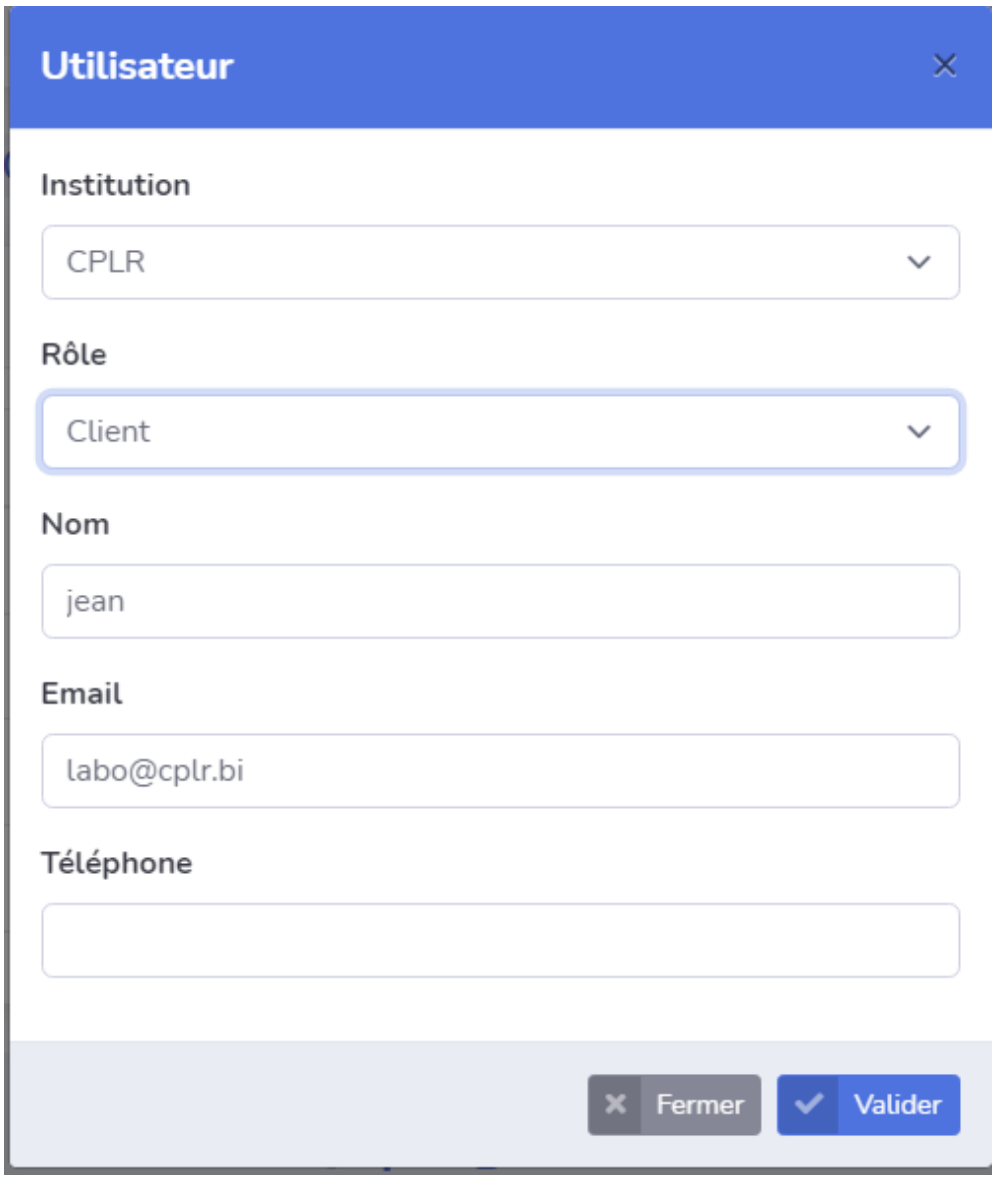


Figure 26: Page d'attribution des rôles à une institution sanitaire

Ensuite, le gestionnaire du CNTS va faire des entrées dans le stock selon la provenance de la collecte qui a été faite.

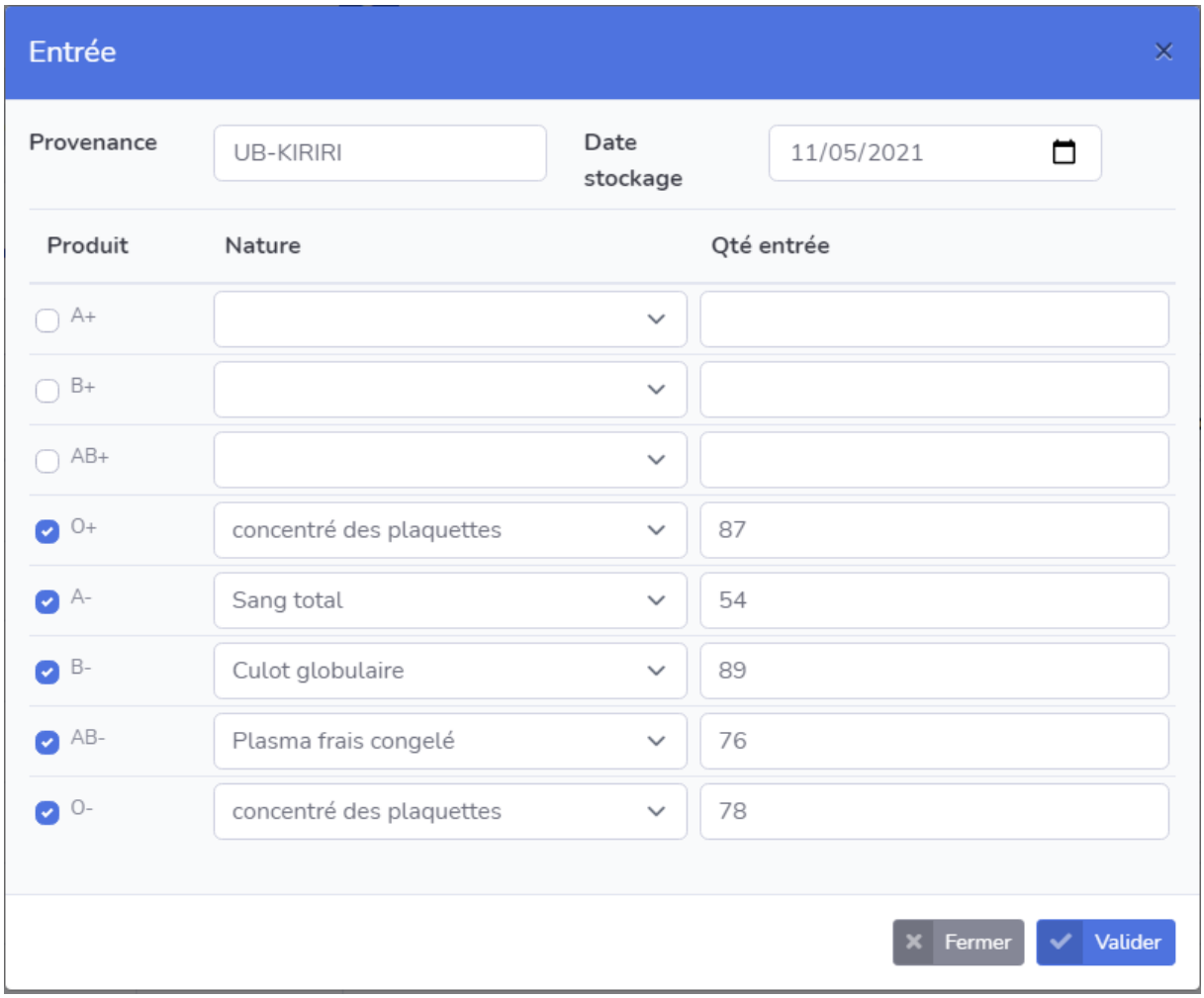


Figure 27: Page pour les entrées dans le stock

Le gestionnaire va visualiser l'état du stock et peut faire le tri par produit, nature ou date d'expiration.

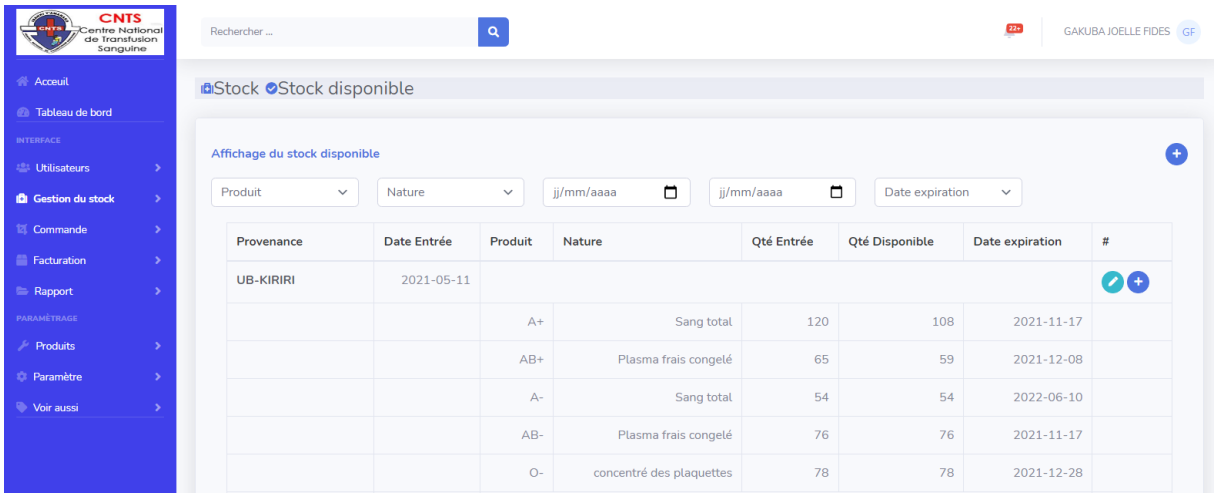


Figure 28: Page pour le stock disponible

Il peut aussi effectuer des livraisons selon le stock disponible et le type de demande, comme c'est indiqué dans la page suivante :

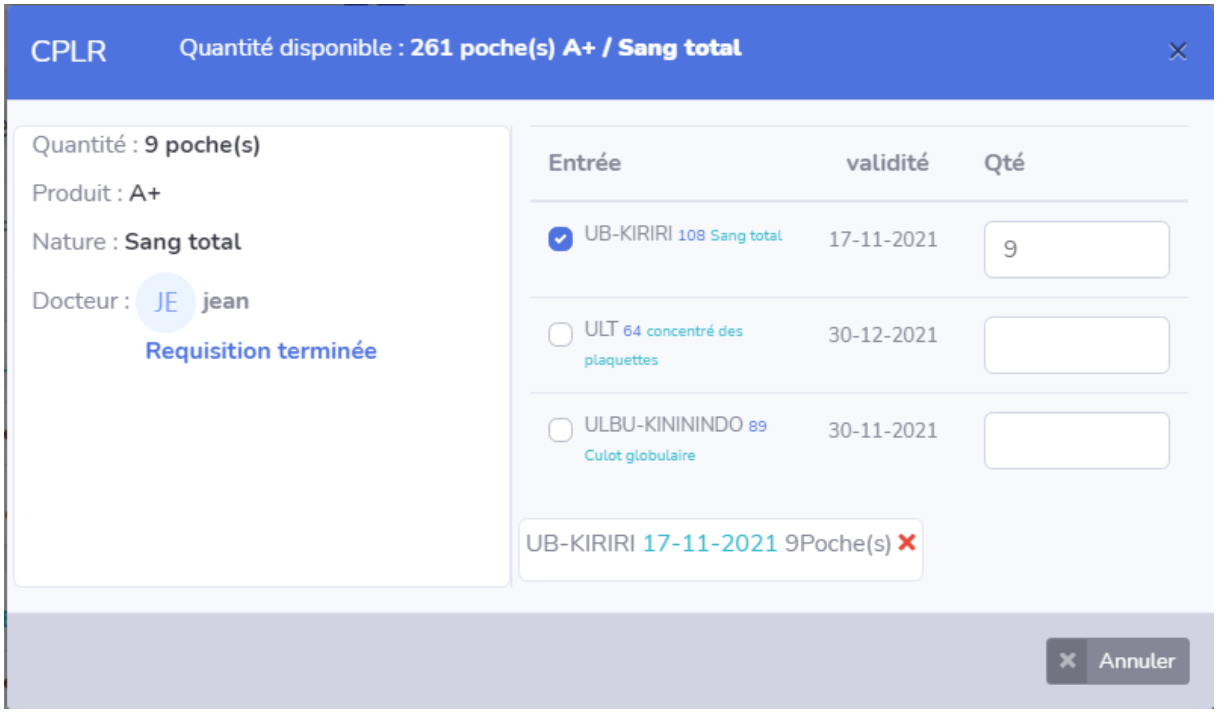


Figure 29: Page de livraison des demandes

Enfin, il va voir les factures et générer la facture et changer l'état de la facture.

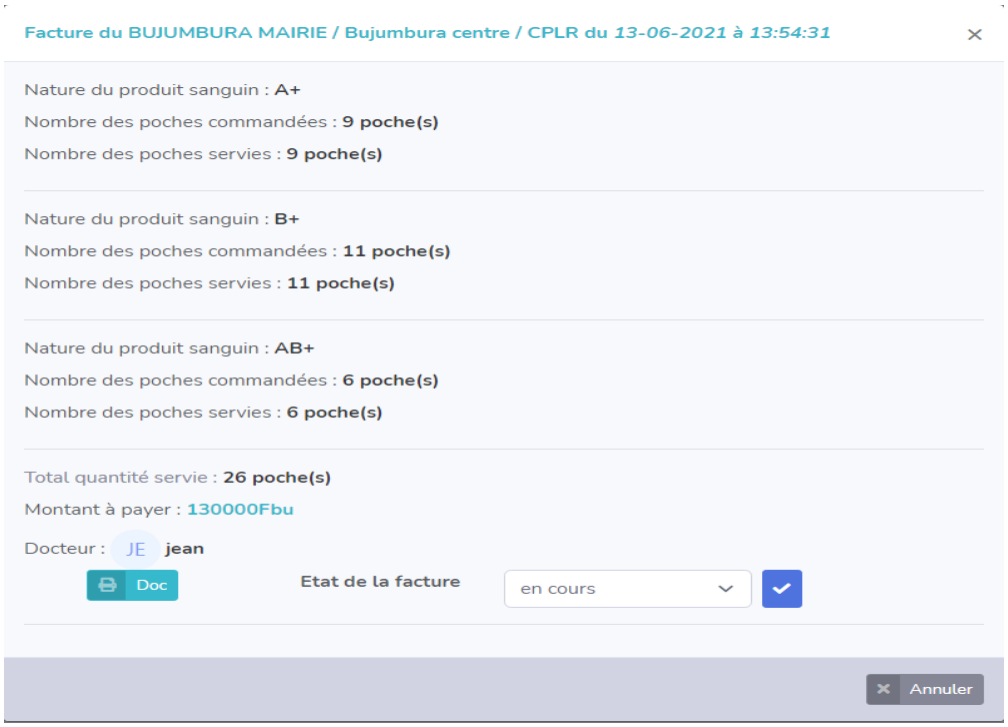


Figure 30: Page de facturation

CNTS
Centre National de Transfusion Sanguine

FACTURE

Bps : BUJUMBURA MAIRIE
Bds : Bujumbura centre
Institution : CPLR

Facture du *13-06-2021*

#	Produit	Nature	poches	servie	prix
1	A+	Sang total	9	9	45000 FBu
2	B+	Culot globulaire	11	11	55000 FBu
3	AB+	Plasma frais congelé	6	6	30000 FBu
Total				26	130000 FBu

Responsable : jean

Figure 31: Facture en .pdf

Il peut aussi voir les différentes notifications liées aux commandes et création d'un compte d'une institution.

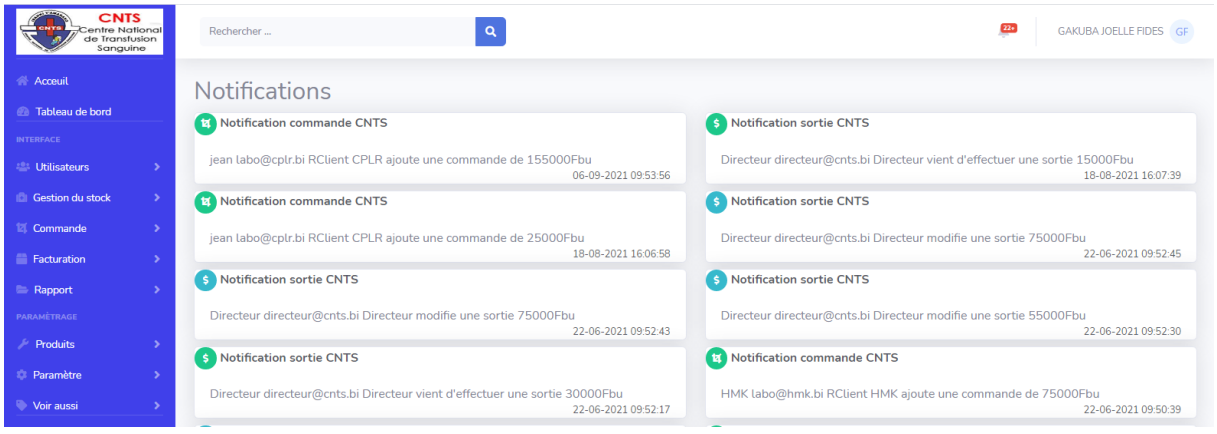


Figure 32: Page des notifications

IV.3.4. Directeur

Le Directeur va faire la gestion des accès de tous les utilisateurs et tout ce que le gestionnaire fait. La figure suivante montre les statistiques sur les entrées des poches, les commandes, les montants en cours de paiement et les dettes.

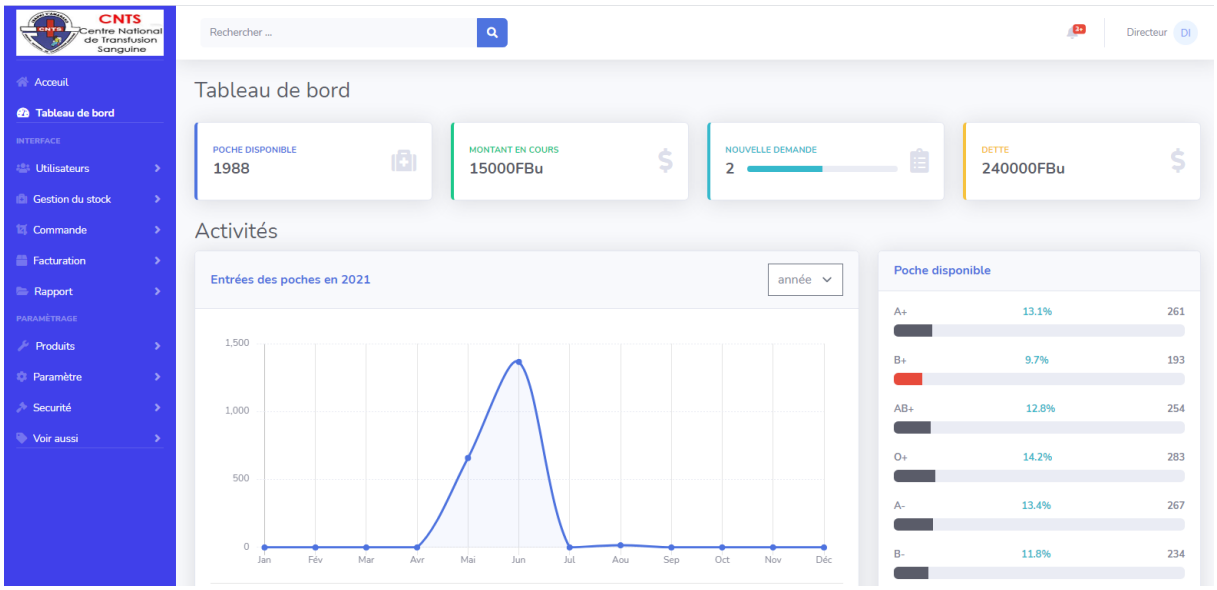


Figure 33: Page des statistiques

Cette figure ci-dessous est destinée à la gestion des rôles aux utilisateurs, le Directeur peut donner le rôle ou l'enlever à un utilisateur.

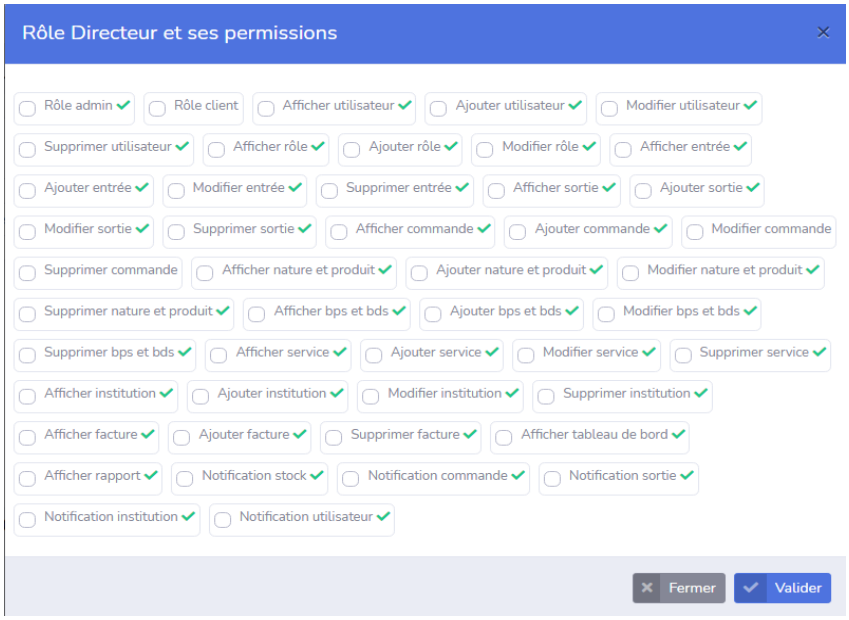
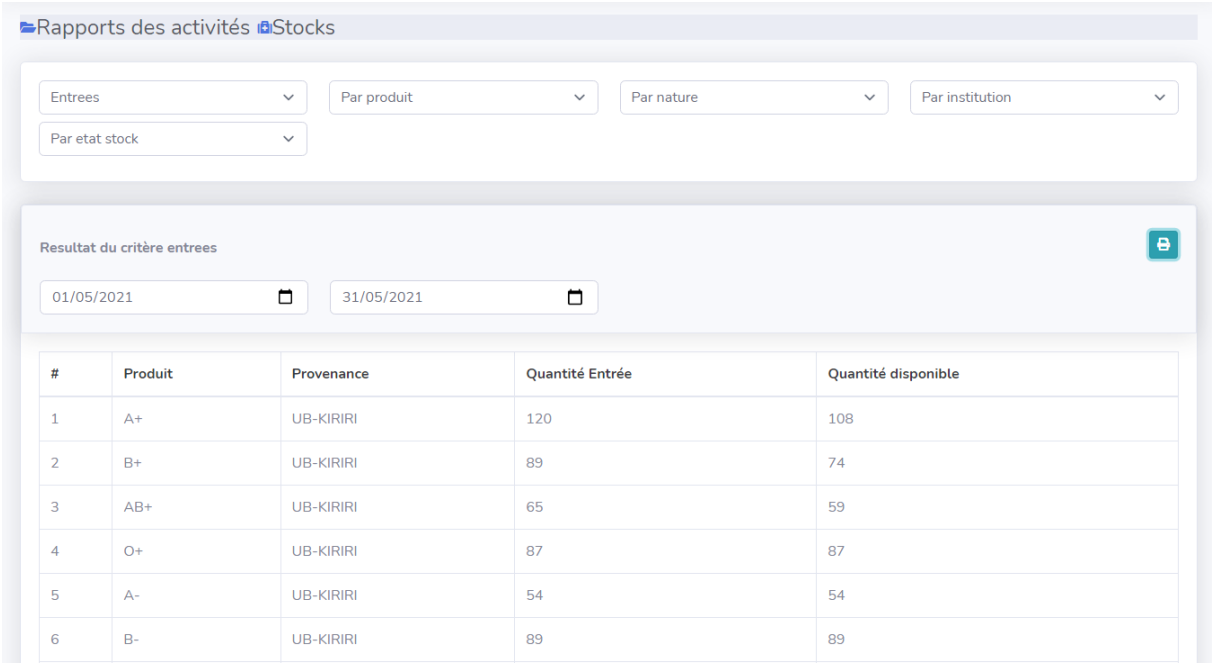


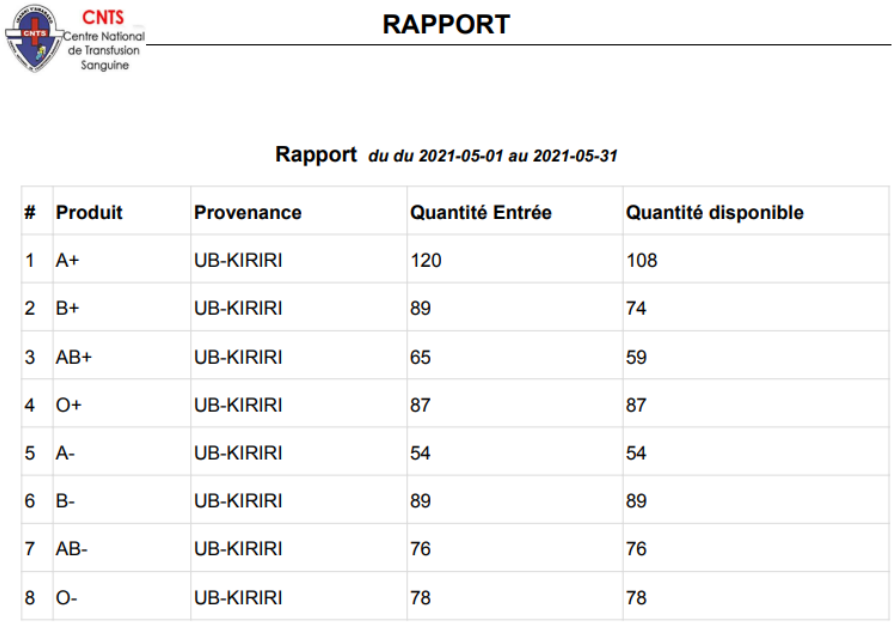
Figure 34: Page de gestion des rôles des utilisateurs

La figure suivante permet au Directeur de voir les différents rapports liés au stock et factures et il peut le télécharger en pdf.



#	Produit	Provenance	Quantité Entrée	Quantité disponible
1	A+	UB-KIRIRI	120	108
2	B+	UB-KIRIRI	89	74
3	AB+	UB-KIRIRI	65	59
4	O+	UB-KIRIRI	87	87
5	A-	UB-KIRIRI	54	54
6	B-	UB-KIRIRI	89	89

Figure 35: Page des rapports



#	Produit	Provenance	Quantité Entrée	Quantité disponible
1	A+	UB-KIRIRI	120	108
2	B+	UB-KIRIRI	89	74
3	AB+	UB-KIRIRI	65	59
4	O+	UB-KIRIRI	87	87
5	A-	UB-KIRIRI	54	54
6	B-	UB-KIRIRI	89	89
7	AB-	UB-KIRIRI	76	76
8	O-	UB-KIRIRI	78	78

Figure 36: Page d'un rapport en .pdf

IV.4. Conclusion

Dans ce chapitre, nous avons parlé pourquoi nous avons choisi de faire une application web en premier lieu, puis nous avons montré les technologies que nous avons utilisées et enfin nous avons montré les différentes pages d'applications qui expliquent le fonctionnement de notre nouveau système d'information.

CONCLUSION ET RECOMMANDATIONS

Conclusion

Au terme de ce travail, il convient de s'auto-évaluer, pour voir si les objectifs que nous nous sommes fixés ont été atteints. Rappelons que l'objectif principal était « la conception et réalisation d'une application web de gestion d'une Banque de sang : Cas du CNTS». Pour atteindre cet objectif, plusieurs méthodes et techniques ont été employés.

Partant du premier chapitre, nous avons fait une brève présentation du CNTS sans oublier d'analyser l'existant et de proposer des solutions. Dans le second chapitre, qui est à la fois théorique et pratique, après avoir introduit la théorie de conception des systèmes d'informations, nous avons modélisé le système d'information du CNTS. Le troisième chapitre fait la modélisation du nouveau système avec le formalisme mathématique de l'algèbre relationnelle. Et enfin, le quatrième chapitre fait la description des outils et techniques qui nous ont aidés dans le codage de notre application et la présentation de notre application.

Grâce à ce procédé méthodique, nous avons pu mener à bien notre projet. Nous pouvons maintenant affirmer que les objectifs spécifiques énumérés trouvent place dans notre application permettant ainsi de résoudre les problèmes du CNT dans la gestion de distribution et approvisionnement des produits sanguins identifiés ci-haut.

En conclusion, nous pouvons dire que les objectifs fixés ont été atteints. Néanmoins, les développeurs le savent, un logiciel sans bug n'existe pas. D'autres diront que « la perfection n'est pas de ce monde ». Malgré les efforts conjugués que nous avons fournis pour les éviter, nous ne pouvons malheureusement pas garantir que ceux-ci ne réapparaitront jamais. Avec la collaboration des utilisateurs nous espérons pouvoir les identifier et les corriger dans les versions futures.

Recommandations

Nous recommandons ce qui suit :

Au CNTS, d'exploiter l'application et de signaler tout disfonctionnement ou fonctionnalité manquante pour nous aider à améliorer l'application.

Aux futurs chercheurs :

1. de baser leurs développements sur des méthodes et techniques approuvées plutôt que de commencer le travail à zéro ;
2. d'améliorer l'application en ajoutant d'autres modules comme la gestion des donneurs et de collecte de sang ;
3. de renforcer la sécurité de l'application.

REFERENCES BIBLIOGRAPHIQUES

[1] P. Abir, Informatisation et pharmacie hospitalière, CHU Amiens-Université de Picardie, mai 2007.

[2] O. Sigaud, Introduction à la modélisation orientée objets avec UML, ENSTA, 2005-2006.

[3] J. M. DeFrance, Première Applications Web 2 .0 avec AJAX et PHP, EYROLLES, 2008.

[4] G. B. J. R. Ivar JACOBSON, Le Processus unifié de développement logiciel, c/o Eyrolles, 2000.

[5] X. e. B. I. Mounier, UML 2 pour les développeurs, EYROLLES, 2008.

[6] F. Felhi, Bases de données, Atlantic Int'l University.

[7] B. F.-T. M. Cristian Darie, AJAX and PHP Building Responsive Web Applications, Packt, 2011.

[8] J. P. e. R. Shaw, Choosing an appropriate technology, WEDC Lough borough University Leicestershire LE11 3TU UK, 2008.

[9] C. Herby, Apprenez à programmer en Java, EYROLLES, 2015.

[10] «WIKIPEDIA_PHP,» [En ligne]. Available: https://fr.wikipedia.org/wiki/PHP. [Accès le 13 05 2021].

[11] «WIKIPEDIA_JAVASCRIPT,» [En ligne]. Available: https://fr.wikipedia.org/wiki/JavaScript. [Accès le 13 05 2021].

[12] M. Nebra, Apprenez à créer des sites web avec html5 et css3, Openclassroom, 2012.

[13] «WIKIPEDIA_LARAVEL,» [En ligne]. Available: https://fr.wikipedia.org/wiki/Laravel. [Accès le 13 05 2021].

[14] «WIKIPEDIA_Bootstrap,» [En ligne]. Available: https://fr.wikipedia.org/wiki/Bootstrap_(framework). [Accès le 13 05 2021].

[15] MYSQL, [En ligne]. Available: https://www.mysql.com/fr/. [Accès le 13 05 2021].

[16] «Wikipedia-visual,» [En ligne]. Available: https://fr.wikipedia.org/wiki/Visual_Studio_Code. [Accès le 24 05 2021].

[17] «wikipedia-xampp,» [En ligne]. Available: https://fr.wikipedia.org/wiki/XAMPP. [Accès le 24 05 2021].

[18] M. Nebra, Concevez votre site avec PHP et Mysql, Openclassroom, 2012.

[19] J. Gabay, UML2 Analyse et Conception, EYROLLES, 2008.

[20] C. Soutou, Apprendre SQL avec MySQL, EYROLLES, 2012.

[21] A. P. a. T. Hawkins, Pro php application Performance, Apress, 2010.

[22] M. Fowler, Patterns of Enterprise Application Architecture, Addison Wesley, 2002.

[23] J.-F. Nogier, Ergonomie du logiciel et design web : Le manuel des interfaces utilisateur, DUNOD, 2008.

ANNEXES

A. Fiche de réquisition individuelle nominative



FICHE DE DEMANDE DU SANG

Nom de l'hôpital demandeur :.....

Date :.....

Nature du produit sanguin :.....

Nombre de poches :.....

Docteur :.....

Signature et cachet du médecin :.....

Nom et prénom du receveur :.....

Age du receveur :.....

Lieu de naissance :.....

Sexe :.....

Service demandeur :.....

Motif de transfusion :.....

Le receveur a-t-il été transfusé ? Oui ☐ Non ☐

Réservé au CENTRE NATIONAL DE TRANSFUSION SANGUINE OU CENTRE REGIONAL DE TRANSFUSION

Groupe du receveur		**Groupe du donneur**	
Epreuve De Compatibilité	N°	Fait avec	
	N°		
	N°		
	N°		

Fait à.....

Signature.....

B. Fiche de réquisition des banques de sang

REPUBLIQUE DU BURUNDI

MINISTERE DE LA SANTE PUBLIQUE
ET DE LA LUTTE CONTRE LE SIDA
BPS :
BDS :
HOPITAL :

CENTRE NATIONAL
DE TRANSFUSION SANGUINE

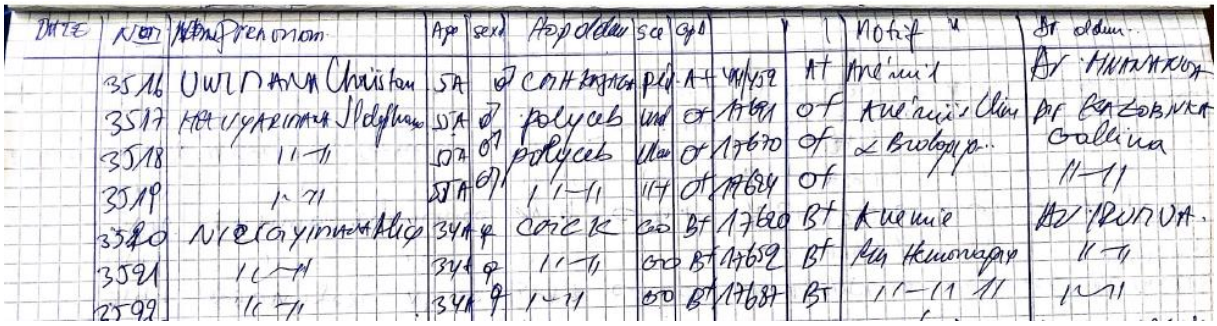
FICHE DE REQUISITION DES PRODUITS SANGUINS

Produits sanguins	Date	Quantité reçue antérieurement	Utilisation du sang antérieurement reçu							Quantité restante	Quantité demandée	Quantité servie	Observations
			Pédiatrie	Maternité	Médecine interne	Bloc opératoire	Néonatologie	Autres services	Quantité détruite				
A+													
B+													
AB+													
0+													
A-													
B-													
AB-													
0-													
Trousses													

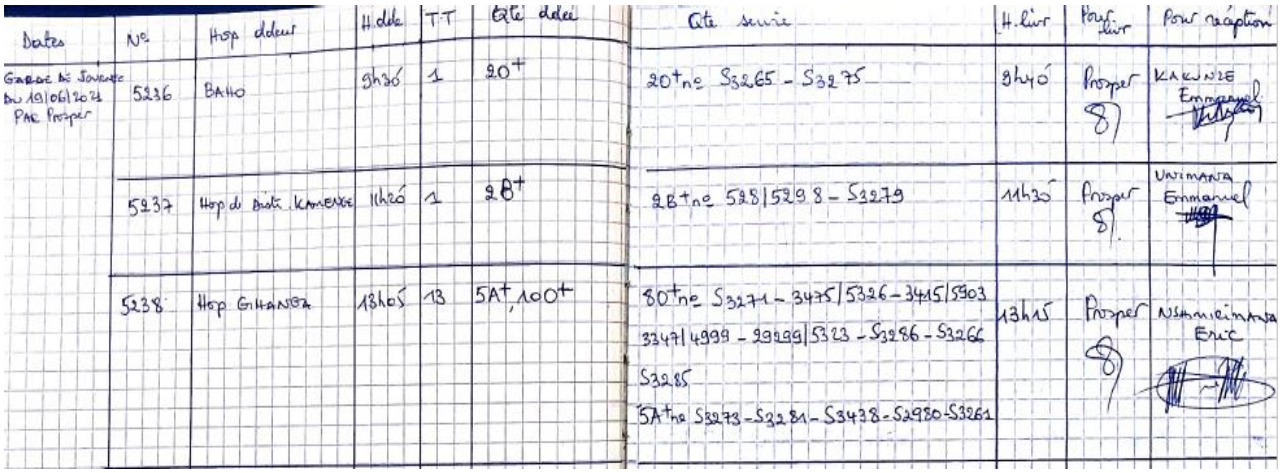
Chefs du service Laboratoire

Directeur de l'hôpital

C. Registre des épreuves de compatibilité



D. Registre de livraison de sang



Dates	N°	Hop ddeur	H.dde	T.T	Qté ddée	Qté servie	H.livr	Pour livr	Pour réception
Garde de journée du 19/06/2021 PAR Prosper	5236	BAHO	9h30	1	20+	20+ne S3265 - S3275	9h40	Prosper	KAKUNZE Emmanuel
	5237	Hop di Dioti KAMENGE	11h20	1	28+	28+ne 528/5298 - S3279	11h35	Prosper	UWIMANA Emmanuel
	5238	Hop GITANGA	13h05	13	5A+,100+	80+ne S3271 - 3475/5326 - 3415/5903 3347/4999 - 99299/5323 - S3286 - S3266 S3285 5A+ne S3273 - S3281 - S3438 - S2980 - S3261	13h15	Prosper	NSHIMIRIMANA Eric

E. Bordereau d'expédition

Centre National de
Transfusion Sanguine
C.N.T.S
B.P.3585
Tél:22 35 76/25 62 94/ 22 64 25

Bordereau d'Expédition n°

Expéditeur:.....
Destinataire:.....
Transporteur: .....
le.....

Nombre de colis	Emballage	Nature du contenu	P.U	P.T	Observation

Signature d'Expéditeur pour expédition conforme

Signature du transporteur pour prise en charge, remise en bon état

Signature du réceptionnaire pour décharge, sous réserves mentionnées en observations