



Projet de Fin d'Études

FILIÈRE

Génie Logiciel

SUJET

Automatisation du Registre du Traitement

Réalisé par

Mehdi CHARIFE

Sous la direction de

Pr. Azeddine EL HASSOUNY
(ENSIAS)

M. Yassine RACHIDI (BCP)
Mme. Imane HLOU (BCP)

Membres du jury

Pr. Azeddine EL HASSOUNY

Pr. Ahmed ETTALBI

Pr. Abdellatif EL FAKER

Année Universitaire : 2024-2025

“

À mes chers parents, dont l'amour et le soutien ont été inestimables, aucune déclaration de gratitude ne serait suffisante pour exprimer la profondeur de mon respect et de ma reconnaissance envers vous. Vos sacrifices et vos encouragements ont été les fondements de ma réussite éducative et personnelle.

À mes amis et camarades étudiants de l'ENSIAS, qui ont été une source constante de motivation et de soutien tout au long de mon parcours académique. Votre amitié et votre collaboration ont été essentielles dans mon développement.

Au corps enseignant du département de génie logiciel, je suis reconnaissant pour vos connaissances partagées, votre patience et vos conseils qui ont éclairé mon chemin académique.

À tous ceux qui ont contribué de près ou de loin à la réalisation de ce travail

Merci.

”

- Mehdi

Remerciement



Résumé

Un registre de traitement est un document qui recense de manière détaillée les traitements de données personnelles effectués par une organisation. Il permet d'identifier les catégories de données traitées, les finalités des traitements, les accès et les communications des données, etc., et ce en maintenant une fiche descriptive pour chaque activité du traitement. Actuellement, la Banque Centrale Populaire dispose de 80 fiches de traitement réparties sur 26 fichiers Excel, ce qui présente une difficulté en matière de revue et de consultation, d'où le besoin de mettre une solution informatisée qui servira à automatiser le registre du traitement. Afin de répondre à cet objectif, une méthodologie de développement a été adoptée, commençant par une conception globale initiale, et suivie d'une construction guidée par les fonctionnalités. Le résultat final est une application web qui permet d'optimiser les processus liés à la gestion du registre du traitement.



Liste des abréviations

CNDP	<i>Commission Nationale de Contrôle de la Protection des Données à caractère Personnel</i>
BCP	<i>Banque Centrale Populaire</i>
DPO	<i>Data Protection Officer</i>
C4	<i>Context, Container, Component, and Code</i>
RBAC	<i>Role Based Access Control</i>
SPA	<i>Single Page Application</i>
API	<i>Application Programming Interface</i>
REST	<i>Representational State Transfer</i>
CI/CD	<i>Continuous Integration and Continuous Delivery/Continuous Deployment</i>

Table des matières

1	Contexte général du projet	1
1.1	Organisme d'accueil	1
1.1.1	Aperçu historique	1
1.1.2	Structure organisationnelle	2
1.1.3	Missions et valeurs	4
1.2	Présentation du projet	5
1.2.1	Client : Fonction Conformité Groupe	5
1.2.2	Problématique	5
1.2.3	Objectif	5
2	Analyse fonctionnelle	6
2.1	État de l'existant	7
2.2	Spécification des besoins	8
2.2.1	Besoins fonctionnels	8
2.2.2	Besoins non-fonctionnels	8
2.3	Acteurs du système	9
2.4	Cas d'utilisations	10
2.4.1	Diagramme de cas d'utilisation global simplifié	10
2.4.2	Diagramme de cas d'utilisation de l'Expert métier	11
2.4.3	Diagramme de cas d'utilisation du RPO	12
2.4.4	Diagramme de cas d'utilisation du DPO	13
2.4.5	Diagramme de cas d'utilisation du GDPO	14
2.4.6	Diagramme de cas d'utilisation de l'agent d'habilitation	15
3	Conception de la solution	16
3.1	Modélisation C4	16
3.2	Diagramme de contexte	17
3.3	Diagramme de conteneurs	18
3.4	Diagramme des composants du serveur web	19
3.5	Diagrammes de classes du serveur d'API	20
3.5.1	Couche modèle	20
3.6	Diagrammes de séquences	21
3.6.1	Ajout d'un paramétrage	21
4	Réalisation	22
4.1	Outils de réalisation	23
4.1.1	Spring	23
4.1.2	React	23
4.1.3	Postgres	24

4.1.4	Docker	24
4.1.5	JUnit	25
4.1.6	Mockito	25
4.1.7	Gradle	26
4.1.8	Checkstyle	26
4.1.9	PMD	26
4.1.10	Git	27
4.1.11	Github	27
4.1.12	Github Actions	27
4.2	Interfaces graphiques	28
4.2.1	Gestion des paramètres	28
4.2.2	Ajout d'une fiche de traitement	29
4.2.3	Liste des fiches de traitement	30

Table des figures

1.1	Logo de la Banque Populaire	1
1.2	Organigramme du groupe BCP	3
2.1	Diagramme de cas d'utilisation global simplifié	10
2.2	Diagramme de cas d'utilisation de l'expert métier	11
2.3	Diagramme de cas d'utilisation du RPO	12
2.4	Diagramme de cas d'utilisation du DPO	13
2.5	Diagramme de cas d'utilisation du GDPO	14
2.6	Diagramme de cas d'utilisation de l'agent d'habilitation	15
3.1	Diagramme de contexte	17
3.2	Diagramme de conteneurs	18
3.3	Diagramme des composants	19
4.1	Logo de Spring	23
4.2	Logo de React	23
4.3	Logo de Postgres	24
4.4	Logo de Docker	24
4.5	Logo de JUnit	25
4.6	Logo de Mockito	25
4.7	Logo de Gradle	26
4.8	Logo de Checkstyle	26
4.9	Logo de PMD	26
4.10	Logo de Git	27
4.11	Logo de Github	27
4.12	Page de paramétrage	28
4.13	Page d'ajout d'une fiche de traitement	29
4.14	Page de la liste des fiches de traitement	30

Introduction Générale

Dans le contexte d'un nombre croissant d'efforts visant à protéger la vie privée au Maroc, la commission nationale de contrôle de la protection des données à caractère personnel (CNDP) a été créée par la loi n°09-08 du 18 février 2009. Cette commission est chargée de vérifier que les traitements des données personnelles sont licites, légaux et qu'ils ne portent pas atteinte à la vie privée, aux libertés et droits fondamentaux de l'homme.

Conformément aux exigences de la CNDP, la Banque Centrale Populaire est tenue de maintenir un registre du traitement détaillant l'ensemble des activités de traitement effectuées par la banque. Chaque traitement doit faire l'objet d'une fiche décrivant un sous-ensemble des caractéristiques du traitement en question. Ces caractéristiques incluent les catégories de données traitées, les finalités du traitement, les personnes concernées, les accès et les communications des données, les transferts des données vers l'étranger, ainsi que plusieurs autres.

Actuellement, le registre du traitement de la banque existe sous forme de fichiers Excel et donc présente une certaine difficulté en matière de revue, de consultation, et de modification. Suite à ces contraintes, la fonction Conformité Groupe pour la BCP a exprimé un besoin d'automatisation du registre du traitement. Le présent travail constitue donc une description structurée de l'ensemble des efforts entrepris afin de répondre à ce besoin.



Chapitre 1

Contexte général du projet

1.1 Organisme d'accueil

Le Groupe BCP est un groupe financier panarabique et universel au service de toutes les catégories socio-professionnelles. À vocation inclusive, il offre à ses clients particuliers, professionnels et entreprises de toutes tailles, des produits bancaires, d'assurance et de service.

La Banque Centrale Populaire constitue l'organe central du Groupe, qui se compose de huit Banques Populaires Régionales (BPR), de trois fondations et de plusieurs filiales au Maroc et à l'international.



FIGURE 1.1 – Logo de la Banque Populaire

1.1.1 Aperçu historique

Fondée en 1961, la BCP a rapidement émergé comme leader du marché des dépôts au Maroc, atteignant 1 milliard de dirhams en 1974. En 1980, elle comptait déjà 500 000 clients et 5 milliards de dirhams en dépôts. La BCP a réalisé son ambition d'expansion à l'international en créant la Banque Chaabi du Maroc à Paris et une succursale à Bruxelles en 1976. En 2000, elle s'est transformée en société anonyme à conseil d'administration et a été cotée à la Bourse de Casablanca en 2004.

Le développement de la BCP s'est poursuivi avec des investissements stratégiques et une expansion continue. En 2015, la banque a porté ses participations dans les BPRs à 52 %, devenant ainsi majoritaire. Dans la même année, la BCP a enregistré une augmentation de 14,4 % de son résultat net, atteignant 2,5 milliards de dirhams et 5,2 millions de clients.

En 2020, la Banque Populaire a participé avec une contribution d'un milliard de dirhams au fonds de lutte contre le coronavirus fondé par le Roi Mohammed VI.

1.1.2 Structure organisationnelle

La Banque Centrale Populaire se distingue par une structure organisationnelle solide et stratégiquement pensée, reflétant son engagement envers l'excellence opérationnelle, la gouvernance efficace et la satisfaction des clients.

Cette organisation, fruit d'un travail de co-construction avec les instances de gouvernance du Groupe et l'équipe dirigeante, s'articule autour de :

- **Direction Générale Banque Commerciale** : structurée autour d'entités Producteurs par segment de clientèle (Particuliers, Professionnels, Marocains Du Monde, TPE, PME et GE) et d'entités Distributeurs, au Maroc et dans les pays de présence en Europe, au Moyen Orient et en Amérique ;
- **Direction Générale BCP et International** : structurée autour, d'une part, d'entités Plateformes Produits et Services, spécialisées et mutualisées à l'échelle du Groupe et, d'autre part, de la Banque de l'International en charge du développement et du pilotage des activités du Groupe en Afrique Subsaharienne, au Moyen Orient et dans l'Océan Indien.
- **Direction Générale Risques Groupe** : structurée autour des filières de maîtrise et de gestion des risques ayant une portée Groupe, menant une triple mission : normative, de contrôle et de service ;
- Consolidation du positionnement stratégique, en rattachement à la **Présidence Direction Générale**, des fonctions en charge des **Finances et Performances Groupe**, de la **Stratégie Groupe**, de la **Conformité et du Développement Durable**, ainsi que celles inhérentes à la **Marque, Capital Humain & Gouvernance Groupe**.

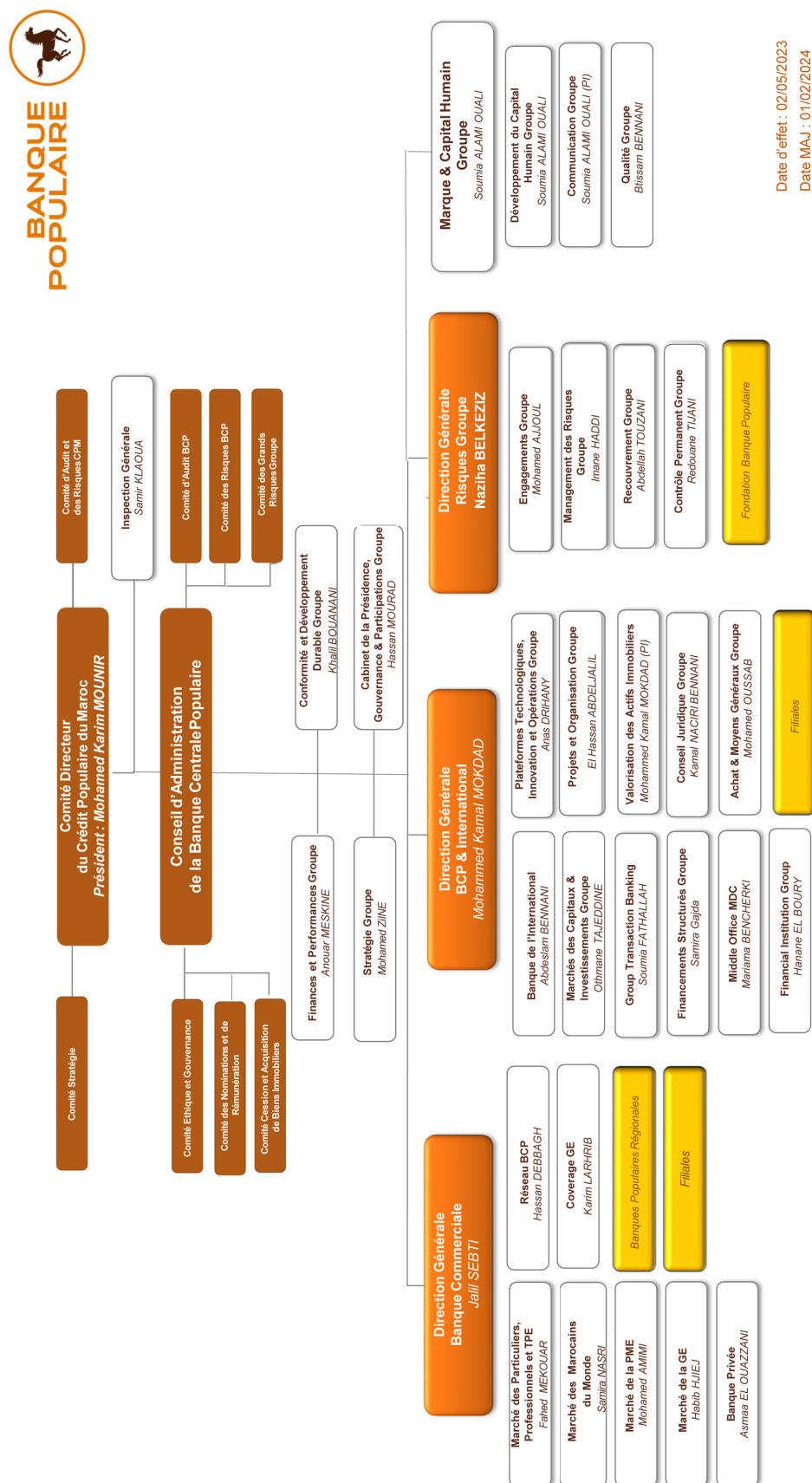


FIGURE 1.2 – Organigramme du groupe BCP

1.1.3 Missions et valeurs

Missions

Les missions de la Banque Centrale Populaire peuvent être divisées en deux piliers principaux :

- Réaliser toutes les opérations bancaires pour lesquelles elle est habilitée en tant qu'établissement de crédit et coordonner la politique financière du Groupe.
- Incarner l'organisme central bancaire des Banques Populaires Régionales en assurant leur refinancement, la gestion de leurs excédents de trésorerie et les services d'intérêt commun pour ses organismes.

Valeurs

Les valeurs de la BCP sont au cœur de son identité et de ses actions. Elles guident ses décisions, inspirent ses équipes, et assurent une relation de confiance avec ses clients et partenaires. Celles-ci incluent :

- **Proximité** : Valeur historique de la Banque Populaire, reflète son héritage coopératif et son ancrage local. Elle consiste à écouter et comprendre les besoins des clients, collaborateurs et partenaires pour adapter ses offres et sa stratégie d'expansion locale. Cette approche garantit la pertinence et la réactivité de ses actions sur le terrain ;
- **Citoyenneté** : Se manifeste par un engagement responsable envers les communautés, les partenaires et les écosystèmes où la Banque Centrale Populaire opère. Elle se traduit par la promotion de l'intérêt collectif dans toutes ses initiatives et ses choix ;
- **Performance** : Stratégie définie par la création de valeur durable pour les clients, collaborateurs, actionnaires et partenaires. Cela se matérialise par un engagement collectif pour atteindre notre ambition commune ;
- **Innovation** : Se caractérise par une réinvention continue de la proposition de valeur, des modes de fonctionnement et de communication, tout en cultivant l'humilité. Elle implique la création d'un environnement de confiance favorable à l'expression de la créativité et à l'audace.

1.2 Présentation du projet

1.2.1 Client : Fonction Conformité Groupe

En référence aux meilleures pratiques et aux recommandations de la directive de Bank Al-Maghrib n°49/G, le Groupe s'est doté, au début de l'exercice 2007, d'une fonction "Conformité". Celle-ci a pour mission principale le contrôle permanent de dernier ressort de l'application des lois, réglementation et normes en vigueur, afin de renforcer son système de gouvernance et la confiance des marchés ainsi que d'asseoir son processus de développement aussi bien sur le marché national qu'à l'international.

Le périmètre d'action de cette fonction couvre quatre activités principales : la conformité réglementaire, la déontologie et la gouvernance, la lutte anti-blanchiment et due diligence ainsi que la supervision du système de contrôle interne du Groupe.

A ce titre, et à fin de respecter les exigences de la CNDP en matière de traitements des données à caractère personnel, la fonction conformité a exprimé le besoin d'automatisation du registre du traitement.

1.2.2 Problématique

Conformément aux directions de la CNDP, la BCP est tenue de maintenir un registre qui décrit l'ensemble des traitements des données à caractère personnel effectuées par la banque. Chaque traitement doit faire l'objet d'une fiche qui permet d'identifier le traitement et la nature des données sur lesquelles il opère.

Actuellement, la tenue du registre de traitement est réalisée à l'aide de fichiers Excels dont la manipulation, la validation, et la consultation comprend un nombre de tâches répétitives qui présentent un risque augmenté d'erreur, de mal-pratique, et un coût élevé en matière de temps et de ressources.

Face à ces contraintes, le besoin d'automatisation du registre de traitement s'avère non-seulement pertinent mais également impératif.

1.2.3 Objectif

Le projet vise à développer une application web qui permettra d'automatiser les processus liées à la gestion du registre du traitement, et ce afin d'assurer la conformité réglementaire de la BCP vis-à-vis les directions de la CNDP concernant les traitements des données à caractère personnel.

Chapitre 2

Analyse fonctionnelle

Le présent chapitre contient une présentation de l'ensemble des fonctions attendues du système, les différents acteurs qui interagiront avec celui-ci, ainsi que la répartition des responsabilités de chaque acteur par rapport aux différentes fonctionnalités. Une partie est également consacrée pour analyser l'état de l'existant et son positionnement vis-à-vis le besoin analysé.

La collecte, l'organisation, et la priorisation des fonctions s'est basée sur l'analyse d'un document d'expression de besoin fourni par la fonction Conformité du Crédit Populaire du Maroc (CPM), ainsi qu'un nombre de réunions effectuées avec les représentants de cette fonction.

2.1 État de l'existant

Initialement, le registre de traitement utilisé par la banque existait sous forme de 80 fiches de traitement réparties sur 26 fichiers Excel. L'ajout, la modification, la consultation, et la validation des fiches de traitements se faisaient manuellement en manipulant des fichiers Excel qui faisaient l'objet de plusieurs transferts par mail entre les différents acteurs impliqués dans les processus liés à la gestion du registre de traitement. Bien que cette approche permettait à la banque de maintenir une certaine visibilité sur l'ensemble de ses traitements des données à caractère personnel, elle présentait certaines lacunes et difficultés dont les plus impactantes étaient :

Structure non standardisée :

- **Variabilité de format** : Les fichiers Excel peuvent avoir des structures différentes, ce qui complique l'importation automatisée.
- **Absence de normalisation** : Les entités peuvent ne pas suivre un schéma cohérent d'un fichier à l'autre, rendant difficile la définition d'un modèle de données uniforme.

Problèmes de qualité de données :

- **Données manquantes ou incomplètes** : Certains champs importants peuvent être vides ou mal remplis.
- **Données incohérentes** : Il peut y avoir des incohérences dans les valeurs (par exemple, des formats de dates différents, des fautes de frappe).
- **Erreurs de saisie** : Les données saisies manuellement sont sujettes à des erreurs humaines, ce qui peut affecter la qualité des informations.

Limitations techniques :

- **Taille des fichiers** : Les fichiers Excel volumineux peuvent poser des problèmes de performance lors de leur lecture et traitement ;
- **Dépendance à un logiciel** : L'accès aux fichiers Excel nécessite souvent l'utilisation d'un logiciel spécifique (Microsoft Excel ou un équivalent), ce qui peut limiter la portabilité.

Gestion des versions :

- **Multiples versions** : La gestion de plusieurs versions des fichiers peut entraîner des problèmes de synchronisation et de traçabilité des modifications.
- **Absence de contrôle de version** : Il peut être difficile de suivre les modifications apportées aux fichiers, surtout si plusieurs personnes y ont accès.

2.2 Spécification des besoins

2.2.1 Besoins fonctionnels

- Tenir à jour le référentiel et la structure de registre de traitement ;
- Saisir les données par fiche et selon la structure proposée en ligne et en colonne ;
- Gérer la traçabilité des saisies, des modifications, et des actualisations des fiches de traitement ;
- Gérer les relations et les renvois entre les fiches traitement ;
- Recherches multicritères, par Catégorie de traitement, Sous-catégorie de traitement, traitement et/ou recommandation CNDP (ou Autorité locale pour les filiales à l'international) ;
- Centraliser l'ensemble des traitements des données à caractère personnel au sein de l'établissement ;
- Disposer d'un registre des traitements exhaustif, clair et facilement accessible ;
- Avoir une vue d'ensemble sur les traitements effectués par les entités de la BCP, et ce afin de mener une démarche d'amélioration continue d'inventaire et de classification des données à caractère personnel ;
- Faire revoir, compléter et valider facilement ces traitements par les RPO et Relais en charge de la protection des données personnelles.

2.2.2 Besoins non-fonctionnels

- **Ergonomie :**
 - Cohérence de l'interface : Par rapport aux attentes de l'utilisateur (facilité, simplicité, clarté, arrangement des écrans) ;
 - Documentation implicite : En introduisant des Messages d'erreur ou d'accompagnement et un système d'aide en ligne ;
 - Facilité d'utilisation : Applicabilité de l'expérience de l'utilisateur dans un contexte analogue, réactions «naturelles» et intuitives ;
 - Minimisation de la possibilité d'erreurs de manipulation : Offrir des listes déroulantes, choix des dates au lieu de les saisir, ... etc ;
 - Minimisation de la possibilité d'erreurs de manipulation : Offrir des listes déroulantes, choix des dates au lieu de les saisir, ... etc.
- **Sécurité :** En plus de la sécurité qu'offre le système d'exploitation, l'application doit permettre :
 - La gestion des habilitations concernant l'accès et l'utilisation de l'application ;
 - La gestion du système d'habilitation doit être du ressort d'un administrateur de l'outil.

2.3 Acteurs du système

La gestion du registre de traitement nécessite la collaboration étroite de plusieurs acteurs dont l'interaction efficace est primordiale pour l'atteinte des objectifs qui constituent la raison d'être du registre de traitement :

- **Expert métier** : Il s'agit de l'acteur le plus familiarisé avec les traitements de données. Il peut être impliqué pour fournir des connaissances spécifiques sur la manière dont les données sont collectées, traitées et utilisées dans une activité ou un processus particulier ;
- **Délégué à la protection des données (DPO)** : Également appelé Data Protection Officer, il est chargé d'assurer la conformité aux réglementations en matière de protection des données au sein d'une organisation. Sa mission principale est d'informer et de conseiller l'organisme, de contrôler le respect du présent règlement et de coopérer avec les autorités de contrôle ;
- **Relais de Protection des données Opérationnelle (RPO)** : Sa responsabilité est d'effectuer un premier niveau de validation pour les fiches de traitement des données ;
- **Global Data Protection Officer (GDPO)** : Il opère à l'échelle mondiale dans une organisation internationale. Il s'assure que les pratiques de protection des données sont cohérentes dans toutes les filiales de l'entreprise, en tenant compte des réglementations spécifiques à chaque juridiction.

La collaboration entre l'expert métier, le Délégué à la Protection des données (DPO), le Relais de Protection des données Opérationnelle (RPO), et le Global Data Protection Officer (GDPO) est essentielle. Cette synergie vise à garantir la conformité et la cohérence des pratiques de protection des données, permettant ainsi d'atteindre les objectifs du registre de traitement.

2.4 Cas d'utilisations

2.4.1 Diagramme de cas d'utilisation global simplifié

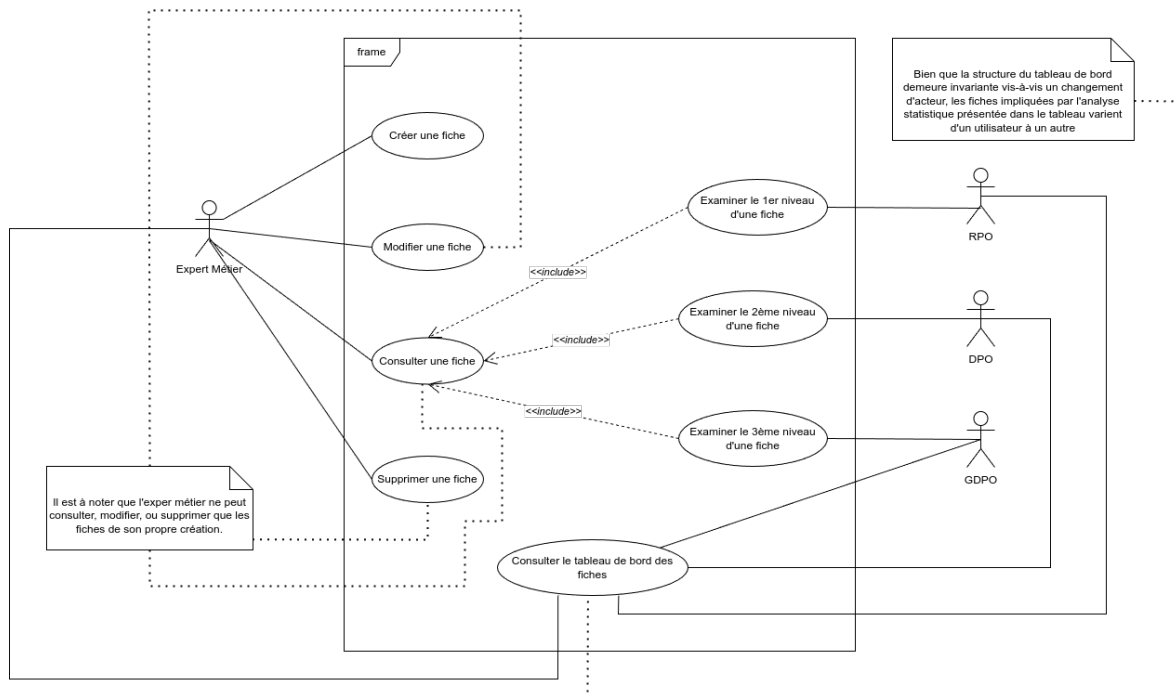


FIGURE 2.1 – Diagramme de cas d'utilisation global simplifié

2.4.2 Diagramme de cas d'utilisation de l'Expert métier

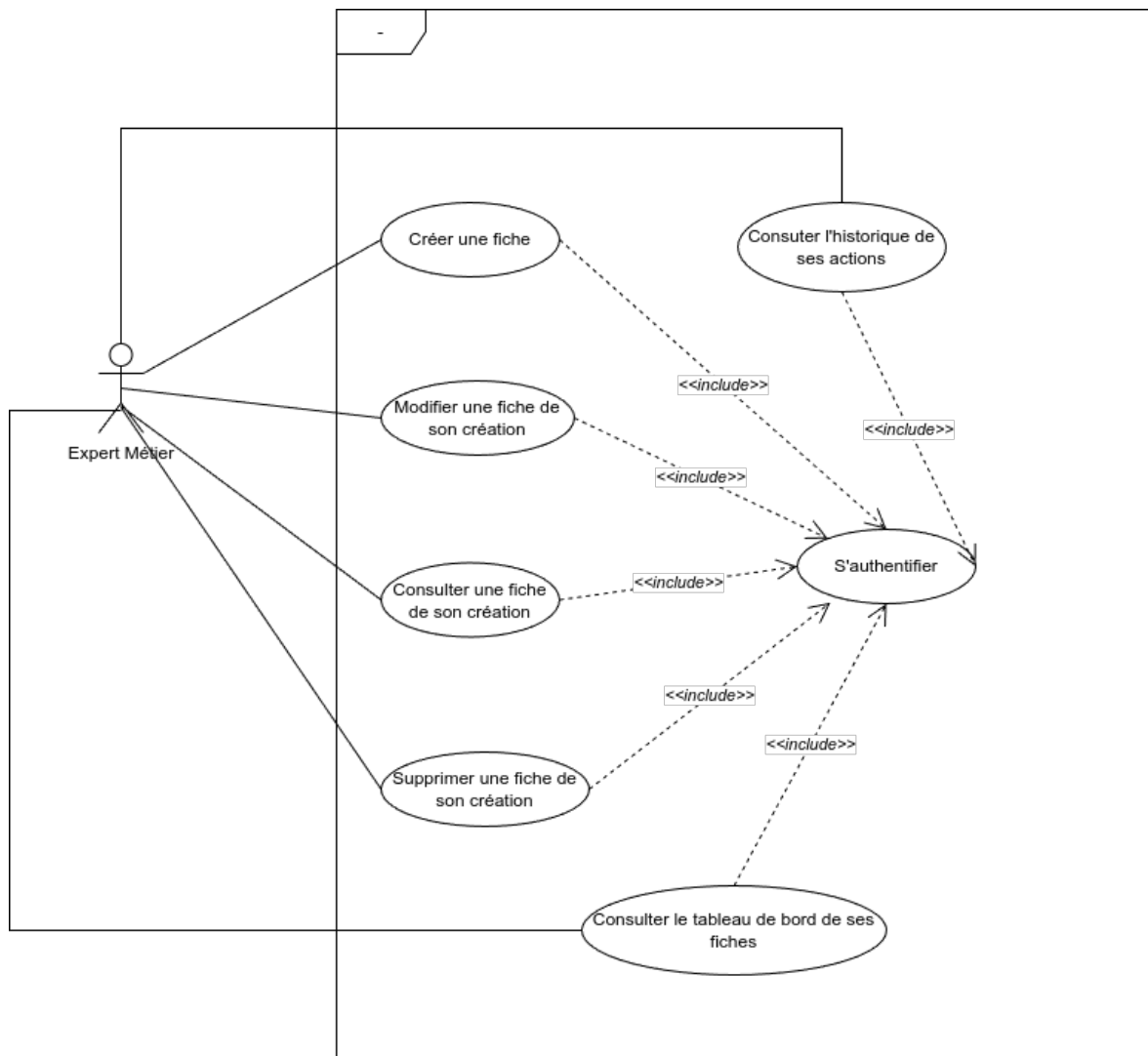


FIGURE 2.2 – Diagramme de cas d'utilisation de l'expert métier

2.4.3 Diagramme de cas d'utilisation du RPO

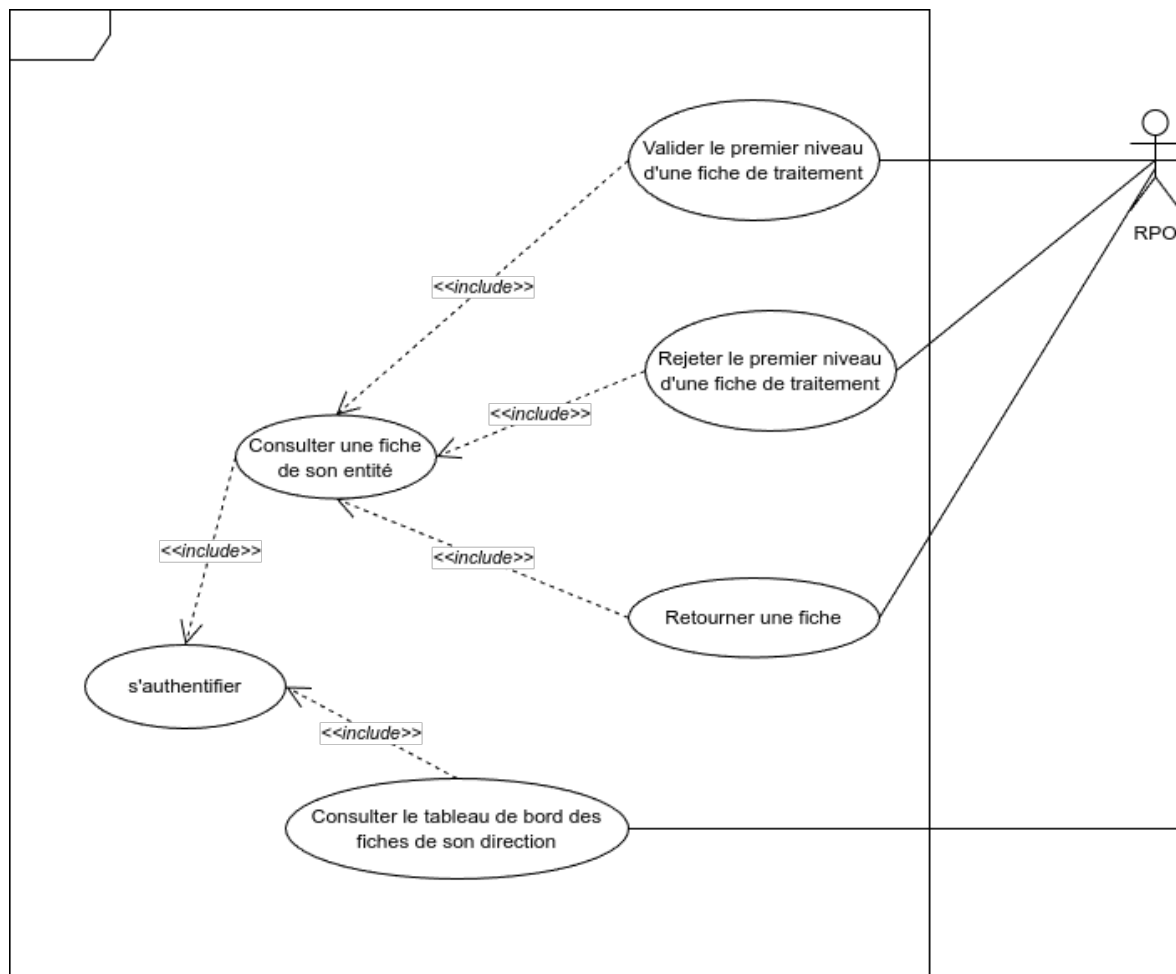


FIGURE 2.3 – Diagramme de cas d'utilisation du RPO

2.4.4 Diagramme de cas d'utilisation du DPO

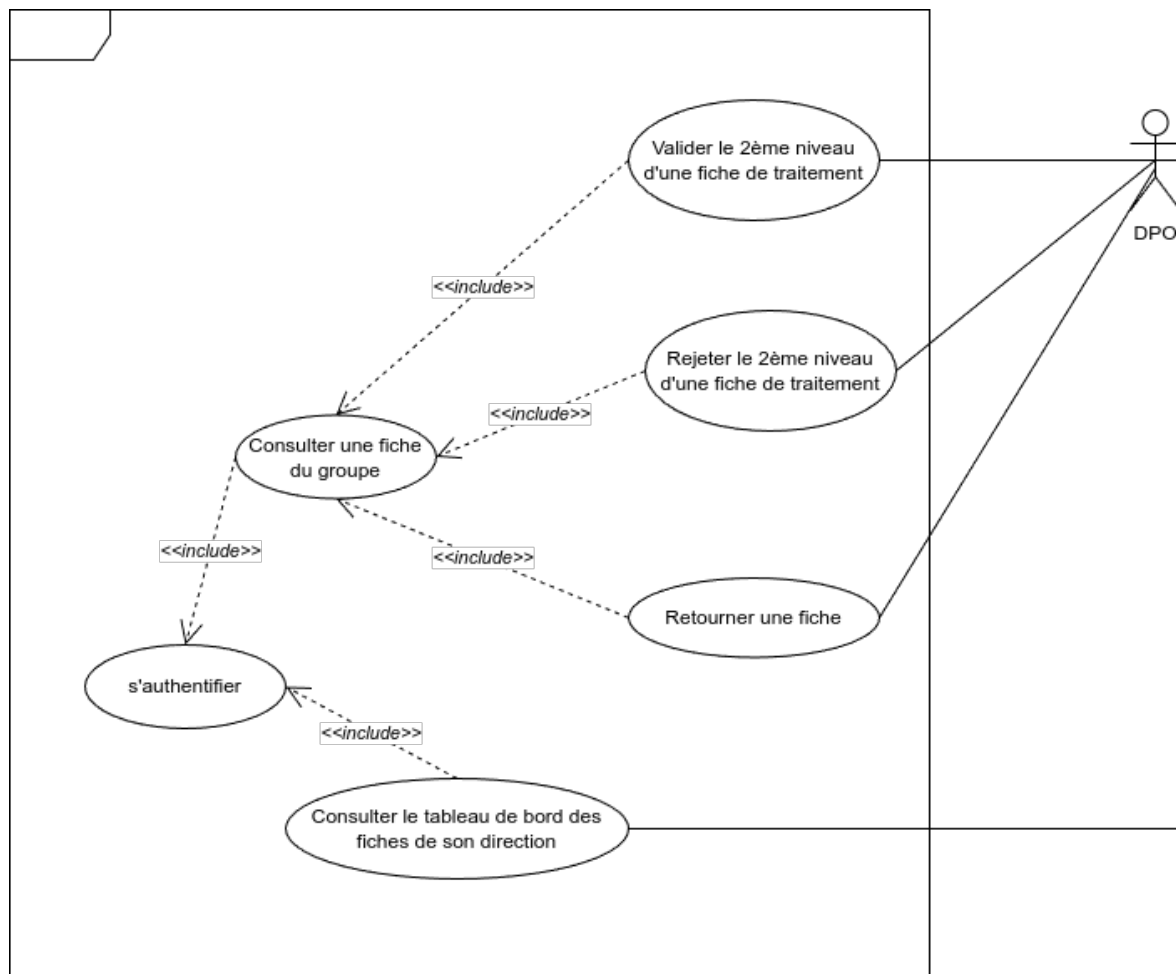


FIGURE 2.4 – Diagramme de cas d'utilisation du DPO

2.4.5 Diagramme de cas d'utilisation du GDPO

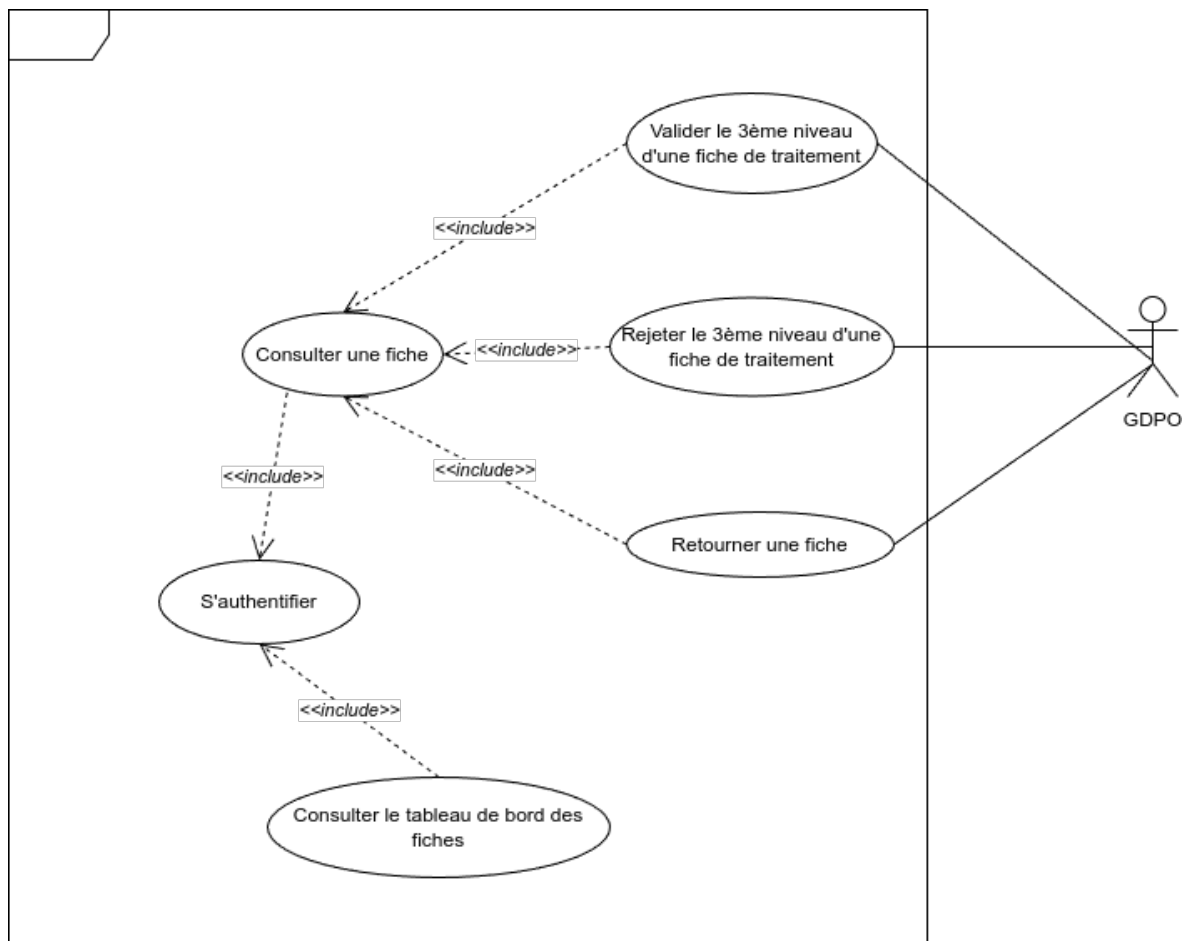


FIGURE 2.5 – Diagramme de cas d'utilisation du GDPO

2.4.6 Diagramme de cas d'utilisation de l'agent d'habilitation

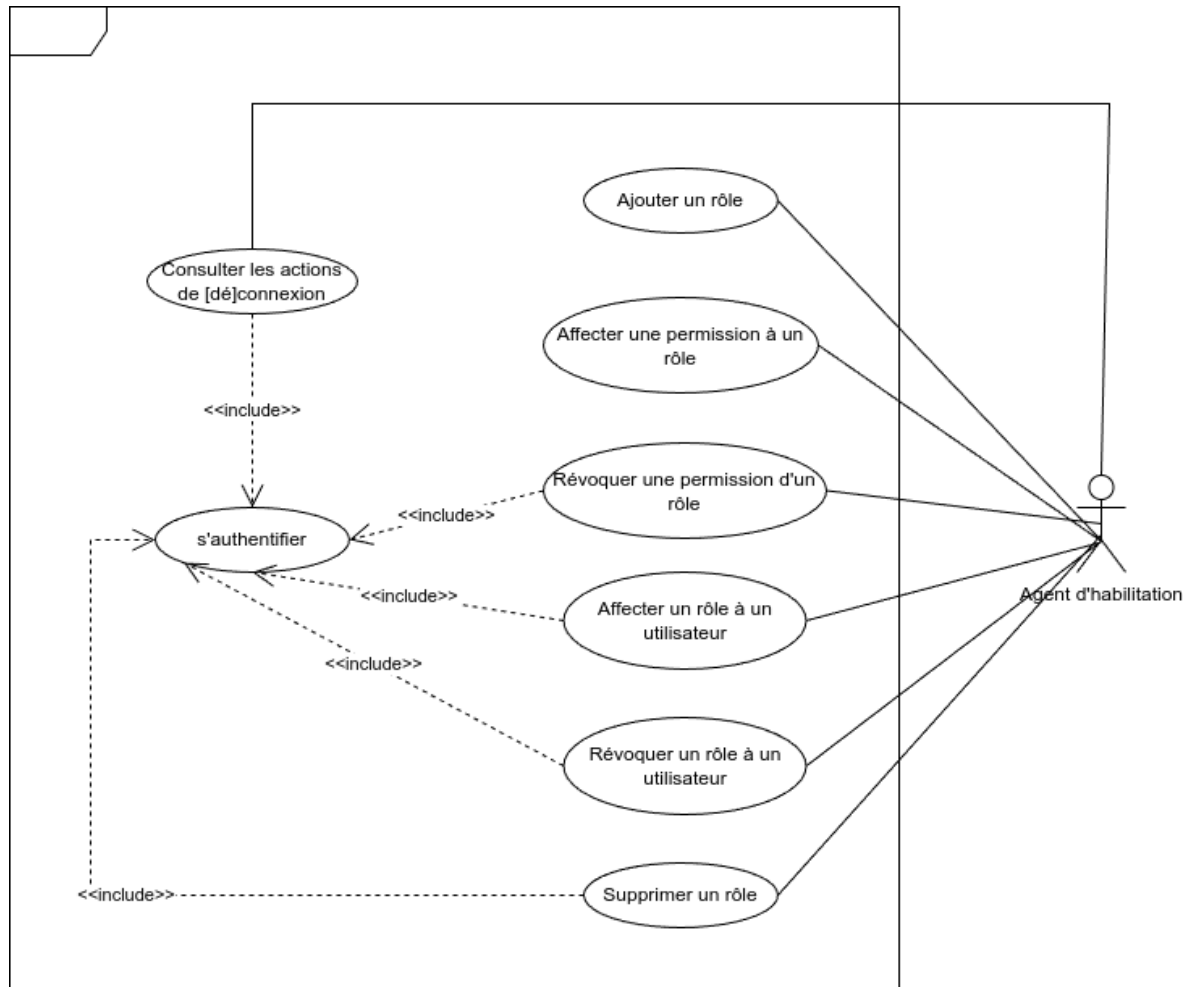


FIGURE 2.6 – Diagramme de cas d'utilisation de l'agent d'habilitation

Chapitre 3

Conception de la solution

3.1 Modélisation C4

Dans le cadre de la conception et de la documentation des architectures logicielles, il est impératif de disposer de méthodes claires et structurées pour représenter les différents aspects du système. La méthode de modélisation C4 (Context, Container, Component, Code), développée par Simon Brown, répond à ce besoin en fournissant un ensemble de diagrammes permettant de capturer et de communiquer efficacement l'architecture d'un système logiciel à différents niveaux de détail.

La méthode C4 se distingue par son approche hiérarchique et progressive, qui permet de passer d'une vue d'ensemble du système à une vue détaillée des composants individuels. Elle se compose de quatre niveaux de diagrammes :

- **Diagramme de contexte** : ils montrent le système dans son environnement, incluant ses relations avec les utilisateurs et d'autres systèmes ;
- **Diagramme de conteneurs** : ils illustrent la décomposition du système en conteneurs interdépendants, chaque conteneur étant un sous-système exécutable déployable indépendamment ;
- **Diagramme de composants** : ils détaillent la décomposition des conteneurs en composants interdépendants et, si nécessaire, leurs liens avec d'autres conteneurs ou systèmes ;
- **Diagramme de code** : ils fournissent des détails supplémentaires sur la conception des éléments architecturaux, souvent en utilisant des notations existantes comme UML, des modèles entité-association ou des diagrammes générés par des environnements de développement intégrés (IDE).

3.2 Diagramme de contexte

Le diagramme de contexte est une représentation visuelle des interactions entre un système et son environnement. Au centre du diagramme se trouve le système lui-même, représenté par un bloc, entouré par des acteurs externes ou des systèmes avec lesquels il interagit. Ces acteurs externes peuvent être des personnes, des organisations, des systèmes informatiques, etc. Les interactions entre le système central et les acteurs externes sont généralement représentées par des flèches indiquant le flux d'informations, de données ou de contrôle entre eux.

La solution d'automatisation du registre de traitement

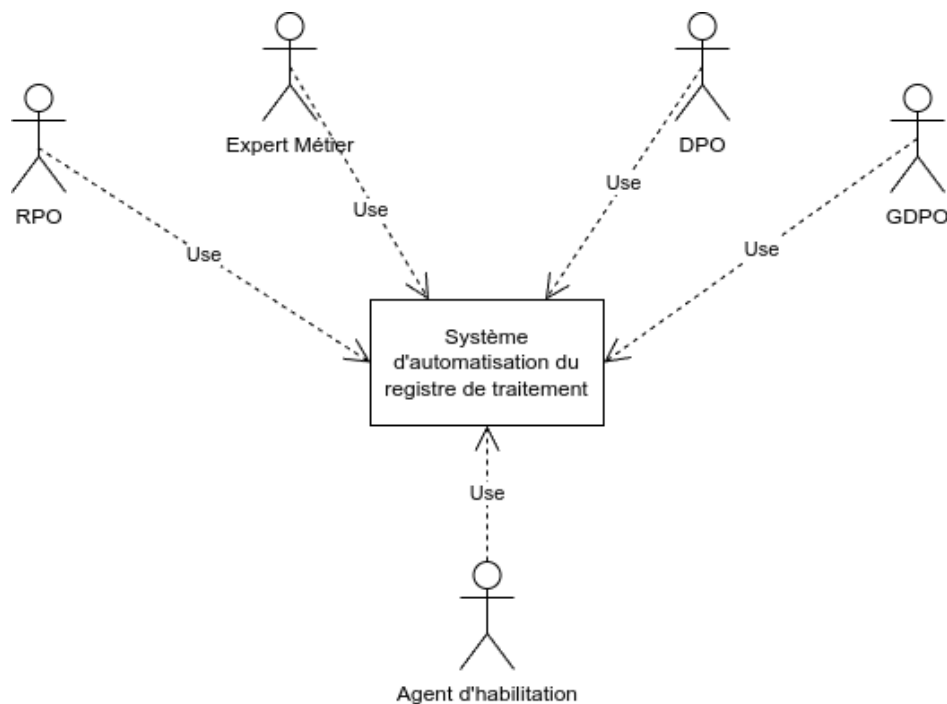


FIGURE 3.1 – Diagramme de contexte

3.3 Diagramme de conteneurs

Le diagramme des conteneurs de la solution d'automatisation du registre de traitement comprend trois éléments principaux :

- **Application web monopage (SPA)** : Permet aux utilisateurs d'interagir avec les fonctionnalités sans rechargement de page. Gère la visualisation, la saisie et la modification des données ;
- **Serveur d'API** : Traite les requêtes HTTP de l'application web. Gère la logique métier, la validation des données, et la gestion des utilisateurs et des autorisations ;
- **Base de données** : Stocke de manière sécurisée toutes les données de l'application. Assure la persistance et la sécurité des données, incluant les fiches de traitement, les informations sur les utilisateurs et les journaux d'activité.

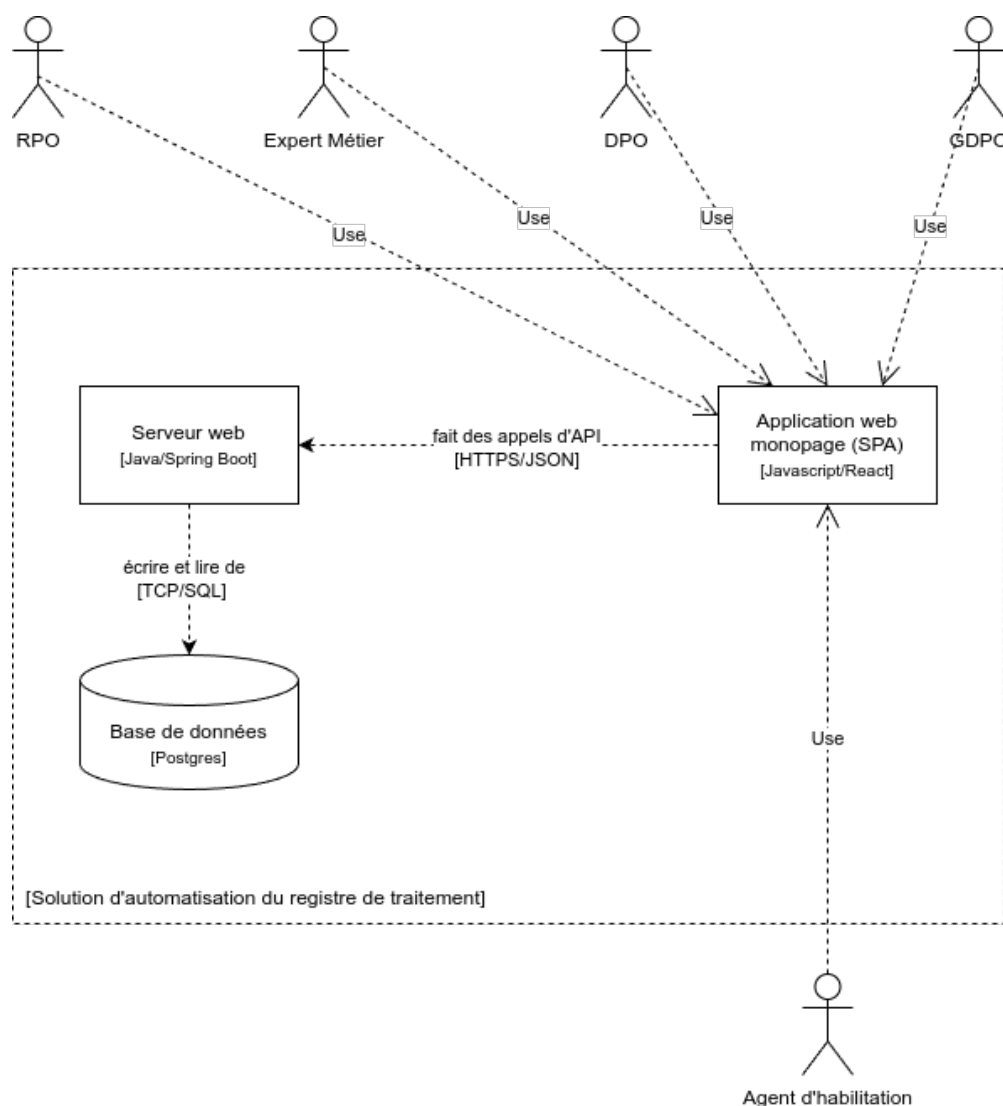


FIGURE 3.2 – Diagramme de conteneurs

3.4 Diagramme des composants du serveur web

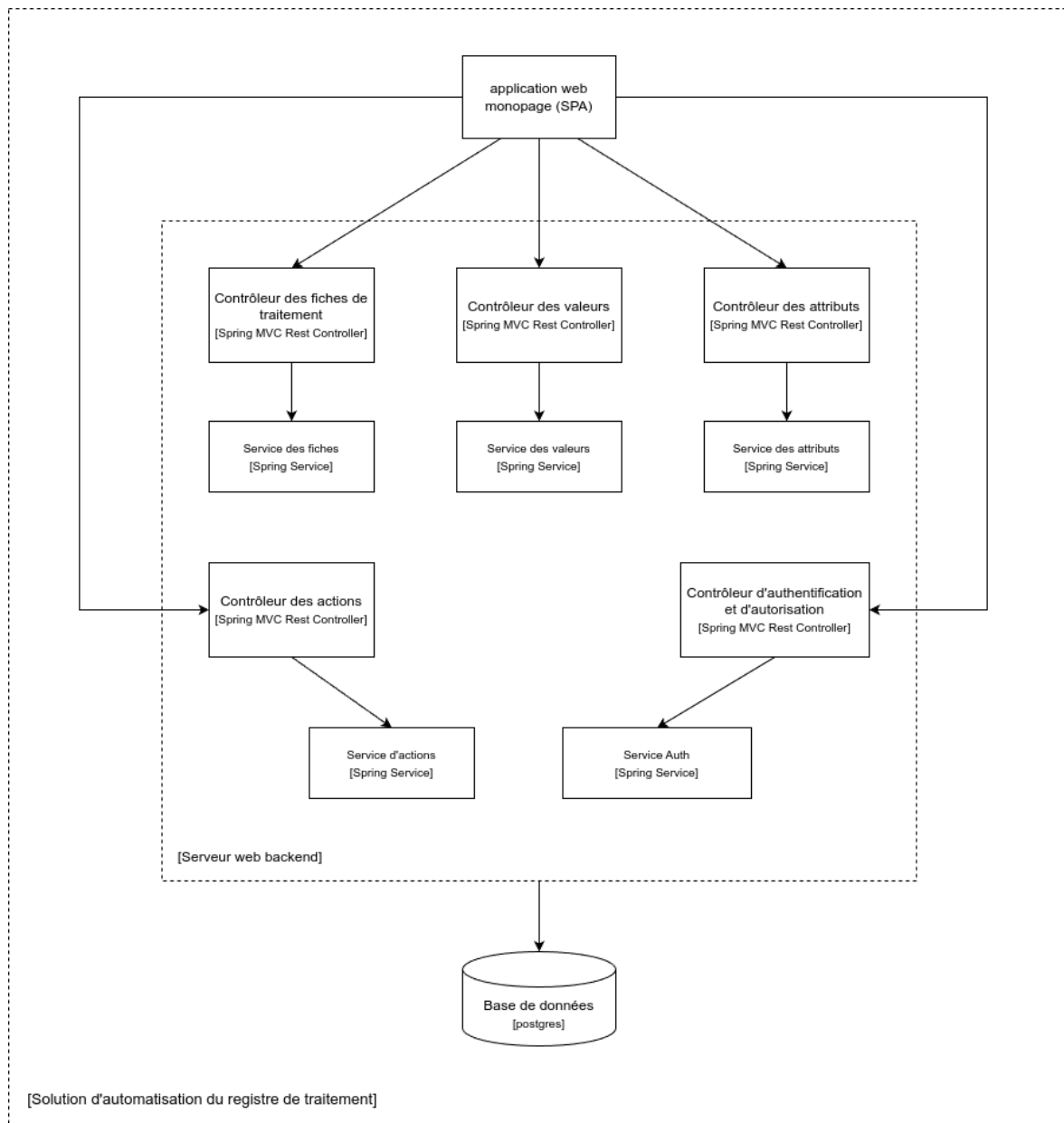
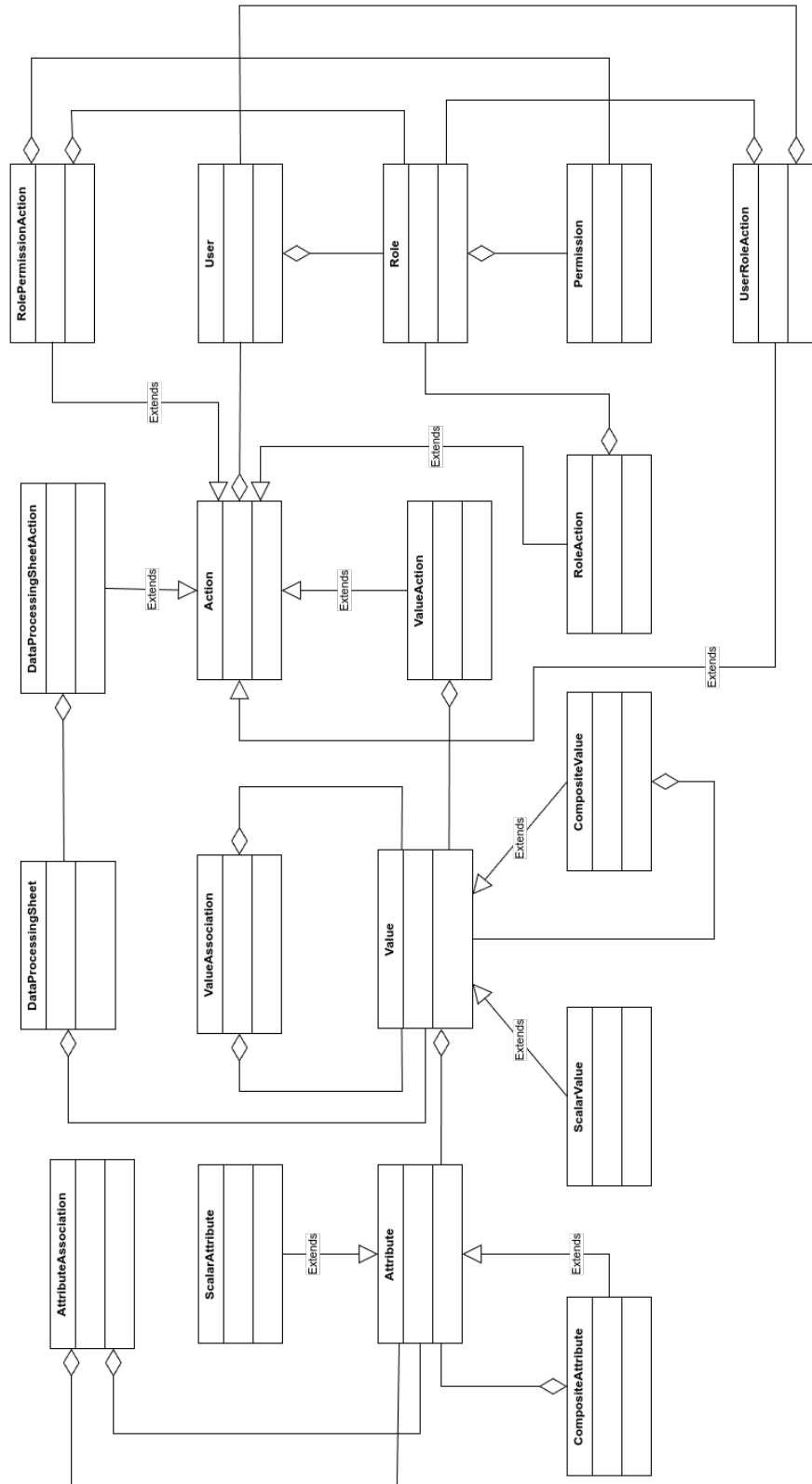


FIGURE 3.3 – Diagramme des composants

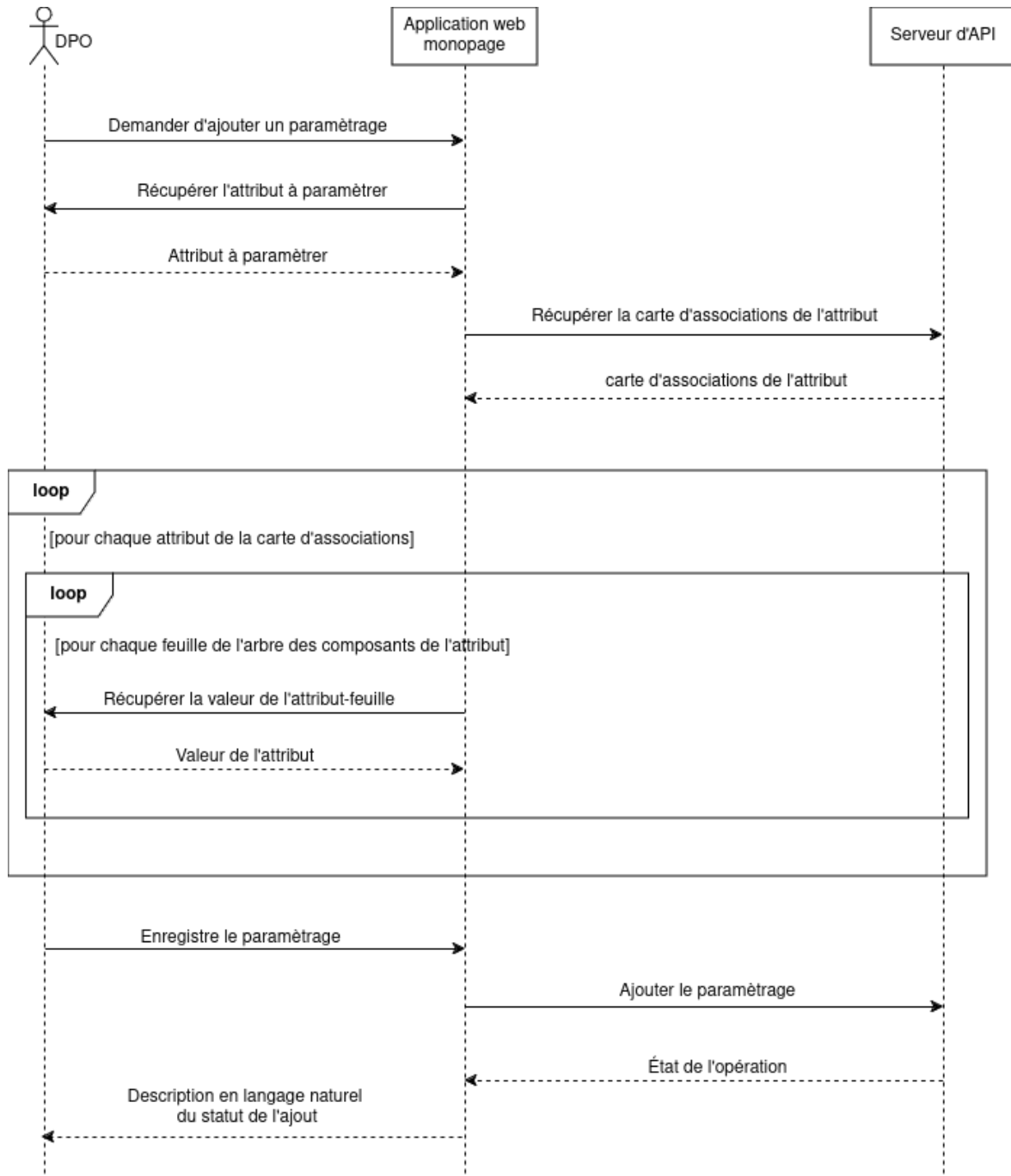
3.5 Diagrammes de classes du serveur d'API

3.5.1 Couche modèle



3.6 Diagrammes de séquences

3.6.1 Ajout d'un paramétrage



Chapitre 4

Réalisation

4.1 Outils de réalisation

4.1.1 Spring

Spring est un framework open source pour construire et définir l'infrastructure d'une application Java, dont il facilite le développement et les tests.



FIGURE 4.1 – Logo de Spring

Spring s'appuie principalement sur l'intégration de trois concepts clés :

- L'inversion de contrôle, assurée de deux façons différentes : la recherche de dépendances et l'injection de dépendances ;
- La programmation orientée aspect ;
- Une couche d'abstraction.

4.1.2 React

React (aussi appelé React.js) est une bibliothèque JavaScript libre. Elle est maintenue par Meta (anciennement Facebook) ainsi que par une communauté de développeurs individuels et d'entreprises depuis 2013.

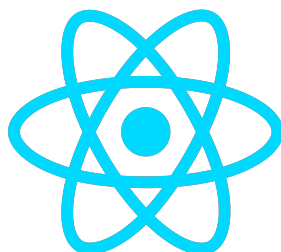


FIGURE 4.2 – Logo de React

Le but principal de cette bibliothèque est de faciliter la création d'application web monopage, via la création de composants dépendant d'un état et générant une page (ou portion) HTML à chaque changement d'état.

4.1.3 Postgres

PostgreSQL est un système de gestion de base de données relationnelle et objet (SGB-DRO). C'est un outil libre disponible selon les termes d'une licence de type BSD.



FIGURE 4.3 – Logo de Postgres

Ce système est comparable à d'autres systèmes de gestion de base de données, qu'ils soient libres (comme MariaDB et Firebird), ou propriétaires (comme Oracle, MySQL, Sybase, DB2, Informix et Microsoft SQL Server). Comme les projets libres Apache et Linux, PostgreSQL n'est pas contrôlé par une seule entreprise, mais est fondé sur une communauté mondiale de développeurs et d'entreprises.

4.1.4 Docker

Docker est une plateforme permettant de lancer certaines applications dans des conteneurs logiciels lancée en 2013.

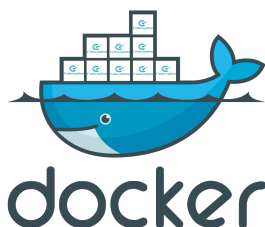


FIGURE 4.4 – Logo de Docker

Selon la firme de recherche 451 Research, Docker est un outil de conteneurisation qui isole une application et ses dépendances dans un conteneur. Contrairement à la virtualisation, il utilise des parties de la machine hôte pour fonctionner, améliorant ainsi la flexibilité et la portabilité de l'application sur différentes machines hôtes, telles que des machines locales, des clouds privés ou publics, ou même des machines nues.

4.1.5 JUnit

JUnit est un framework de test unitaire pour le langage de programmation Java, créé par Kent Beck et Erich Gamma.



FIGURE 4.5 – Logo de JUnit

JUnit définit deux types de fichiers de tests. Les `TestCase` (cas de test) sont des classes contenant un certain nombre de méthodes de tests. Un `TestCase` sert généralement à tester le bon fonctionnement d'une classe. Une `TestSuite` permet d'exécuter un certain nombre de `TestCase` déjà définis.

4.1.6 Mockito

Mockito est un cadre de test open source pour Java, publié sous la licence MIT. Ce cadre permet la création d'objets doubles de test (objets fictifs) dans des tests unitaires automatisés dans le but du développement piloté par les tests (TDD) ou du développement piloté par le comportement (BDD).



FIGURE 4.6 – Logo de Mockito

Le nom et le logo du cadre sont un jeu de mots sur les mojitos, un cocktail à base de rhum, de menthe et de citron vert.

4.1.7 Gradle

Gradle est un moteur de production fonctionnant sur la plateforme Java. Il permet de construire des projets en Java, Kotlin, Scala, Groovy voire C++.



FIGURE 4.7 – Logo de Gradle

Gradle allie les atouts de Apache Maven et Apache Ant : il allie l'utilisation de conventions à la manière de Maven (convention plutôt que configuration) avec la flexibilité de Ant pour décrire les tâches de construction, avec une cohérence forte dans l'interface de programmation des tâches.

4.1.8 Checkstyle

Checkstyle est un outil d'analyse statique de code utilisé dans le développement de logiciels pour vérifier si un code source Java est conforme vis-à-vis un nombre de règles.



FIGURE 4.8 – Logo de Checkstyle

L'outil a été initialement développé par Oliver Burn en 2001 et est désormais maintenu par une équipe internationale de développeurs bénévoles. Cela garantit que Checkstyle évolue continuellement pour répondre aux besoins de la communauté Java.

4.1.9 PMD



FIGURE 4.9 – Logo de PMD

PMD est un outil d'analyse statique de code. Il peut être utilisé pour détecter de possibles erreurs de programmation, vérifier les règles d'un style de programmation, ou mesurer des indicateurs de qualité de code, comme des mesures de complexité. L'analyse produit un rapport lisible par le programmeur.

4.1.10 Git



FIGURE 4.10 – Logo de Git

Git est un logiciel de gestion de versions décentralisé. C'est un logiciel libre et gratuit, créé en 2005 par Linus Torvalds, auteur du noyau Linux, et distribué selon les termes de la licence publique générale GNU version 2. Le principal contributeur actuel de Git, et ce depuis plus de 16 ans, est Junio C Hamano.

4.1.11 Github

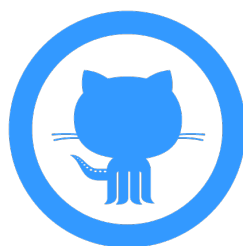


FIGURE 4.11 – Logo de Github

GitHub est un service web d'hébergement et de gestion de développement de logiciels, utilisant le logiciel de gestion de versions Git. Le site est développé en Ruby on Rails et Erlang par Chris Wanstrath, PJ Hyett et Tom Preston-Werner. GitHub propose des comptes professionnels payants, ainsi que des comptes gratuits pour les projets de logiciels libres.

4.1.12 Github Actions

GitHub Actions est un service d'automatisation intégré à GitHub, permettant de créer des workflows pour gérer les processus de développement logiciel. Ces workflows sont définis dans des fichiers YAML et peuvent inclure des actions prédéfinies ou personnalisées pour automatiser des tâches telles que les tests, la construction et le déploiement d'applications. Cela simplifie l'intégration continue (CI) et le déploiement continu (CD) directement depuis les dépôts GitHub.

4.2 Interfaces graphiques

4.2.1 Gestion des paramètres

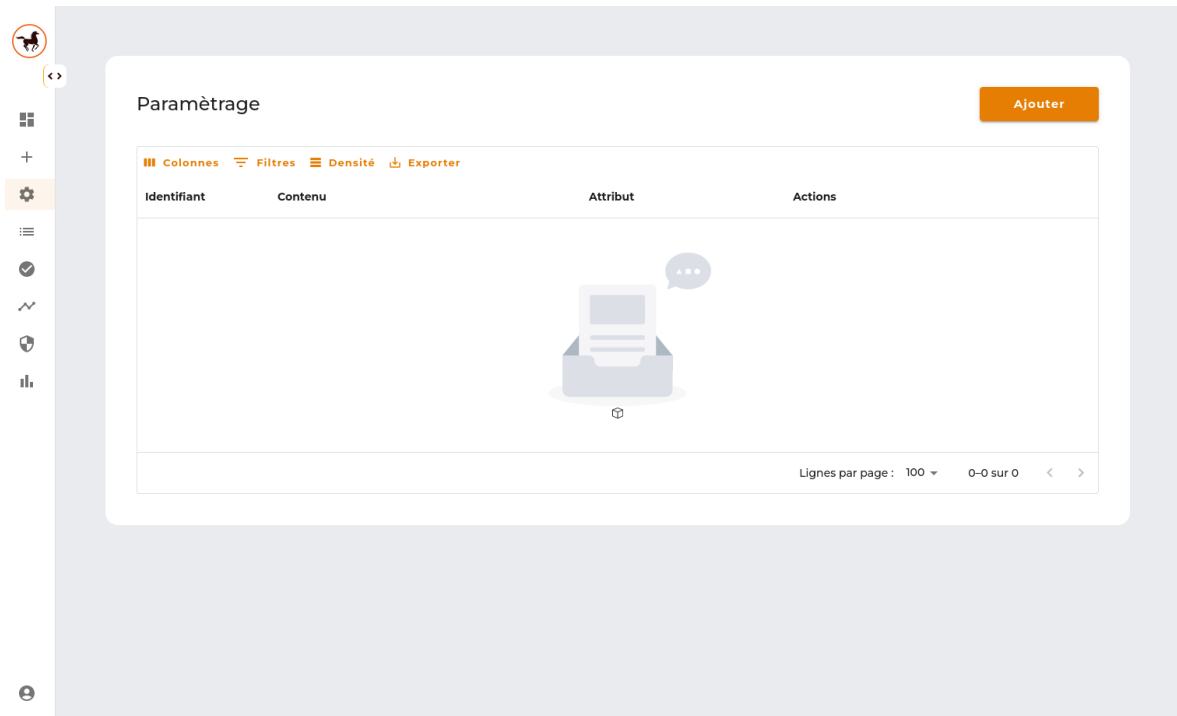


FIGURE 4.12 – Page de paramétrage

4.2.2 Ajout d'une fiche de traitement

BANQUE POPULAIRE
GRANDS ENSEMBLES

Tableau de bord
+ Nouvelle fiche
Paramétrage
Liste des fiches
Validation
Traçabilité
Habilitation
Reporting

Najib Sadki

1 Identification de la fiche
2 Données traitées
3 Catégorie des données
4 Transferts
5 Mesures de sécurité

Nom du traitement *

Responsable du traitement
Banque Centrale Populaire

Représentant du traitement

Responsable conjoint

Direction concernée *

Logiciel *

Catégorie de personnes concernées ⓘ

Numéro de récépissé CNDP ⓘ

Base légale

Délibération par la CNDP

Date de récépissé CNDP
dd/mm/yyyy

Date de délibération par la CNDP
dd/mm/yyyy

Finalité +

Suivant

FIGURE 4.13 – Page d'ajout d'une fiche de traitement

4.2.3 Liste des fiches de traitement

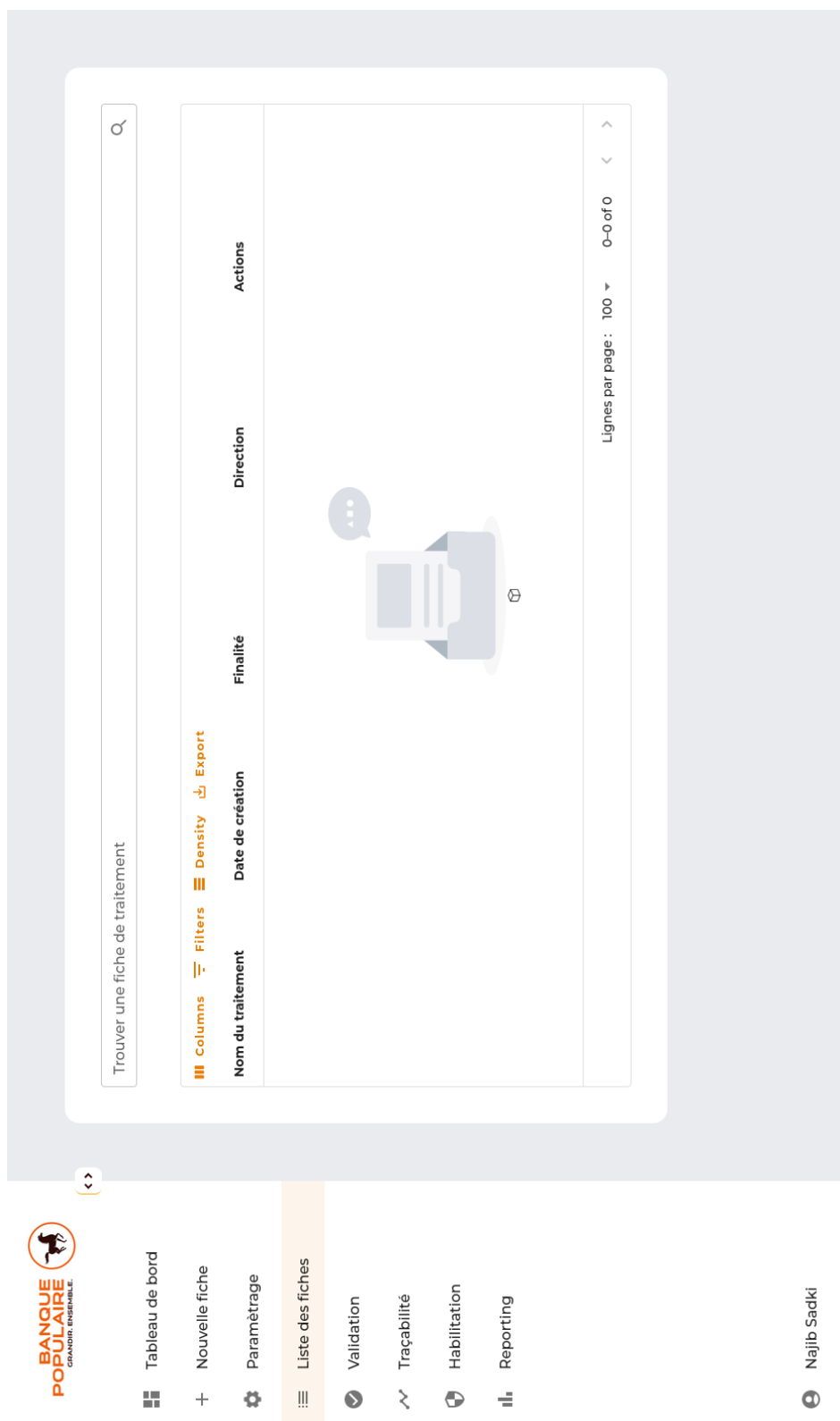


FIGURE 4.14 – Page de la liste des fiches de traitement

Conclusion Finale

Conclusion et perspectives



Bibliographie