

République Algérienne Démocratique et Populaire  
Ministère de l'Enseignement Supérieur et de la Recherche  
Scientifique



جامعة بجاية  
Tasdawit n Bgayet  
Université de Béjaïa



Groupement Reggane

Université A. MIRA – Béjaïa

Groupement Reggane Nord

Faculté des Sciences Exactes  
Département d'Informatique

## Mémoire de fin de cycle

En vue de l'obtention du diplôme de Master Professionnel en  
Informatique  
Option : Génie Logiciel

---

# Conception d'un système de gestion et d'analyse des données de plateformes gazières

---

*Réalisé par :*  
TIKHERBATINE Kenza

*Encadré par :*  
BEDJOU Khaled

**Devant le jury composé de :**  
AIT ABDELOUHAB Karima (Présidente)  
DJEBARI Nabil (Examinateur)  
CHIBANI Samia (Examinatrice)  
CHABANE Sarah (Examinatrice)

Année universitaire : 2024/2025

# Table des matières

<b>Table des matières</b>	<b>1</b>
<b>Table des figures</b>	<b>4</b>
<b>Introduction Générale</b>	<b>9</b>
<b>1 Etude et analyse de l'existant</b>	<b>12</b>
1.1 Introduction . . . . .	12
1.2 Représentation de groupement reggane nord . . . . .	12
1.2.1 Localisation du projet (GRN) . . . . .	13
1.2.2 CPF (Central Processing Facility) . . . . .	14
1.2.3 Organisation fonctionnelle des services techniques au sein du CPF . . . . .	15
1.2.4 Présentation des services techniques du CPF . . . . .	16
1.3 Processus de stockage des données des capteurs . . . . .	17
1.3.1 Capteurs . . . . .	18
1.3.2 Process History Database (PHD) . . . . .	20
1.4 Architecture réseau de communication entre les systèmes industriels et le serveur PHD . . . . .	20
1.5 Etude des solutions existantes . . . . .	22
1.5.1 WIMS (WellMaster Integrity Management System) . . . . .	22
1.5.2 Synergi Life . . . . .	23
1.5.3 SAP MM (Materials Management) . . . . .	25
1.5.4 UPS (Uniformance Process Studio) . . . . .	25
1.5.5 AVOCET . . . . .	26
1.6 Problématique . . . . .	27
1.7 Objectifs . . . . .	27
1.8 Solution envisagée . . . . .	28
1.9 Méthodologie de conception . . . . .	29
1.9.1 Langage de Modélisation Unifié (UML) . . . . .	29
1.9.2 Processus Unifié (UP) . . . . .	30
1.10 Pertinence d'UML et de l'UP dans le projet . . . . .	31

1.11	Illustration par des maquettes fonctionnelles . . . . .	32
1.12	Conclusion . . . . .	34
<b>2</b>	<b>Phase de Conception</b>	<b>35</b>
2.1	Spécification des besoins . . . . .	36
2.1.1	Besoins fonctionnelles . . . . .	36
2.1.2	Besoins non fonctionnelles . . . . .	38
2.2	Acteurs et leur cas d'utilisation . . . . .	39
2.2.1	Acteurs du système . . . . .	39
2.2.2	Cas d'utilisation . . . . .	40
2.3	Gestion des utilisateurs . . . . .	44
2.3.1	Diagramme de cas d'utilisation . . . . .	45
2.3.2	Diagramme de séquence "Créer un utilisateur" . . . . .	46
2.3.3	Diagramme de classe . . . . .	47
2.4	Gestion des puits . . . . .	48
2.4.1	Diagramme de cas D'utilisation . . . . .	48
2.4.2	Diagramme de Séquence "Rechercher un puit" . . . . .	49
2.4.3	Diagramme de classe . . . . .	50
2.5	Gestion de production . . . . .	51
2.5.1	Diagramme de cas d'utilisation . . . . .	51
2.5.2	Diagramme de séquence "saisir une production" . . . . .	52
2.5.3	Diagramme de classe . . . . .	53
2.6	Gestion des analyses de gaz . . . . .	54
2.6.1	Diagramme de cas d'utilisation . . . . .	54
2.6.2	Diagramme de séquence . . . . .	55
2.6.3	Diagramme de classe . . . . .	56
2.7	Le modèle relationnel . . . . .	56
<b>3</b>	<b>Realisation de l'application</b>	<b>59</b>
3.1	Environnement et outils de developpement . . . . .	60
3.1.1	Outils de communication . . . . .	60
3.1.2	Outils de conception et prototypage . . . . .	60
3.1.3	Environnement de codage . . . . .	61
3.1.4	Langages et frameworks . . . . .	61
3.1.5	Bibliothèques . . . . .	62
3.1.6	Base de données . . . . .	63
3.2	Charte graphique . . . . .	64
3.2.1	Identité de marque . . . . .	64
3.2.2	Couleurs . . . . .	64
3.2.3	Typographie . . . . .	64

3.2.4	Logo . . . . .	65
3.3	Présentation de la plateforme . . . . .	65
3.3.1	L'interface utilisateur "Interface de connexion" . . . . .	65
3.3.2	L'interface utilisateur "Interface de gestion des utilisateurs" . . . . .	66
3.3.3	L'interface utilisateur "Interface de saisie de production journalière" . . . . .	66
3.3.4	L'interface utilisateur "Interface de saisie des prévisions journalière " . . . . .	67
3.3.5	L'interface utilisateur "Interface de visualisation des KPI" . . . . .	68
3.3.6	L'interface utilisateur "Interface de saisir des analyses labo" . . . . .	69
3.4	Conclusion . . . . .	69
<b>Conclusion Générale</b>		<b>70</b>
<b>Bibliographie</b>		<b>72</b>
<b>ANNEXE</b>		<b>75</b>

# Table des figures

1.1	Partenaires du GRN : Sonatrach, Repsol et Wintershall Dea . . . . .	13
1.2	Localisation du projet Reggane Nord sur la carte de l'Algérie . . . . .	14
1.3	Vue de la zone process du CPF (Central Processing Facility) . . . . .	15
1.4	Organisation des départements et services techniques du CPF sous la di- rection des Opérations . . . . .	16
1.5	processus de stockage de données capturées . . . . .	17
1.6	exemple d'un capteur de débit de gaz . . . . .	18
1.7	Automate programmable industriel (PLC) installé sur site, à proximité d'un puit . . . . .	19
1.8	Architecture réseau avec zone DMZ entre le DCS et les applications métiers	21
1.9	Illustration des annulaires installés, des pressions mesurées et du statut du puits. . . . .	22
1.10	État des puits et leur intégrité . . . . .	23
1.11	Interface utilisateur d'accueil de Synergi . . . . .	24
1.12	Les Phases du cycle de vie du processus unifié . . . . .	31
1.13	Interface utilisateur qui affiche les informations generale de puit et celle de la phase forage . . . . .	33
1.14	Interface utilisateur qui affiche les informations d'intégrité de puit et celle de la phase production . . . . .	33
2.1	Diagramme de cas d'utilisation "Gestion des utilisateurs" . . . . .	45
2.2	Diagramme de séquence "créer un utilisateur" . . . . .	46
2.3	Diagramme de classe "Gestion des utilisateurs" . . . . .	47
2.4	Diagramme de cas d'utilisation "Gestion des puits" . . . . .	48
2.5	Diagramme de Séquence "Rechercher un puit" . . . . .	49
2.6	Diagramme de classe "Gestion des puits" . . . . .	50
2.7	Diagramme de cas d'utilisation "Gestion de production" . . . . .	51
2.8	Diagramme de séquence "Saisir une production" . . . . .	52
2.9	Diagramme de classe "Gestion de production" . . . . .	53
2.10	Diagramme de cas d'utilisation "Gestion des analyses de gaz" . . . . .	54
2.11	Diagramme de séquence "Ajouter une analyse" . . . . .	55

2.12	Diagramme de classe "Gestion des analyses de gaz" . . . . .	56
3.1	Logo officiel de l'application GasScope . . . . .	65
3.2	Interface utilisateur "Interface de connexion" . . . . .	66
3.3	Interface utilisateur "Interface de gestion des utilisateur" . . . . .	66
3.4	Interface utilisateur "Interface de saisie de production journalière" . . . . .	67
3.5	Interface utilisateur "Interface de saisie de production journalière-suite" . . . . .	67
3.6	Interface utilisateur "Interface de saisie des prévisions journalière" . . . . .	68
3.7	Interface utilisateur "historique des prévisions journalière saisies" . . . . .	68
3.8	Interface utilisateur "Interface de visualisation des KPI" . . . . .	69
3.9	Interface utilisateur "Interface de saisir des analyses labo" . . . . .	69

## Liste des abréviations

- **XP** : Exploitation
- **EP** : Engineering and Production
- **WI** : Well Integrity
- **CPF** : Central Processing Facility
- **DCS** : Distributed Control System
- **UPS** : Uniformance Process Studio
- **PHD SCADA** : Process History Database SCADA
- **HSE** : Health, Safety, and Environment
- **CPF** : Central Processing Facility

# Remerciements

Avant toute chose, je rends grâce à Dieu Tout-Puissant pour le don de la vie, pour la force et la patience qu'Il m'a accordées tout au long de ce parcours, ainsi que pour toutes les prières exaucées qui m'ont permis d'atteindre cet objectif.

Je tiens à exprimer toute ma reconnaissance et ma profonde gratitude à mon encadrant universitaire, M. BEDJOU Khaled, pour son accompagnement, ses conseils précieux et sa disponibilité tout au long de ce travail.

Mes remerciements s'adressent également à l'entreprise GRN, qui m'a accueillie et m'a offert l'opportunité d'effectuer ce stage en son sein. Je tiens particulièrement à remercier M. Mohamed Berket, ingénieur au département EP, et M. Talbi Abdelhamid, ingénieur au département XP, pour leur encadrement bienveillant, leurs orientations et leurs conseils enrichissants.

J'adresse aussi mes vifs remerciements à M. Bilal Ben Yahia, ingénieur au sein de l'entreprise, pour ses orientations et son aide qui m'ont grandement facilité l'intégration et l'avancement de mon stage.

Je remercie également les membres du jury pour l'intérêt qu'ils portent à ce travail, ainsi que pour le temps consacré à son évaluation.

Mes pensées reconnaissantes vont à l'ensemble des enseignants et au personnel administratif de notre département, pour la qualité de l'encadrement et leur soutien tout au long de mon parcours universitaire.

Enfin, un merci tout particulier à ma famille, dont l'amour, le soutien et les sacrifices silencieux ont toujours été ma plus grande force. À mes parents, je dis merci du fond du cœur pour leur patience, leurs encouragements et leur confiance inconditionnelle.

# Dédicace

Je dédie ce mémoire avec toute ma gratitude et mon affection à ceux qui ont été présents tout au long de ce parcours.

À mes chers parents, pour leur amour inconditionnel, leur soutien constant et leurs conseils précieux qui m'ont toujours guidée dans mes choix et m'ont donné la force de persévérer.

À mes deux petits frères, **Yasmina** et **Abderraouf**, pour leur affection, leur joie de vivre et leur présence qui m'ont toujours donné de l'énergie et du courage pour avancer dans ce projet.

À mes deux tantes, **Rosa** et **Sabrina**, ainsi qu'à leurs filles, **Lina**, **Amira** et **Lya**, pour leur amour, leur bienveillance et leur présence chaleureuse tout au long de ce projet.

À mes amies, **Thilelli**, **Amel**, **Katia**, **Nour el Houda**, et **Mounia**, pour leur amitié sincère, leurs encouragements, leurs rires partagés et le soutien apporté dans les moments difficiles.

Enfin, à moi-même, pour ma persévérance, ma détermination et ma capacité à surmonter les obstacles rencontrés durant ce long chemin. Ce mémoire est le fruit de ces efforts, de ma patience et de mon engagement personnel.

**TIKHERBATINE Kenza**

# Introduction Générale

Le secteur de l'énergie, et en particulier l'exploitation gazière, est un domaine stratégique qui repose intrinsèquement sur la gestion et l'analyse de vastes quantités de données industrielles pour assurer la sécurité, l'efficacité et la performance des opérations. La capacité à suivre avec précision les processus de production, à prendre des décisions rapides et à optimiser la gestion des ressources est devenue primordiale. Dans ce contexte, les systèmes d'information industriels jouent un rôle central en permettant la collecte, le traitement et l'analyse des données issues des capteurs installés sur les sites de production.

Dans le cadre de ce travail, le stage a été réalisé au sein du Groupement Reggane Nord (GRN), une entité chargée de l'exploitation du champ gazier de Reggane situé dans le sud-ouest algérien. Le GRN assure la supervision complète de la production gazière, depuis l'extraction au niveau des puits jusqu'au traitement au sein du CPF (Central Processing Facility). Pour mener à bien ses missions, l'entreprise s'appuie sur plusieurs systèmes d'information industriels tels qu'AVOCET pour le suivi de la production, Uniformance Process Studio (UPS) pour l'analyse des données de procédé, WIMS pour la gestion d'intégrité des puits, Synergi Life pour la gestion HSE, ou encore SAP MM pour les opérations de maintenance et de logistique. Ces outils constituent l'écosystème numérique du groupement et jouent un rôle central dans la gestion quotidienne des opérations.

Cependant, malgré leur importance, les outils actuellement en place au sein d'entités comme le Groupement Reggane Nord (GRN), tels qu'AVOCET, Uniformance Process Studio (UPS), WIMS, Synergi Life ou SAP MM, présentent des limites significatives. Ces faiblesses incluent l'absence d'automatisation des indicateurs de performance (KPI), des processus manuels fastidieux pour l'extraction et l'analyse des données, des interfaces utilisateur surchargées et peu intuitives, ainsi que des restrictions d'accès limitant la collaboration entre les ingénieurs. Ces contraintes opérationnelles entraînent une perte de temps considérable, une surcharge de travail et un risque accru d'erreurs humaines, rendant indispensable la recherche d'une solution plus adaptée aux besoins spécifiques des équipes techniques du CPF (Central Processing Facility).

C'est dans cette optique que s'inscrit le présent projet de fin d'études, dont l'objectif principal est la conception d'une solution logicielle visant à surmonter les insuffisances des systèmes existants. Plus précisément, notre travail vise à automatiser le calcul et l'affichage des KPI, à offrir une visualisation claire, dynamique et filtrable des données de production, à proposer une interface utilisateur ergonomique et centrée sur les besoins des ingénieurs (XP, EP, Mesure, Intégrité), à réduire la dépendance aux extractions manuelles, à faciliter la prise de décision rapide et à garantir une gestion sécurisée des accès. La solution envisagée, nommée **GasScope**, se veut être une plateforme moderne, automatisée et ergonomique, optimisant l'extraction et l'analyse des données de production.

Le présent mémoire est structuré en trois chapitres distincts pour présenter notre démarche :

- Le premier chapitre, intitulé *Étude et analyse de l'existant*, permettra de poser les fondations du travail. Il présentera le contexte du Groupement Reggane Nord (GRN), les services techniques du CPF, les processus de stockage des données issues des capteurs via les PLC, DCS et PHD, l'architecture réseau de communication, ainsi qu'une analyse critique des solutions logicielles existantes. Ce chapitre aboutira à la formulation d'une problématique claire, à la définition des objectifs du projet et à la proposition de maquettes fonctionnelles initiales.
- Le deuxième chapitre, intitulé *Spécification des besoins et Conception*, sera dédié à la modélisation de notre solution. Il démarrera par une spécification détaillée des besoins fonctionnels et non fonctionnels de la plateforme, avant de présenter la méthodologie de conception basée sur l'Unified Modeling Language (UML) et l'Unified Process (UP), en justifiant leur choix. Ce chapitre identifiera les acteurs du système et leurs cas d'utilisation, et modélisera l'architecture du système en packages distincts, illustrés par des diagrammes de cas d'utilisation, de classes et de séquences, ainsi que par le modèle relationnel de la base de données.
- Le troisième chapitre, consacré à la *Réalisation*, détaillera l'environnement et les outils de développement utilisés (outils de communication comme Notion et Google Meet, de conception comme Draw.io et Figma, l'environnement de codage VS Code), les langages et frameworks (Django, React, Tailwind CSS, i18next), les bibliothèques logicielles mobilisées (React Toastify, react-country-flag, Print JS, Heroicons, xlsx, jsPDF), et la base de données PostgreSQL avec l'ORM de Django. Ce chapitre inclura également la charte graphique de notre logo et de l'application, ainsi qu'une présentation détaillée de l'application avec des captures d'écran explicatives. Ce travail s'achèvera par une conclusion générale qui synthétisera les apports de notre projet et ouvrira des perspectives pour ses dé-

veloppements futurs.

# Chapitre 1

## Etude et analyse de l'existant

### 1.1 Introduction

Le secteur de l'énergie, en particulier celui de l'exploitation gazière, repose de plus en plus sur des systèmes d'information industriels pour assurer un suivi précis, une prise de décision rapide et une gestion efficace des opérations. Ces systèmes permettent la collecte, le traitement et l'analyse des données issues des capteurs installés sur les sites de production, tout en facilitant le travail des ingénieurs des différents services.

Toutefois, malgré leur utilité, les outils actuellement en place tels qu'AVOCET, UPS ou encore WIMS, présentent plusieurs limites : accès restreint aux données, manque d'automatisation, interfaces peu ergonomiques ou encore difficultés d'extraction de données ciblées. Ces contraintes ralentissent le travail des ingénieurs et limitent l'exploitation optimale des données de production.

Dans ce chapitre, nous allons présenter l'environnement dans lequel s'inscrit notre étude, en décrivant le Groupement Reggane Nord (GRN), les services impliqués, et les outils actuellement utilisés. L'objectif est d'identifier les forces et les faiblesses de ces systèmes existants, en vue de proposer par la suite une solution plus adaptée aux besoins opérationnels du GRN.

### 1.2 Représentation de groupement reggane nord

Le Groupement Reggane Nord (GRN) est une entité issue d'un partenariat entre Sonatrach, Repsol et Wintershall Dea, ayant pour mission l'exploration, l'exploitation et le traitement du gaz naturel dans le champ de Reggane Nord, situé à environ 1500 km au sud-ouest d'Alger. Opérant sous contrat avec Sonatrach, ce partenariat stratégique permet de combiner expertises et ressources pour assurer une production conforme aux spécifications contractuelles, tout en optimisant les processus au sein du CPF (Central Processing Facility)[17].

La figure suivante 1.1 présente les logos des trois sociétés formant le Groupement Reggane Nord (GRN).



FIGURE 1.1 – Partenaires du GRN : Sonatrach, Repsol et Wintershall Dea

### 1.2.1 Localisation du projet (GRN)

Le champ gazier de Reggane Nord est localisé sur les blocs 351C et 352C, couvrant une surface de 12 217,5 km<sup>2</sup>. Ces blocs se trouvent dans la Wilaya d'Adrar, au sud-ouest de l'Algérie. Ils appartiennent au bassin de Reggane, situé à environ 1500 kilomètres d'Alger.

La figure 1.2 illustre la localisation géographique du projet Reggane Nord sur la carte de l'Algérie.



FIGURE 1.2 – Localisation du projet Reggane Nord sur la carte de l’Algérie

### 1.2.2 CPF (Central Processing Facility)

Le **CPF (Central Processing Facility)** constitue la zone industrielle principale du Groupement Reggane Nord (GRN), dédiée au traitement du gaz naturel extrait des différents puits de la région. Il regroupe l’ensemble des unités nécessaires pour garantir la qualité, la sécurité et la conformité du gaz avant son exportation.

Les installations du CPF sont composées de plusieurs unités fonctionnelles, parmi lesquelles :

- \* Réception du gaz brut,
- \* Unités de traitement du gaz (séparation triphasique, absorption à l’amine, déshydratation, etc.),
- \* Unité de suppression des gaz acides,
- \* Unité de séparation des composés lourds,
- \* Unité de stabilisation des condensats,
- \* Unité de compression et de comptage du gaz produit,
- \* Système de traitement des eaux résiduelles,
- \* Utilités : fuel gaz, air comprimé, azote, eau déminéralisée, injection de produits chimiques, production d’électricité, stockage gasoil, générateur de secours.

La figure 1.3 présente une vue générale de la **zone process du CPF**, où sont regroupées les principales installations assurant le traitement complet du gaz naturel au sein du GRN.



FIGURE 1.3 – Vue de la zone process du CPF (Central Processing Facility)

### 1.2.3 Organisation fonctionnelle des services techniques au sein du CPF

Le **CPF** du GRN est placé sous la responsabilité de la **Direction des Opérations**, qui coordonne l'ensemble des activités techniques liées à la production, au suivi du processus gazier et à la maintenance des installations.

Cette direction supervise trois départements principaux au sein du CPF :

- **Le département EP** : responsable du suivi de la production, de l'intégrité des puits et des interventions. Il inclut les services Mesure, Well Integrity et Intervention.
- **Le département XP** : chargé du suivi du procédé de traitement du gaz. Il regroupe notamment les services XP et Labo.
- **Le département Maintenance** : dédié à la maintenance des équipements et infrastructures, avec trois services spécialisés : Instrumentation, Méthodes, Électricité et Mécanique.

La figure 1.4 illustre cette organisation hiérarchique et fonctionnelle des services techniques qui interviennent quotidiennement au sein du CPF.

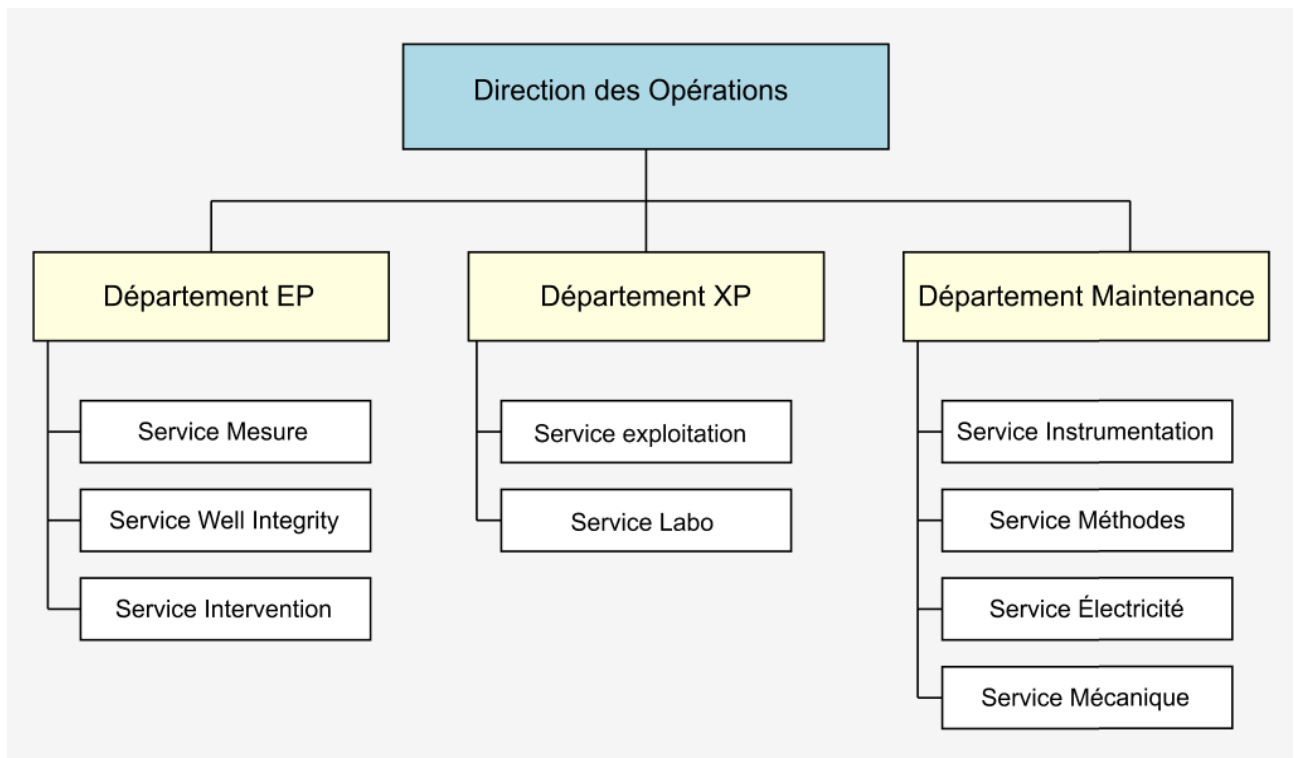


FIGURE 1.4 – Organisation des départements et services techniques du CPF sous la direction des Opérations

### 1.2.4 Présentation des services techniques du CPF

Le CPF comprend plusieurs services répartis entre les départements EP, XP et Maintenance, comme indiqué dans l'organigramme précédent. Dans le cadre de ce travail, seuls les services directement impliqués dans le déroulement du stage sont présentés. Il s'agit des services suivants : Mesure, Well Integrity , XP et Labo.

#### service Mesure

Le service Mesure est chargé du suivi et du contrôle des mesures des flux de production de gaz. Il garantit la fiabilité des données collectées pour assurer une gestion précise des volumes produits.

#### Service Well Integrity (WI)

Le **WI** joue un rôle crucial dans la gestion et le suivi de l'intégrité des puits afin d'assurer leur sécurité et leur performance tout au long de leur cycle de vie. Il est responsable de la surveillance de l'intégrité des puits, Gestion des risques liés aux puits et l'intervention en cas d'anomalies.

## Service XP

Le service Exploitation assure le suivi quotidien du traitement du gaz, surveillant les débits, la qualité du gaz, la production de condensat et d'eau, ainsi que la consommation énergétique. Il contrôle également la production des puits et s'assure de la conformité du gaz aux spécifications contractuelles. En cas de dépassement des seuils réglementaires, l'ingénieur XP prend des mesures correctives pour stabiliser les paramètres, comme en cas d'excès de H<sub>2</sub>S<sup>1</sup>.

## Service Labo

Le service LABO, rattaché au service XP, joue un rôle crucial dans le suivi et le contrôle de la qualité du gaz à différentes étapes du processus de production. Il effectue des analyses quotidiennes sur des échantillons prélevés à sept points stratégiques du circuit gazier pour garantir la conformité du gaz exporté. Ces dernières seront envoyées au service XP.

### 1.3 Processus de stockage des données des capteurs

Le schéma dans la figure ci-dessous 1.5 représente le processus de stockage des données des capteurs dans un système de supervision industrielle en trois étapes.

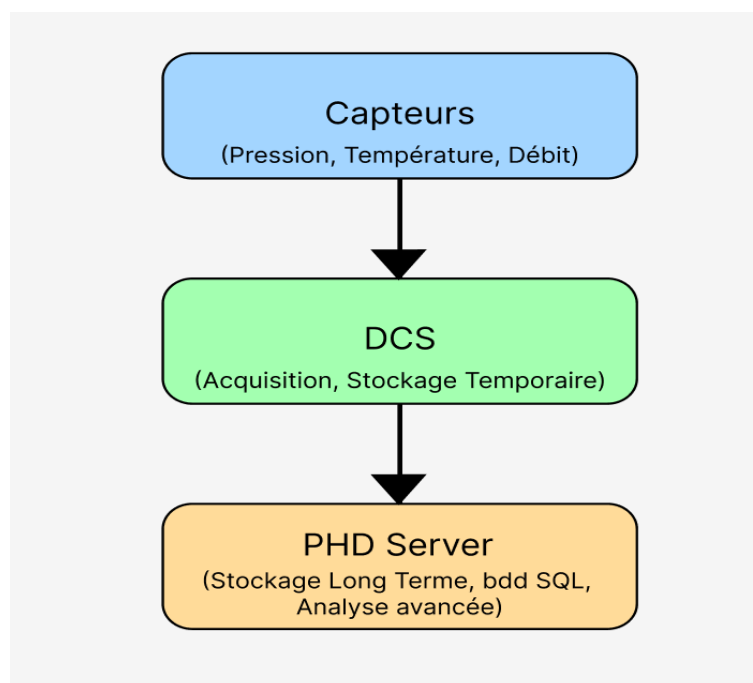


FIGURE 1.5 – processus de stockage de données capturées

1. H<sub>2</sub>S désigne le sulfure d'hydrogène, un gaz toxique, corrosif et potentiellement mortel présent en faibles quantités dans certains flux d'hydrocarbures.

### 1.3.1 Capteurs

Ce sont des capteurs installés au niveau des puits de gaz permettent de surveiller en temps réel les paramètres critiques de production de gaz (température, pression, débit..). Les données analogiques capturées au niveau de ces capteurs sont celles reçues au DCS (Distributed Control System), où elles sont traitées et utilisées pour le suivi et l'optimisation des opérations. Voici la figure 1.6 illustre un exemple de capteur installé sur le flow line venant de puit de gaz.



FIGURE 1.6 – exemple d'un capteur de débit de gaz

Les données analogiques venant des capteurs sont transmises au PLC<sup>2</sup>, qui les traitent et prennent des décisions automatisées, comme l'ajustement des vannes ou le déclenchement d'alarmes en cas d'anomalies. Pour assurer une communication fluide et efficace entre les différents équipements, un switch industriel est utilisé à proximité du puit. Il permet de centraliser et d'acheminer les données entre les PLC, le DCS (Distributed Control System). Grâce à cette infrastructure, les opérations des puits sont optimisées, sécurisées et supervisées à distance en temps réel.

---

2. Un contrôleur logique programmable (PLC) est un dispositif de calcul utilisé dans les environnements d'automatisation et de fabrication. Il contrôle différents processus électromécaniques basés sur des entrées et des règles spécifiques de traitement. Le fonctionnement de base d'un PLC passe par : Détecter l'état des dispositifs d'entrée connectés au PLC. Ensuite, Exécuter l'instruction du programme en fonction de l'entrée. Enfin, Faites fonctionner tous les périphériques de sortie connectés au PLC. Par exemple : alarmes, voyants lumineux, vannes et/ou affichages visuels [32]

Comme illustré sur la figure 1.7, l'automate programmable industriel (PLC) est installé directement sur le site du puits. Il permet la collecte des données de terrain et leur transmission vers le système de supervision central.



FIGURE 1.7 – Automate programmable industriel (PLC) installé sur site, à proximité d'un puits

## Système de Contrôle Distribué (DCS)

Un *système de contrôle distribué* (DCS) est une plateforme dédiée au contrôle et à l'exploitation automatisés d'une usine ou d'un procédé industriel. Il regroupe dans un seul environnement automatisé plusieurs composants essentiels : l'interface homme-machine (IHM), les solveurs logiques, l'historien, une base de données commune, la gestion des alarmes, ainsi qu'une suite d'ingénierie intégrée [9].

En pratique, le DCS est installé dans la salle de contrôle et permet de superviser et de réguler les opérations industrielles en temps réel. Il assure le contrôle automatique des équipements et des procédés, la supervision de l'état de l'installation, la collecte et le traitement des données issues des capteurs répartis sur le site, ainsi que l'automatisation d'actions selon des règles définies. Les signaux analogiques mesurés sont convertis en données numériques, puis affichés de manière visuelle sur des interfaces homme-machine (IHM), facilitant ainsi la prise de décision par les opérateurs. Toutefois, bien que le DCS fournisse une visualisation instantanée des paramètres du procédé, il dispose d'une capacité de stockage limitée. Pour répondre aux besoins d'archivage à long terme et permettre

des analyses approfondies, des solutions complémentaires telles que le *Process History Database* (PHD) sont intégrées au système global.

### 1.3.2 Process History Database (PHD)

Le Process History Database (PHD) de Honeywell<sup>3</sup> est une base de données conçue pour enregistrer en temps réel les données d'un site industriel, les stocker sur le long terme, et permettre une consultation intelligente de ces données en fonction des besoins opérationnels. Ce système repose sur une solide expérience de Honeywell dans l'intégration des systèmes de contrôle en temps réel avec les systèmes d'information des usines. Grâce à sa flexibilité et à ses capacités de personnalisation, le PHD s'adapte à divers environnements industriels, tout en offrant des outils puissants pour la gestion des données de processus.

Parmi ses principales fonctionnalités [8], on retrouve :

- La définition de variables (*tags*) avec des attributs hérités de classes.
- La collecte de données en temps réel depuis plusieurs ordinateurs connectés en réseau.
- Des options avancées pour le traitement et la compression des données à l'entrée.
- Un système d'archivage multi-fichiers et multi-niveaux pour un stockage long terme.
- Des outils intelligents de recherche et de manipulation des données adaptés aux systèmes d'information.
- Une communication fluide entre clients et serveurs, même sur plusieurs machines.
- Des calculs virtuels basés sur les valeurs des *tags*.
- Une gestion et un contrôle centralisés à partir d'un programme d'administration dédié.

## 1.4 Architecture réseau de communication entre les systèmes industriels et le serveur PHD

Après le stockage des données provenant des capteurs, ces informations sont exploitées à travers une architecture client-serveur.

La figure 1.8 présente l'architecture réseau utilisée au sein de GRN pour assurer la communication entre les systèmes industriels (comme le DCS) et les applications métiers.

---

3. Honeywell est une entreprise américaine spécialisée dans les technologies industrielles, l'automatisation, et les logiciels industriels, notamment les systèmes de supervision et de contrôle de procédés. Pour plus d'informations : <https://www.honeywell.com>.

Cette architecture repose sur le concept de **DMZ**<sup>4</sup>, une zone de sécurité intermédiaire qui permet de filtrer et contrôler les échanges entre le réseau industriel (niveau 02) et les applications métiers (niveau 01).

Le **pare-feu**<sup>5</sup> agit comme un gardien entre ces deux niveaux. Il autorise uniquement les flux de données nécessaires, comme :

- les flux **HTTPS**<sup>6</sup> depuis les applications métiers (AVOCET, SAP, etc.),
- les requêtes **SQL**<sup>7</sup> pour interroger ou récupérer les données stockées dans le serveur PHD.

Dans la **DMZ**, on trouve plusieurs serveurs essentiels :

- **Le serveur PHD** : il stocke les données collectées en temps réel depuis le terrain.
- **L'E-Server** : un serveur intermédiaire utilisé pour les applications.
- **Le serveur Antivirus** : pour protéger les serveurs contre les menaces connues.

Ce dispositif garantit une séparation stricte entre les systèmes critiques (comme le DCS) et les systèmes accessibles aux utilisateurs métier, assurant à la fois **sécurité, fiabilité et disponibilité** des données industrielles.

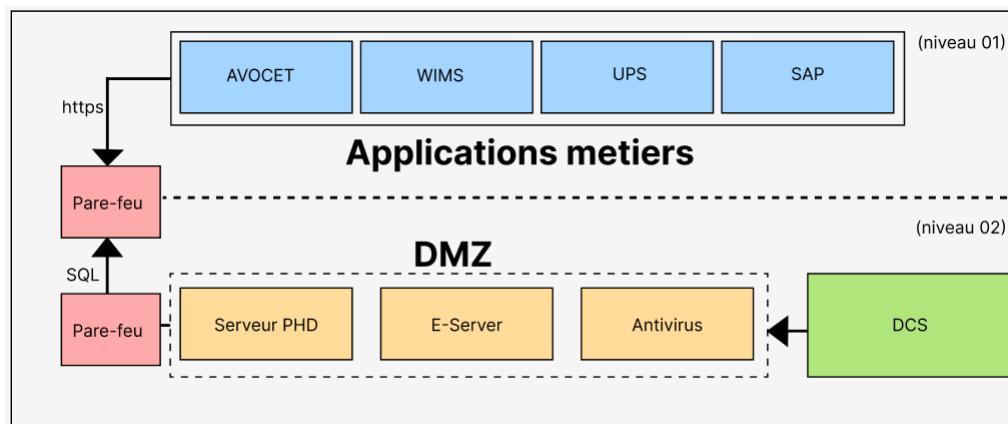


FIGURE 1.8 – Architecture réseau avec zone DMZ entre le DCS et les applications métiers

4. DMZ : Demilitarized Zone. Il s'agit d'une zone réseau intermédiaire placée entre deux pare-feux. Elle permet d'exposer certains serveurs (comme le PHD) à des utilisateurs internes ou externes tout en protégeant le cœur du réseau. Voir [3].

5. Pare-feu : Un pare-feu est un dispositif de sécurité réseau qui filtre les connexions entrantes et sortantes selon des règles prédéfinies.

6. HTTPS : HyperText Transfer Protocol Secure, est une version sécurisée du protocole HTTP. Il permet d'assurer la confidentialité et l'intégrité des données échangées sur le web.

7. SQL : est un langage permettant d'interroger, insérer et modifier les données dans une base de données relationnelle.

## 1.5 Etude des solutions existantes

### 1.5.1 WIMS (WellMaster Integrity Management System)

Le *WellMaster Integrity Management System* (WIMS) est une solution logicielle développée par la société **Peloton**, intégrée dans la plateforme WellView, et conçue pour assurer une gestion rigoureuse de l'intégrité des puits tout au long de leur cycle de vie. WIMS permet aux opérateurs pétroliers et gaziers de surveiller, documenter et analyser les paramètres critiques liés à l'état des puits.

L'outil offre une interface conviviale qui donne un accès structuré aux données techniques des équipements de puits, tout en facilitant la visualisation des barrières de sécurité. Cette fonctionnalité est essentielle pour garantir le confinement des fluides, prévenir les défaillances, et soutenir la prise de décision. WIMS contribue ainsi à améliorer la sécurité, l'efficacité et la conformité opérationnelle [23].

L'intégrité du puit est évaluée par la comparaison entre la pression mesurée sur les annulaires fixé sur le puits et le **MAOP** (*Maximum Allowable Operating Pressure*) qui est la pression maximale à laquelle le puit peut fonctionner en toute sécurité. La valeur % obtenue détermine l'intégrité de puit qui varie entre ces couleurs comme est illustré dans la figure 1.10.

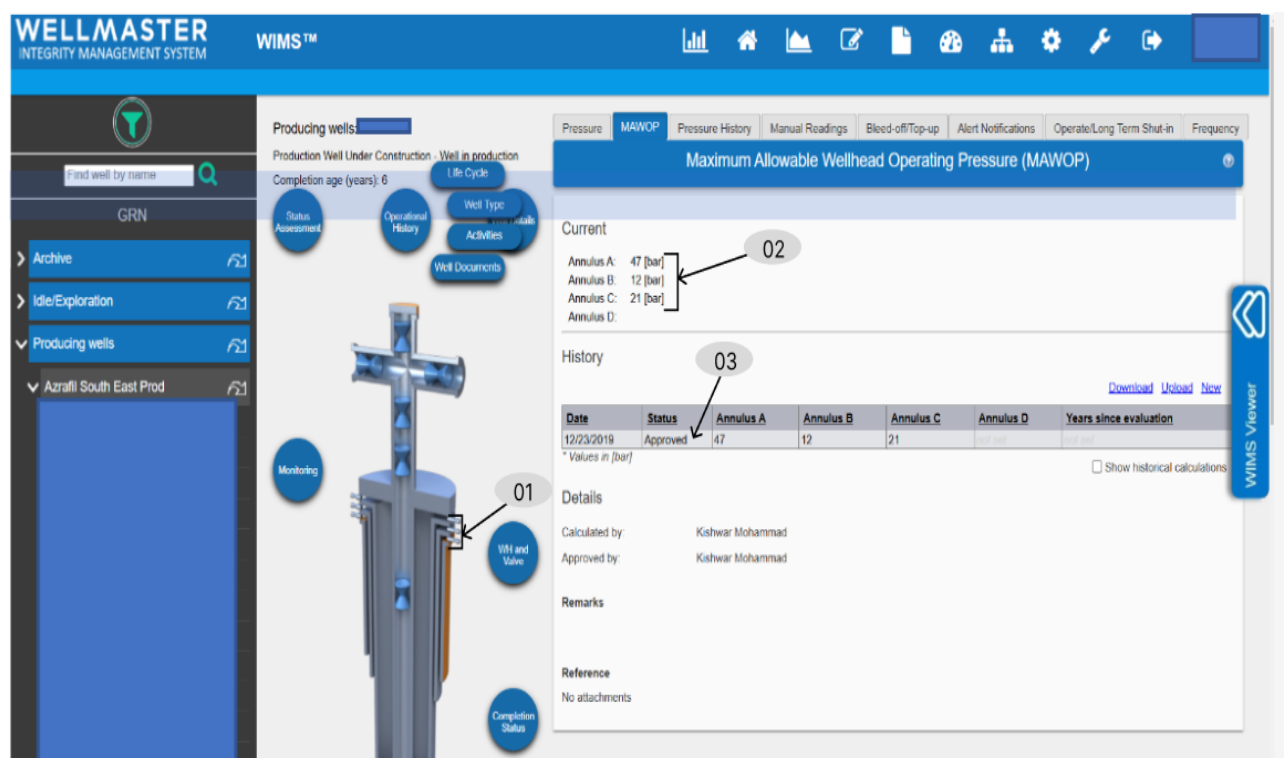


FIGURE 1.9 – Illustration des annulaires installés, des pressions mesurées et du statut du puits.

La Figure 1.9 est composée de trois parties :

- **Partie 1** : Représente les annulaires installés.
- **Partie 2** : Affiche les valeurs de pression mesurées sur les trois annulaires (A, B, C) le 23/12/2019. Ces valeurs sont comparées avec le MAOP.
- **Partie 3** : Présente le statut du puits, qui a été défini comme **Approved** en fonction de l'analyse des pressions mesurées.

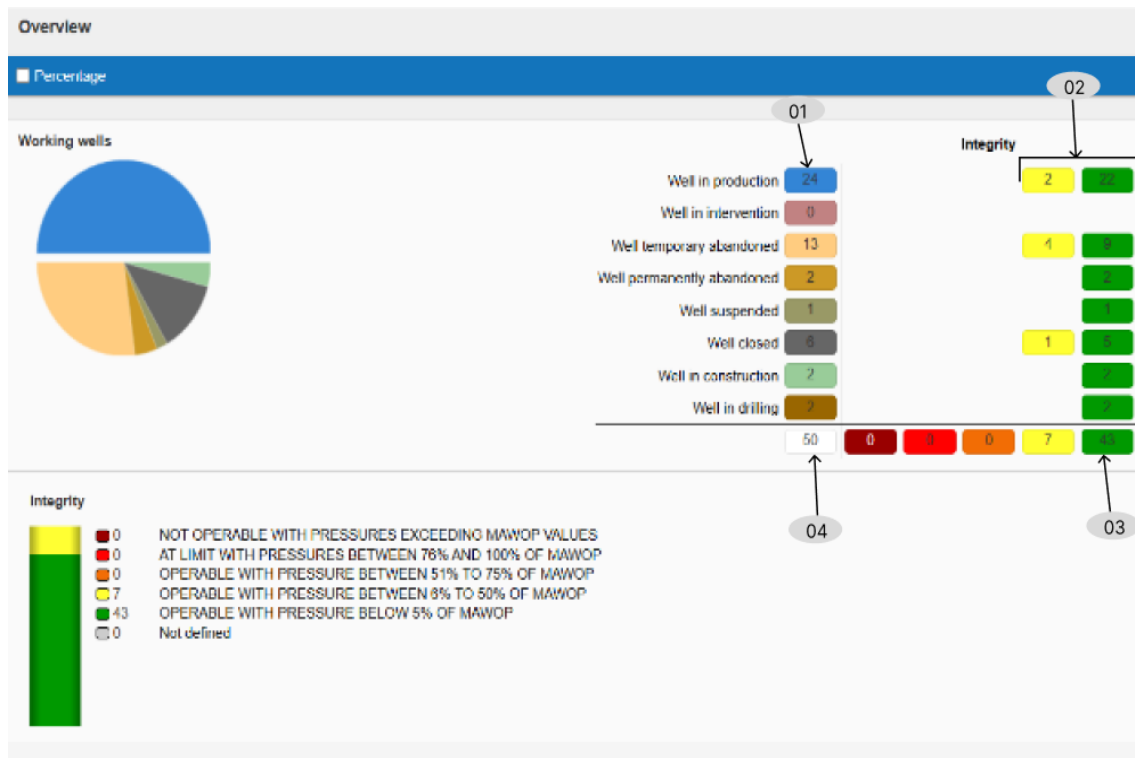


FIGURE 1.10 – État des puits et leur intégrité

La Figure 1.10 présente d'un côté les différents types de puits (en production, en intervention, temporairement abandonnés, définitivement abandonnés, suspendus, en forage, en construction, fermé) et de l'autre, leur intégrité :

- **Partie 1** : Nombre de puits en production, qui est de **24**.
- **Partie 2** : Intégrité des puits en production : **2 puits en jaune** et **22 en vert**.
- **Partie 3** : Nombre total de puits en état **vert**, soit **43**.
- **Partie 4** : Nombre total de puits, sans distinction de leur type.

### 1.5.2 Synergi Life

Synergi Life est un logiciel de gestion utilisé pour la déclaration, le suivi et l'analyse des incidents.

L'objectif est de signaler, enquêter et assurer le suivi des cas afin que la direction dispose d'informations immédiates pouvant être converties en tableaux de bord et en

statistiques. Cela permet d'optimiser l'utilisation des ressources de manière efficace et efficiente afin de réduire les risques et de prévenir les pertes dans l'ensemble du domaine HSE (Hygiène, Sécurité et Environnement). Utilisé par le service HSE du GRN.

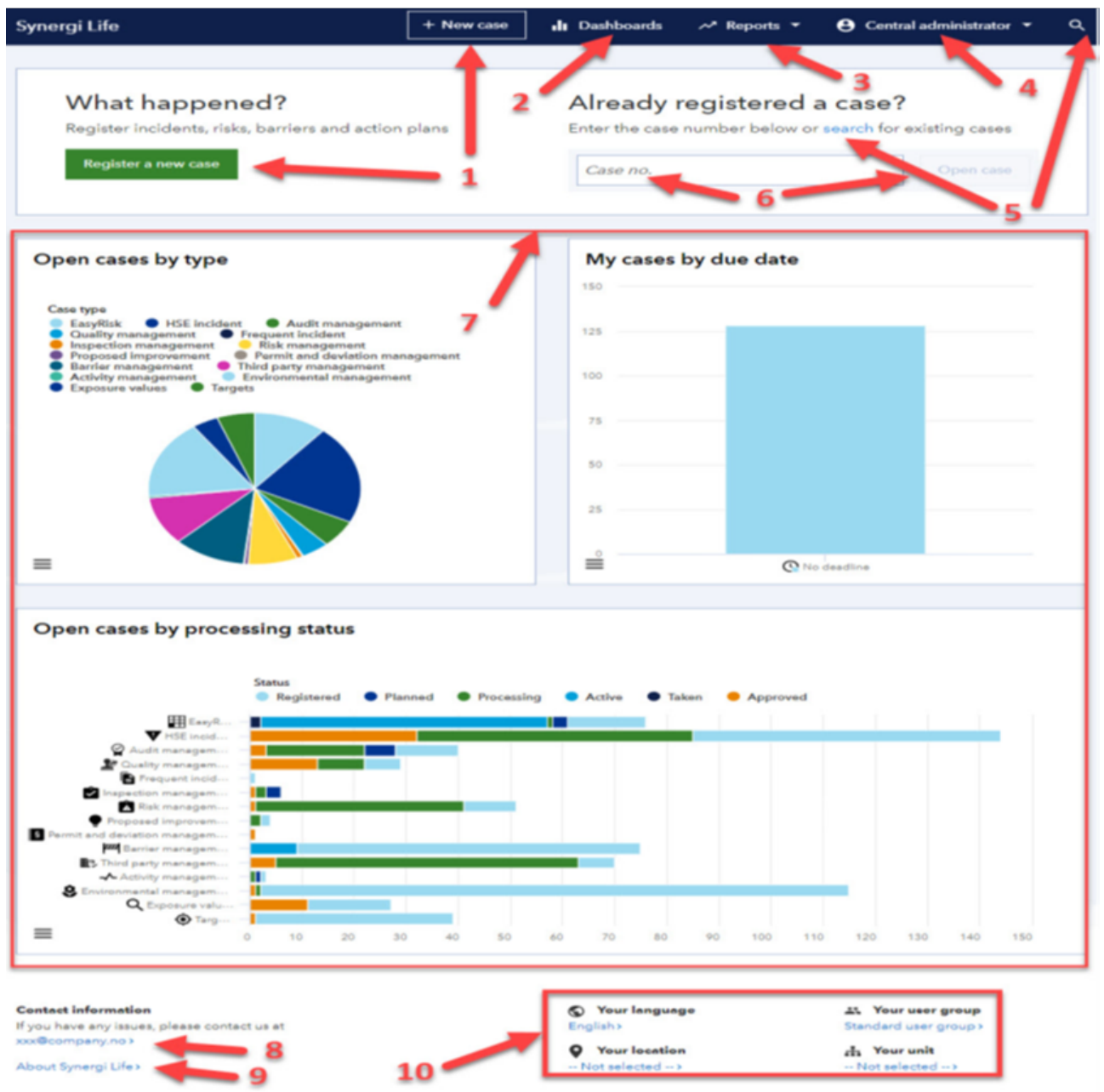


FIGURE 1.11 – Interface utilisateur d'accueil de Synergi

### Fonctionnalités disponibles depuis la page d'accueil

1. Enregistrement de nouveaux cas
2. Tableaux de bord - Voir chapitre séparé
3. Rapports
4. Menu de la page d'accueil

5. Recherche - Voir chapitre séparé
6. Ouvrir un cas en saisissant son numéro de dossier
7. Tableau de bord avec recherches - Voir chapitre séparé
8. Informations de contact
9. À propos de Synergi Life - Numéro de version
10. Paramètres utilisateur - Voir chapitre séparé

### 1.5.3 SAP MM (Materials Management)

SAP MM est un module du système SAP dédié à la gestion des matériaux et des ressources. Au sein du département XP du GRN, il est utilisé pour :

- \* Gérer les approvisionnements et les commandes de matériel.
- \* Suivre les stocks et les mouvements de matériaux.
- \* Planifier les besoins en matériaux en fonction des projets et des opérations en cours.
- \* Assurer une intégration fluide avec les autres modules SAP, tels que la gestion des finances et la maintenance.

### 1.5.4 UPS (Uniformance Process Studio)

*Uniformance Process Studio* est un outil logiciel développé par **Honeywell**<sup>8</sup> permettant aux ingénieurs de visualiser, explorer et manipuler les données historisées dans le système PHD. Il sert d'interface de configuration, d'analyse et d'accès aux données de production en temps réel ou archivées, afin de mieux surveiller les performances des processus industriels et détecter les anomalies [10].

#### Points forts

- \* **Accès aux données du PHD** : Permet aux ingénieurs d'accéder aux données de production pour analyser les performances des processus.
- \* **Analyse des performances** : Aide à identifier les problèmes potentiels et à améliorer l'exploitation des ressources.
- \* **Intégration avec l'environnement industriel** : Compatible avec les systèmes Honeywell, facilitant la collecte et l'exploitation des données.

---

8. Honeywell est une entreprise américaine spécialisée dans les technologies industrielles, l'automatisation, et les logiciels industriels, notamment les systèmes de supervision et de contrôle de procédés. Pour plus d'informations : <https://www.honeywell.com>.

## Points faibles

- \* **Absence d'automatisation des KPI**<sup>9</sup> : Les indicateurs de performance doivent être calculés manuellement, ce qui est long et répétitif.
- \* **Processus manuel fastidieux** : Les ingénieurs doivent extraire, organiser et analyser les données sous Excel, augmentant le risque d'erreurs.

### 1.5.5 AVOCET

*AVOCET Production Operations Software* est une plateforme développée par **Schlumberger** (désormais SLB)<sup>10</sup>, conçue pour surveiller, gérer et optimiser la production pétrolière et gazière. Elle centralise les données de production, permet la planification des opérations, le suivi des puits, et offre des outils de reporting. Au sein de GRN, elle est utilisée principalement par les départements EP et XP pour la gestion quotidienne des puits et l'analyse de performance [27].

## Points forts

- \* **Centralisation et analyse des données** : AVOCET permet de collecter et d'analyser en temps réel et de manière centralisée les données de production de gaz.
- \* **Accès aux données en temps réel** : Mise à jour des données toutes les 10 minutes, facilitant la prise de décision rapide.
- \* **Optimisation de l'exploitation** : Utilisé par les départements EP et XP pour améliorer la gestion et l'optimisation des champs gaziers.
- \* **Intégration avec PHD et DCS** : Permet une communication avec d'autres systèmes de gestion des données industrielles.

## Points faibles

- \* **Visualisation graphique limitée et illisible** : L'ingénieur ne peut pas ajuster l'intervalle de temps selon ses besoins pour l'affichage des données. De plus, les graphiques générés sont difficiles à interpréter, obligeant les ingénieurs à extraire manuellement les données pour les traiter via Excel, ce qui augmente le risque d'erreurs.

---

9. KPI (Key Performance Indicator) désigne un indicateur clé de performance utilisé pour mesurer l'efficacité d'un processus ou d'une activité. Réf. : Bernard Marr, *Key Performance Indicators : The 75+ Measures Every Manager Needs to Know*, Pearson Education, 2015.

10. SLB (anciennement Schlumberger) est une entreprise multinationale spécialisée dans les technologies de services pétroliers, notamment les logiciels, la gestion des puits, la production et l'exploration. Pour plus d'informations : <https://www.slb.com>.

- \* **Extraction de données limitée** : Difficulté à extraire des données sur une longue période (limitée à un mois et demi), et obligation d’extraire l’ensemble des données sans pouvoir cibler certains paramètres uniquement.
- \* **Accès limité en mode Read/Write** : Le système impose un nombre maximum d’utilisateurs pouvant accéder simultanément en mode **Read & Write**, générant une surcharge organisationnelle entre les ingénieurs.
- \* **Absence de séparation claire entre les données** : Les données de production et d’exploitation sont mélangées, ce qui complique la recherche d’informations spécifiques par les ingénieurs, notamment ceux du service Mesure.
- \* **Interface surchargée et peu intuitive** : L’interface utilisateur est dense, mal organisée, et présente une surcharge d’informations qui rend la navigation difficile.

## 1.6 Problématique

L’analyse des systèmes actuellement utilisés au sein du Groupement Reggane Nord, notamment AVOCET et Uniformance Process Studio (UPS), a mis en évidence plusieurs limites entravant l’efficacité des ingénieurs dans le suivi et l’optimisation de la production. Parmi les faiblesses relevées figurent : l’absence d’automatisation des indicateurs de performance (KPI), des processus manuels fastidieux d’extraction et d’analyse des données sous Excel, une interface utilisateur surchargée et peu intuitive, ainsi que des restrictions d’accès en mode collaboratif. De plus, l’absence de séparation claire entre les données d’exploitation et de production rend la navigation complexe pour les différents services. Ces contraintes opérationnelles entraînent une perte de temps, une surcharge de travail et un risque accru d’erreurs humaines. Il devient alors indispensable de réfléchir à une solution plus adaptée, qui réponde aux besoins spécifiques des équipes techniques du CPF.

## 1.7 Objectifs

L’objectif principal de ce travail est de concevoir une solution logicielle qui surmonte les limites identifiées dans les systèmes existants. Plus précisément, il s’agit de :

- \* Automatiser le calcul et l’affichage des indicateurs de performance (KPI) à partir des données du PHD ;
- \* Permettre une visualisation claire, dynamique et filtrable des données de production ;
- \* Offrir une interface utilisateur ergonomique, centrée sur les besoins des ingénieurs des services XP, EP, Mesure et Intégrité ;
- \* Réduire la dépendance aux extractions manuelles et aux traitements sous Excel ;

- \* Faciliter la prise de décision rapide grâce à des visualisations intelligentes (tableaux et graphiques) ;
- \* Assurer une gestion sécurisée des accès selon les rôles des utilisateurs.

## 1.8 Solution envisagée

Face aux limites identifiées dans les systèmes AVOCET et UPS, notamment en matière de visualisation, d'extraction et d'analyse des données de production, une solution applicative sur mesure est envisagée. Elle vise à offrir une plateforme moderne, automatisée et ergonomique, répondant aux besoins réels des services Mesure et Exploitation du GRN.

### 1. Optimisation de l'extraction des données

L'extraction manuelle et restreinte des données dans AVOCET, notamment la limite à un mois et demi, constitue un frein majeur pour les ingénieurs. La solution envisagée permet de :

- \* Accéder à l'historique des données sur plusieurs années, sans restriction ;
- \* Filtrer dynamiquement les données selon une plage de dates personnalisée ou selon des paramètres précis ;
- \* Exporter les données automatiquement dans des formats adaptés (CSV, Excel, PDF), réduisant les manipulations répétitives.

### 2. Automatisation du calcul des KPI

Pour pallier l'absence d'automatisation dans UPS, un module de calcul automatique des *Key Performance Indicators* (KPI) est intégré. Il permettra de :

- \* Générer en temps réel les indicateurs de performance à partir des données issues du PHD ;
- \* Visualiser les KPI dans un tableau de bord interactif, filtrable par période et par puits ;
- \* Exporter les rapports de performance de manière standardisée.

### 3. Gestion intelligente des droits d'accès

La limitation du nombre de comptes **Read/Write** dans AVOCET complexifie la collaboration. La nouvelle solution intègre :

- \* Un système de gestion des accès basé sur les rôles, permettant une attribution flexible et temporaire des permissions ;

- \* Une interface de visualisation en temps réel des connexions actives en mode **Read/Write** ;
- \* Une file d'attente intelligente pour prioriser les demandes d'accès selon l'urgence ou la fonction.

#### 4. Interface utilisateur améliorée

Enfin, pour dépasser les problèmes d'ergonomie et de surcharge d'interface rencontrés dans AVOCET, la solution proposera :

- \* Une interface épurée et responsive, adaptée à différents types d'utilisateurs (ingénieurs, techniciens, superviseurs) ;
- \* Une organisation hiérarchique claire des informations, avec navigation intuitive ;
- \* Une personnalisation des vues selon les profils utilisateurs.

## 1.9 Méthodologie de conception

### 1.9.1 Langage de Modélisation Unifié (UML)

Un **diagramme UML** fournit une représentation visuelle d'un aspect d'un système. Les diagrammes UML aident les architectes et les développeurs à comprendre une application, à collaborer autour de sa conception et à la faire évoluer [14]. UML permet ainsi de décrire aussi bien les aspects **statiques** que **dynamiques** d'un système à travers différents types de diagrammes.

Dans le cadre de notre projet, nous présentons principalement quatre types de diagrammes UML :

- **Diagramme de cas d'utilisation** : en langage UML, ces diagrammes modélisent le comportement global d'un système et permettent de capturer ses exigences. Ils décrivent les fonctions générales et la portée du système, tout en identifiant les interactions entre le système et ses acteurs. Les cas d'utilisation et les acteurs précisent ce que le système fait et comment il est utilisé, sans toutefois détailler son fonctionnement interne [12].
- **Diagramme de séquence** : il s'agit d'un diagramme d'interaction UML représentant les échanges de messages entre objets, dans le cadre d'un scénario particulier du système.
- **Diagramme de classes** : ces diagrammes constituent les plans du système ou d'un sous-système. Ils permettent de modéliser les objets qui le composent, d'illustrer les relations entre ces objets et de décrire leurs responsabilités ainsi que les services qu'ils offrent [13].

## 1.9.2 Processus Unifié (UP)

Le processus unifié (UP) est une méthode de développement logiciel qui vise à transformer les besoins des utilisateurs en un système logiciel. Cette approche repose sur l'utilisation de composants logiciels reliés les uns aux autres par des interfaces définies clairement. L'utilisation de la méthode UP implique la création de plans d'élaboration et de construction du système logiciel en utilisant le langage UML. Bien que UML soit intégré dans le processus unifié, ce dernier se distingue véritablement par trois expressions clés :

- Une orientation vers les cas d'utilisation
- Une attention particulière portée à l'architecture
- Une approche itérative et incrémentale.

Ces aspects font la singularité de la méthode UP.

### Les phases du cycle de développement UP

Le cycle de développement UP organise les tâches et les itérations en quatre phases qui sont les suivantes :

1. **Création (Incubation) :** La première étape du processus unifié consiste à définir clairement la portée du système, en distinguant ce qui sera inclus à l'intérieur du système et ce qui restera en dehors. Cette phase implique également l'identification des acteurs impliqués, la clarification des besoins et exigences, ainsi que la conception d'une architecture de base fonctionnelle. Il est essentiel d'identifier les risques majeurs susceptibles de perturber le bon déroulement du projet.
2. **Élaboration :** La deuxième étape du processus consiste à élaborer le système en se basant sur la compréhension initiale de ses fonctionnalités et des risques critiques. L'objectif principal de cette phase est de stabiliser l'architecture du système en affinant le modèle de cas d'utilisation initial, en capturant de nouveaux besoins éventuels, et en analysant et concevant la majorité des cas d'utilisation formulés. Dans l'idéal, cette étape peut par ailleurs inclure l'implémentation et les tests des cas d'utilisation initiaux.
3. **Construction :** Au cours de cette étape, il est essentiel de recueillir tous les besoins qui n'ont pas encore été identifiés, car il sera difficile de le faire lors de la prochaine phase. Ensuite, il est nécessaire de poursuivre l'analyse, la conception et en particulier la mise en œuvre de tous les cas d'utilisation. En fin de compte, les développeurs doivent fournir une version du système qui peut être exécutée.
4. **Transition :** La dernière étape du processus UP a pour objectif de vérifier si le système répond aux besoins des utilisateurs, de détecter d'éventuelles défaillances, de compléter la documentation logicielle et d'adapter le produit à l'environnement (installation et mise en place).

La figure 1.12 qui suit est la représentation des phases du cycle de vie d'UP

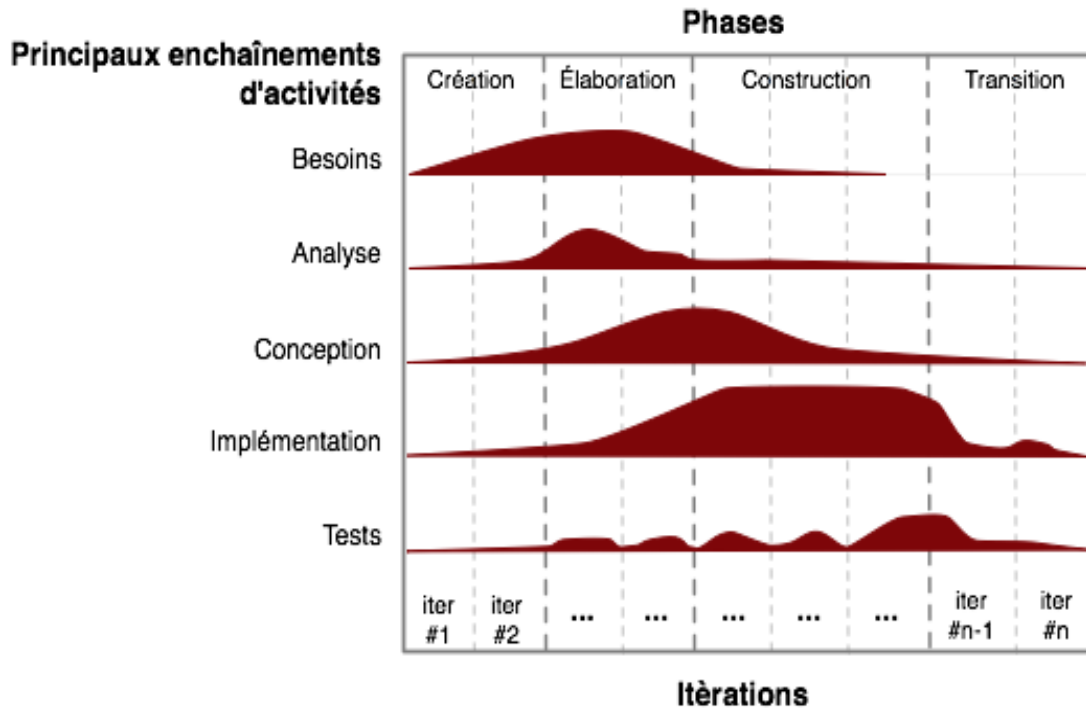


FIGURE 1.12 – Les Phases du cycle de vie du processus unifié

## 1.10 Pertinence d'UML et de l'UP dans le projet

Le langage **UML** et le processus unifié (**UP**) ont été choisis pour la modélisation et la conduite de notre projet de gestion de la production pétrolière et gazière pour plusieurs raisons essentielles.

Premièrement, **UML (Unified Modeling Language)** est un langage standard largement utilisé pour la modélisation de systèmes orientés objet. Il permet de représenter de manière visuelle les entités du système (telles que les puits, les sprints de production, les utilisateurs, etc.), leurs relations, ainsi que les processus métier comme la validation de données, l'interprétation des analyses ou la gestion des interventions. Cette représentation graphique facilite non seulement la compréhension du système, mais également la communication entre les différentes parties prenantes du projet (développeurs, encadrants, utilisateurs métier).

Deuxièmement, le **processus unifié (UP)** est une méthode de développement itérative et incrémentale, centrée sur les cas d'utilisation et l'architecture du système. Cette approche nous a permis de structurer notre travail selon des phases claires (inception, élaboration, construction, transition), tout en apportant progressivement les fonctionnalités clés de l'application, comme la gestion des puits, l'affichage des productions, l'analyse graphique, ou encore l'export des rapports. Elle a également facilité l'adaptation aux retours

et aux évolutions des besoins métier.

Enfin, l'utilisation combinée d'UML et de UP a permis une **conception rigoureuse** de notre système, tout en assurant une documentation claire et durable. Cela facilitera la maintenance future et l'extension du logiciel, tout en rendant la structure du projet compréhensible pour d'autres développeurs ou intervenants, même en l'absence des membres initiaux de l'équipe.

## 1.11 Illustration par des maquettes fonctionnelles

Dans le but de concrétiser la solution envisagée et de fournir une vision préliminaire de l'interface utilisateur, deux maquettes ont été conçues. Celles-ci visent à illustrer la manière dont l'ingénieur du service EP pourrait interagir avec l'application proposée. La première maquette, illustrée par la **figure 1.13**, représente une interface intuitive affichant les informations générales d'un puits, incluant sa localisation, ses caractéristiques techniques, ainsi que les données relatives à la phase de forage. Cette visualisation a été pensée pour améliorer la lisibilité des données et leur structuration, tout en assurant une ergonomie supérieure à celle des outils actuels tels qu'AVOCET.

La seconde maquette, présentée dans la **figure 1.14**, propose un aperçu de la page dédiée au suivi de l'intégrité des puits. L'objectif est de faciliter la navigation entre les différentes périodes de maintenance. Les éléments visuels, tels que les icônes, les couleurs et la hiérarchisation de l'information, ont été conçus pour offrir une meilleure clarté et un accès rapide à l'essentiel.

Ces maquettes jouent un rôle essentiel dans la phase de conception, en permettant de valider les choix fonctionnels avant l'implémentation, tout en recueillant des retours de la part des utilisateurs finaux.

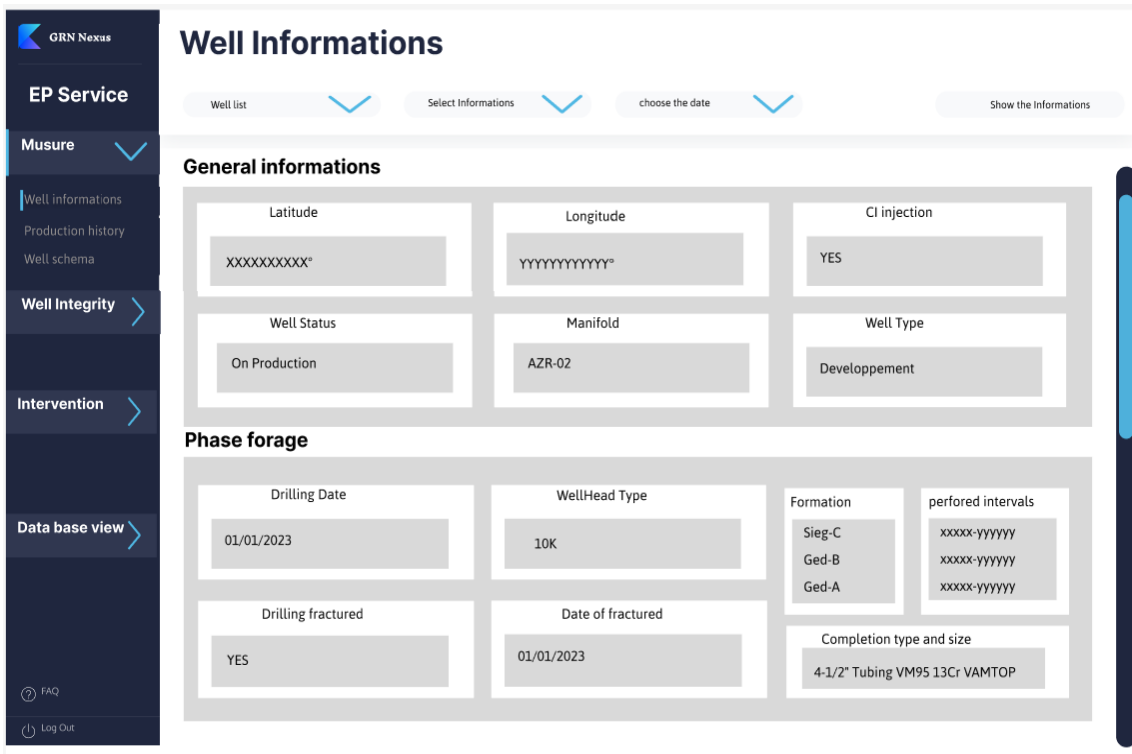


FIGURE 1.13 – Interface utilisateur qui affiche les informations generale de puit et celle de la phase forage

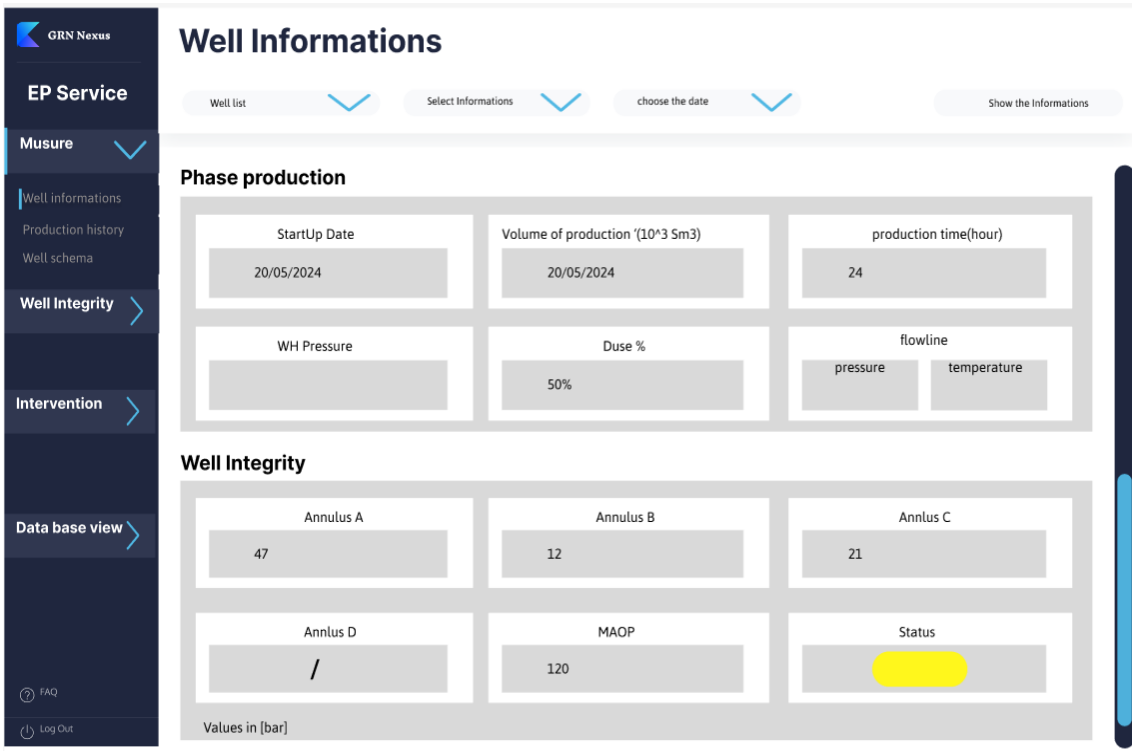


FIGURE 1.14 – Interface utilisateur qui affiche les informations d'integrité de puit et celle de la phase production

## 1.12 Conclusion

Ce premier chapitre a permis de poser les fondations de notre travail en s'appuyant sur l'étude de terrain menée au sein du Groupement Reggane Nord (GRN) durant la période de stage. À travers une immersion au sein des différents services techniques du CPF, une compréhension détaillée des infrastructures, des processus de collecte et de stockage des données, ainsi qu'une analyse approfondie des outils actuellement en usage (AVOCET, UPS, WIMS, etc.), nous avons pu identifier un ensemble de limitations opérationnelles affectant l'efficacité des ingénieurs.

Ces lacunes – notamment l'absence d'automatisation des indicateurs de performance, la surcharge d'informations dans les interfaces, les restrictions d'accès collaboratif et la difficulté d'extraction des données – ont mis en lumière un besoin réel d'évolution des outils informatiques au sein du GRN.

C'est dans cette perspective qu'a été formulée une problématique claire et structurée, suivie de la définition d'objectifs concrets et d'une solution envisagée sous forme d'un système plus ergonomique, intelligent et adapté aux besoins des utilisateurs finaux. Des maquettes fonctionnelles ont également été proposées afin de visualiser les premiers contours de cette future solution.

Le chapitre suivant sera consacré à la spécification détaillée des besoins fonctionnels et techniques, ainsi qu'à la conception du système en s'appuyant sur des outils de modélisation comme UML.

# Chapitre 2

## Phase de Conception

# Introduction

Le processus de conception de notre plateforme, que nous avons nommée « **GasScope** », s'est appuyé sur une approche méthodique visant à modéliser de manière précise les besoins métier et les interactions entre les différents utilisateurs et composants du système. Ce chapitre présente les différentes étapes réalisées dans le cadre de la conception fonctionnelle et technique du système.

Les principales étapes suivies sont les suivantes :

1. **Spécification des besoins fonctionnels et non fonctionnels** : Cette étape permet de décrire ce que le système doit faire (fonctionnalités attendues) ainsi que les contraintes de performance, de sécurité, d'ergonomie ou de maintenabilité à respecter.
2. **Identification des acteurs et des cas d'utilisation** : Cette étape a permis de recenser les différents profils utilisateurs (administrateur, ingénieur XP, service EP...) ainsi que leurs interactions avec la système.
3. **Création des diagrammes de cas d'utilisation** : Ces diagrammes ont servi à visualiser les différents scénarios d'utilisation du système, selon les rôles attribués à chaque acteur.
4. **Élaboration des diagrammes de séquence** : Ces représentations ont permis de modéliser les échanges entre les utilisateurs et les composants du système, étape essentielle pour comprendre la logique des traitements.
5. **Conception du diagramme de classes et du modèle relationnel** : Ces modèles ont constitué la base de la structure de la base de données et ont servi de référence pour l'implémentation technique.

Cette démarche, soutenue par l'utilisation du langage UML et du processus unifié (UP), a permis d'assurer une conception claire, cohérente et évolutive de notre application.

## 2.1 Spécification des besoins

### 2.1.1 Besoins fonctionnelles

Les besoins fonctionnels définissent les caractéristiques et les comportements du système qui sont nécessaires pour répondre aux exigences des utilisateurs et pour remplir les fonctions attendues.

Ci-dessous le tableau [2.1](#) des besoins fonctionnels que nous avons recueillis à GRN.

<b>Acteur</b>	<b>Besoins</b>	<b>Fonctionnalités attendues</b>
Administrateur / IT	Gérer les utilisateurs et leurs accès	Création, modification et suppression de comptes. Attribution des rôles (administrateur, ingénieur, technicien, etc.). Gestion sécurisée des privilèges.
	Sécurité et organisation des données	Mise en place d'une architecture sécurisée. Restriction d'accès selon profil. Suivi des connexions et audit des activités.
Département EP	Accès rapide aux informations générales d'un puits	Interface centralisée affichant les données statiques du puits : nom, statut, type, localisation, formations, etc.
	Suivi de la production à une date donnée	Consultation des données de production (gaz, eau, condensat...) pour n'importe quelle date, sans limitation d'historique.
	Suivi de l'intégrité d'un puits	Accès à l'état d'intégrité (anomalies, pressions, vannes...) à une date donnée.
	Suivi et gestion des interventions	Interface listant les interventions passées avec détails : date, nature, statut, utilisateur ayant effectué l'action.
	Visualisation graphique claire	Génération de graphiques sur les paramètres (production, pressions, ouverture de vanne...) à des fins d'analyse et de présentation.
	Export des graphes et données	Exportation des graphiques (PDF, image) et données (Excel) pour intégration dans des rapports ou supports techniques.
	Interface intuitive	Navigation rapide, accès simplifié aux données utiles, ergonomie adaptée aux besoins métier.
Département XP	Gestion de la production générale du CPF	Enregistrement quotidien du gaz expédié, fuel gas, gaz torché, acide gaz, condensats, eau, avec suivi des prévisions.
	Suivi de la consommation électrique	Saisie et visualisation de la consommation au niveau du CPF et de la base de vie.
	Vérification du gaz de vente	Consultation des résultats d'analyse (CO, PCS, etc.) pour s'assurer de la conformité aux spécifications commerciales.

*Suite du tableau à la page suivante*

Acteur	Besoins	Fonctionnalités attendues
	Suivi des indicateurs (KPI)	Calcul automatique des taux de torchage, rendement, taux de réalisation des objectifs, production moyenne, etc.
	Visualisation graphique des performances	Courbes de production journalières, mensuelles, annuelles, avec possibilité de comparaison avec les objectifs fixés.
	Exportation de rapports et graphiques	Export des données et des graphes pour communication avec la direction ou lors des réunions techniques.
Service LABO	Centralisation des données d'analyse	Interface dédiée pour saisir les résultats d'analyses de gaz directement dans le système (gaz produit, gaz de vente).
	Notification au département XP	Envoi automatique d'une alerte ou notification dès qu'une analyse est validée pour consultation par le service XP.
	Traçabilité des analyses	Archivage sécurisé des analyses avec date, type de gaz, et auteur. Consultation de l'historique par période.
	Interface de saisie adaptée	Formulaire de saisie simple, structuré, avec contrôle des champs d'analyse (CO, PCS, humidité, etc.).

TABLE 2.1 – Tableau récapitulatif des besoins et fonctionnalités attendues par service

### 2.1.2 Besoins non fonctionnelles

Les besoins non fonctionnels définissent les critères de qualité et les contraintes qui doivent être respectées par le système pour répondre aux exigences des utilisateurs.

Les besoins non fonctionnels clés de notre plateforme sont les suivants :

- **Internationalisation** : La plateforme doit intégrer un mécanisme d'internationalisation permettant son utilisation en plusieurs langues, notamment le français et l'anglais. Cette caractéristique vise à assurer une accessibilité accrue pour des profils d'utilisateurs variés et à favoriser une adoption plus large du système dans un contexte organisationnel où la diversité linguistique peut constituer un facteur déterminant.
- **Interface utilisateur ergonomique** L'interface doit être intuitive, simple à prendre en main et adaptée aux besoins métier. Les formulaires, tableaux, graphiques et modules doivent être organisés de manière claire, avec un design cohérent pour faciliter

l'usage au quotidien.

- **Sécurité et contrôle d'accès** : Protection des données et gestion des droits d'accès des utilisateurs.
- **Documentation utilisateur intégrée** Une rubrique d'aide ou de documentation doit être accessible selon le rôle de l'utilisateur. Elle doit expliquer les principales fonctionnalités, les processus métier, et guider l'utilisateur dans l'utilisation de la plateforme.
- **Performance et rapidité** : Optimisation des requêtes pour garantir un affichage rapide des données.
- **Compatibilité** : Assurer le bon fonctionnement sur différents navigateurs et appareils.
- **Expérience utilisateur optimisée** : L'expérience utilisateur doit être fluide, agréable et adaptée aux contextes professionnels. L'affichage des données, les interactions (clics, filtres, exports) et les messages du système doivent être clairs et cohérents.

Dans la prochaine section, nous allons définir les acteurs qui seront amenés à interagir avec notre plateforme ainsi que les cas d'utilisation clés pour chacun de ces acteurs.

## 2.2 Acteurs et leur cas d'utilisation

### 2.2.1 Acteurs du système

Les acteurs de notre système sont les suivants :

<b>Acteur</b>	<b>Rôle</b>
<b>Tous les acteurs</b>	Accéder au système en s'authentifiant, se déconnecter.
<b>Chef de service ITC</b>	Gérer les comptes des systèmes existants au GRN, gérer les puits (ajouter, supprimer, modifier les puits).
<b>Service EP (Intégrité ou Mesure)</b>	Suivre les données de production des puits, analyser l'intégrité des puits, prendre des décisions en cas d'anomalie, transmettre les décisions au service intervention.
<b>Service XP</b>	Suivre le processus de production du CPF, analyser les résultats des analyses du labo et prendre des décisions pour l'action.
<b>Service Intervention</b>	Accéder à la partie commentaire pour décrire les interventions, consulter l'historique des interventions passées.
<b>Service Labo</b>	Effectuer des analyses quotidiennes, remplir un formulaire d'analyse, notifier le service XP lorsqu'une analyse est terminée.
<b>Directeur des opérations</b>	Accéder à toutes les interfaces des sous-utilisateurs pour gérer les données et prendre des décisions globales.

TABLE 2.2 – Tableau des acteurs et de leurs rôles

### 2.2.2 Cas d'utilisation

« Les cas d'utilisation, également connus sous le nom de UseCases en anglais, constituent une méthode de modélisation intégrée à l'UML. Leur fonction est de décrire les interactions qui ont lieu entre un système et ses acteurs, qu'ils soient des utilisateurs ou d'autres systèmes. » [14]

Nous avons identifié les besoins qui doivent être pris en compte. En conséquence, nous avons défini les cas d'utilisation que notre plateforme devra gérer

Acteur(s)	Cas d'utilisation principaux	Extensions / Options
<b>Package 1 : Gestion des utilisateurs</b>		
Administrateur (Chef ITC)	<ul style="list-style-type: none"> <li>— Créer / Ajouter un utilisateur</li> <li>— Modifier un utilisateur</li> <li>— Supprimer / Désactiver un utilisateur</li> <li>— Attribuer / Modifier un rôle</li> <li>— Gérer les droits d'accès</li> <li>— Consulter la liste des utilisateurs</li> </ul>	<ul style="list-style-type: none"> <li>— Filtrer</li> <li>— Rechercher</li> <li>— S'authentifier</li> </ul>
<b>Package 2 : Gestion des données liées aux puits</b>		
Service EP (acteur principal)	<ul style="list-style-type: none"> <li>— Gérer les interventions sur les puits</li> <li>— Gérer l'intégrité des puits</li> <li>— Gérer la production des puits</li> <li>— Gérer les informations générales d'un puits</li> <li>— Gérer les manifolds</li> <li>— Gérer les périmètres</li> </ul>	<ul style="list-style-type: none"> <li>— Rechercher</li> <li>— Filtrer</li> <li>— Exporter</li> <li>— Imprimer</li> <li>— S'authentifier</li> </ul>

*Suite à la page suivante*

### Suite du tableau des cas d'utilisation

Acteur(s)	Cas d'utilisation principaux	Extensions / Options
Services XP, LABO, Admin ITC (acteurs secondaires)	<ul style="list-style-type: none"> <li>— Consulter la liste des puits</li> <li>— Consulter les interventions</li> <li>— Consulter la production des puits</li> <li>— Consulter l'intégrité des puits</li> <li>— Consulter les informations générales d'un puits</li> </ul>	<ul style="list-style-type: none"> <li>— Filtrer</li> <li>— Exporter</li> <li>— Imprimer</li> </ul>
<b>Package 3 : Gestion de la production générale</b>		
Service XP (acteur principal)	<ul style="list-style-type: none"> <li>— Saisir la production journalière (gaz, condensat, eau)</li> <li>— Saisir les prévisions journalières (gaz, condensat, fuel, torché, acide gaz)</li> <li>— Saisir les paramètres opératoires (pression CPF, runtime, etc.)</li> <li>— Saisir la consommation électrique (CPF, base de vie)</li> <li>— Saisir les explications de manque à produire</li> </ul>	<ul style="list-style-type: none"> <li>— S'authentifier</li> </ul>

*Suite à la page suivante*

### Suite du tableau des cas d'utilisation

Acteur(s)	Cas d'utilisation principaux	Extensions / Options
Services XP, EP, LABO (consultation)	<ul style="list-style-type: none"> <li>— Consulter l'historique de la production journalière</li> <li>— Consulter l'historique des prévisions journalières</li> <li>— Consulter l'historique des paramètres opératoires</li> <li>— Consulter l'historique de la consommation électrique</li> <li>— Consulter l'historique des explications</li> <li>— Consulter les KPI</li> </ul>	<ul style="list-style-type: none"> <li>— Filtrer</li> <li>— Exporter</li> <li>— Imprimer</li> </ul>
<b>Package 4 : Gestion des données d'analyse laboratoire</b>		
Service LABO (acteur principal)	<ul style="list-style-type: none"> <li>— Gérer une analyse (ajouter, modifier, supprimer)</li> <li>— Gérer les spécifications gaz de vente (ajouter, modifier, supprimer)</li> <li>— Vérifier l'état du gaz de vente</li> </ul>	<ul style="list-style-type: none"> <li>— Filtrer</li> <li>— Exporter</li> <li>— Imprimer</li> <li>— S'authentifier</li> </ul>
Services XP, EP (consultation)	<ul style="list-style-type: none"> <li>— Consulter l'historique des analyses</li> </ul>	<ul style="list-style-type: none"> <li>— Filtrer</li> <li>— Exporter</li> <li>— Imprimer</li> </ul>

TABLE 2.3 – Tableau récapitulatif des cas d'utilisation par package

Maintenant que nous avons défini les différents acteurs et les cas d'utilisation clés pour chacun d'eux, nous allons séparer la modélisation en huit packages distincts, et illustrer chaque package avec les diagrammes UML adéquats.

## **2.3 Gestion des utilisateurs**

Dans ce premier package de notre projet, nous allons nous concentrer sur la gestion des utilisateurs de différents services de GRN.

### 2.3.1 Diagramme de cas d'utilisation

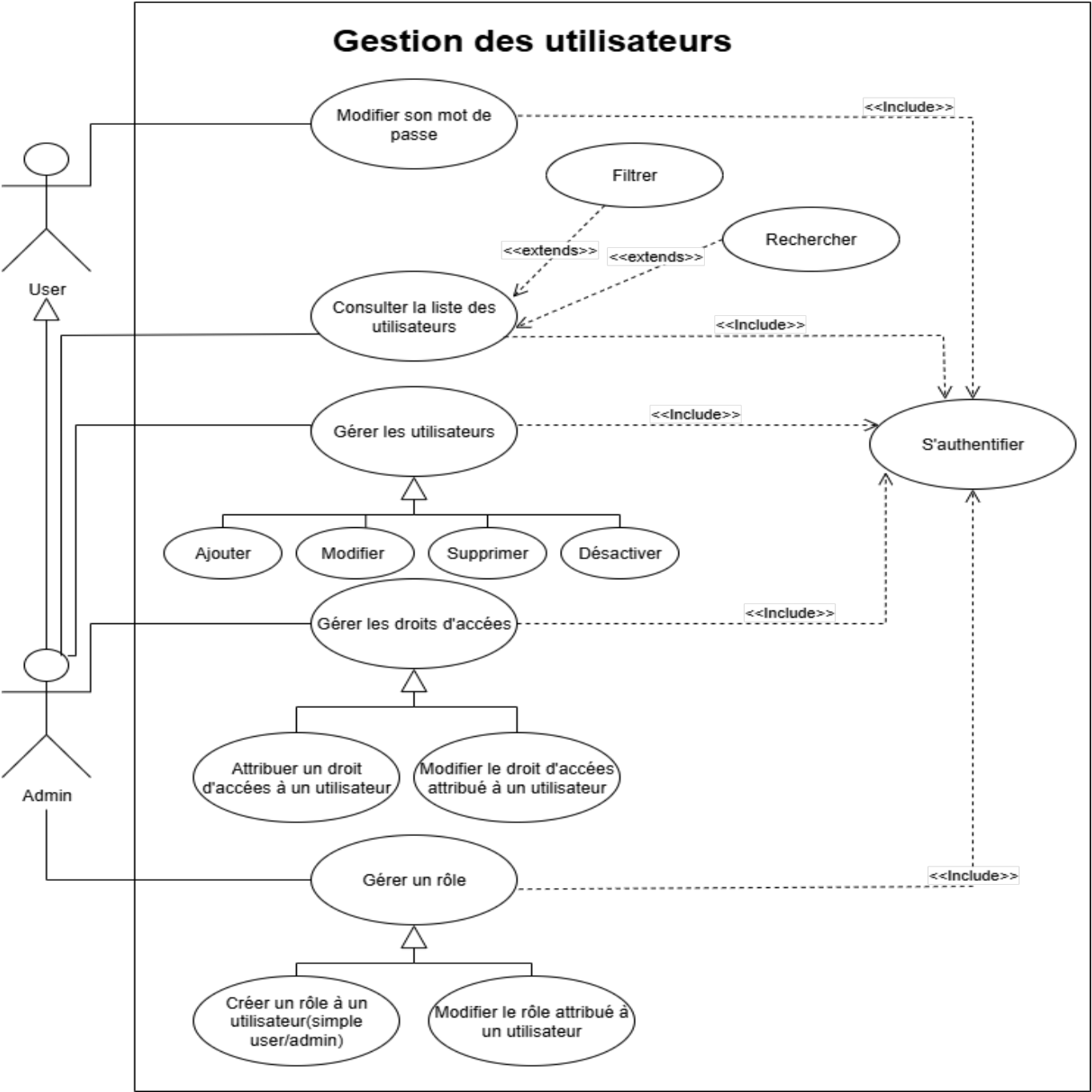


FIGURE 2.1 – Diagramme de cas d'utilisation "Gestion des utilisateurs"

### 2.3.2 Diagramme de séquence "Créer un utilisateur"

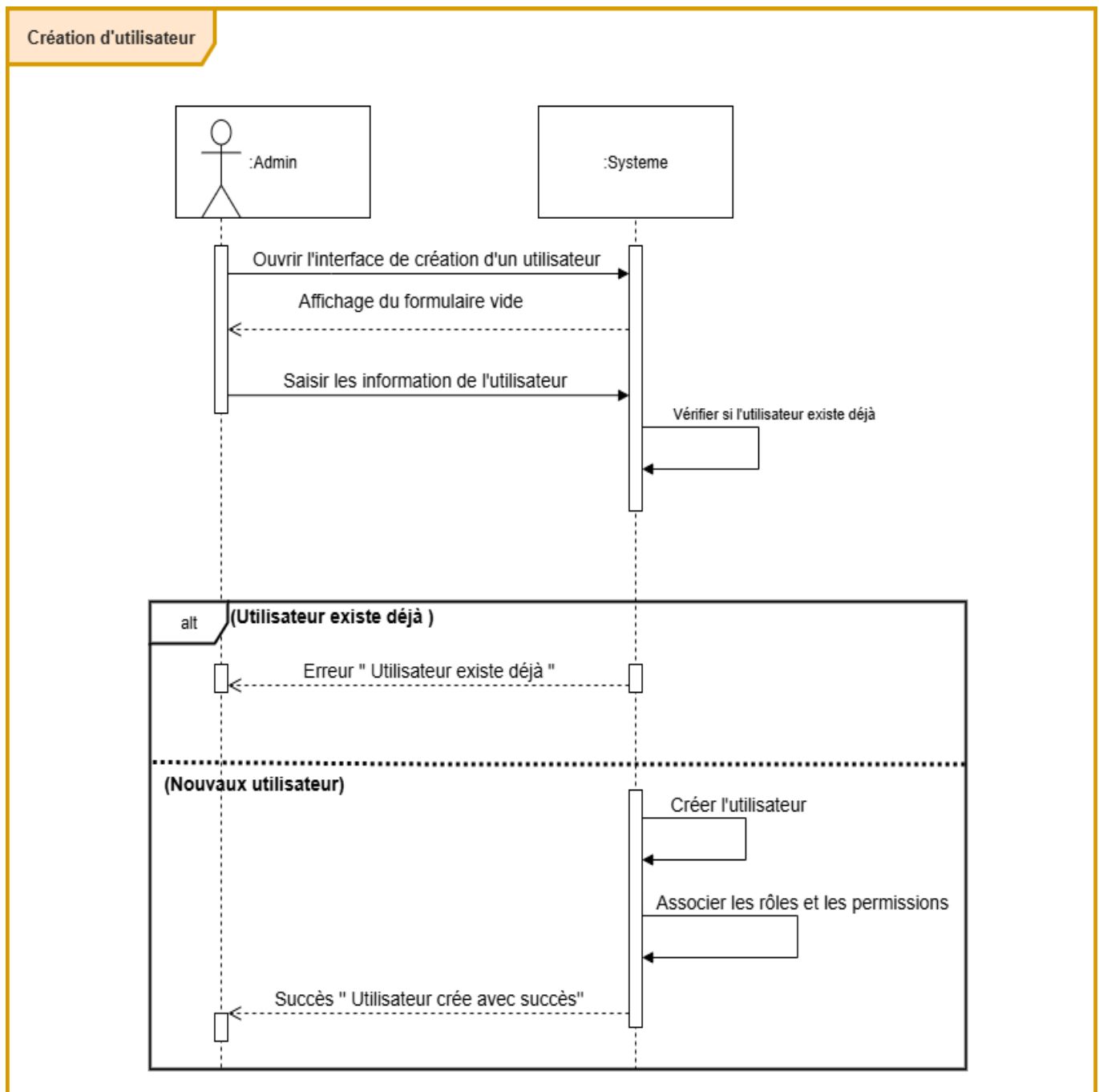


FIGURE 2.2 – Diagramme de séquence "créer un utilisateur"

### 2.3.3 Diagramme de classe

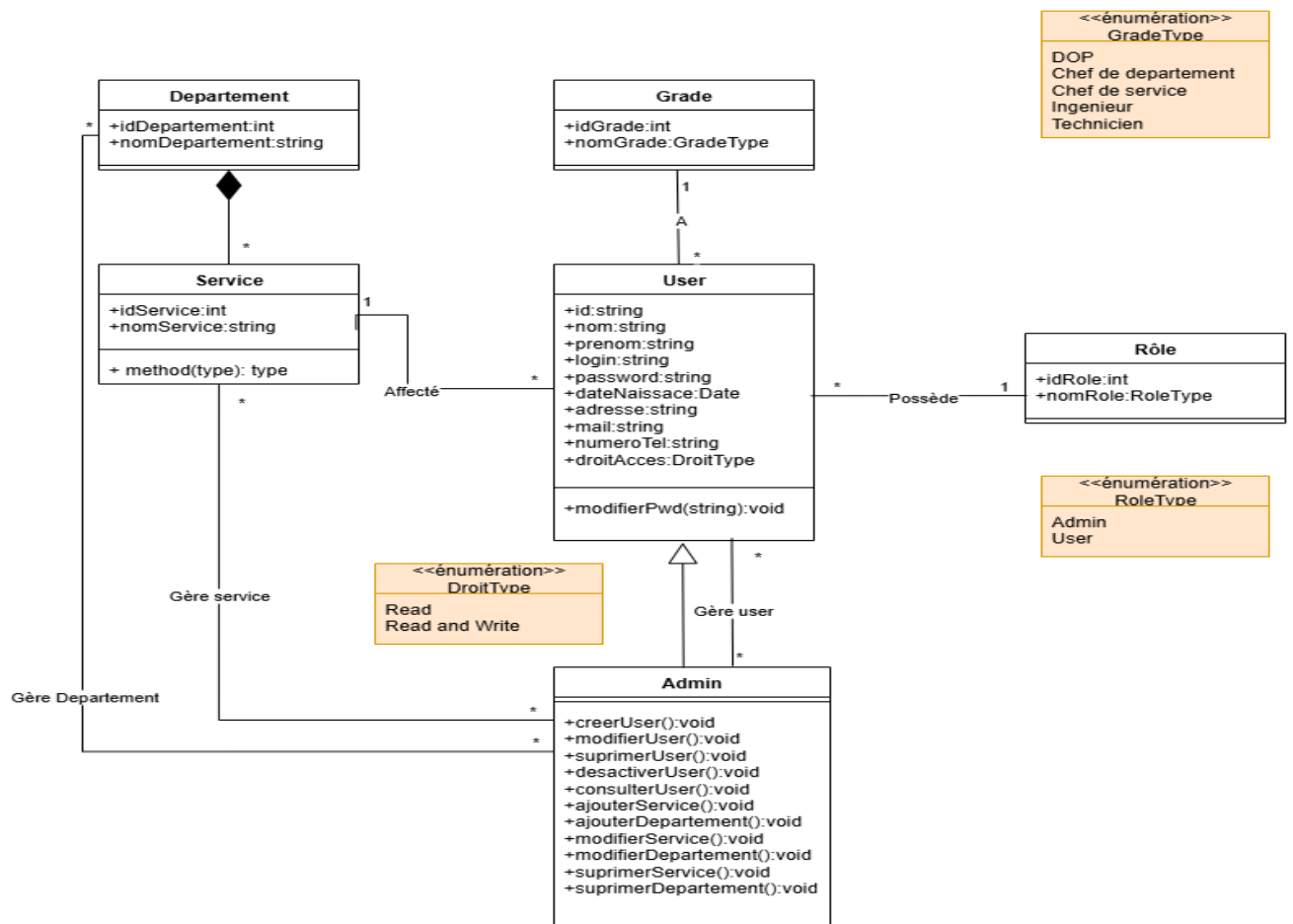


FIGURE 2.3 – Diagramme de classe "Gestion des utilisateurs"

## 2.4 Gestion des puits

### 2.4.1 Diagramme de cas D'utilisation

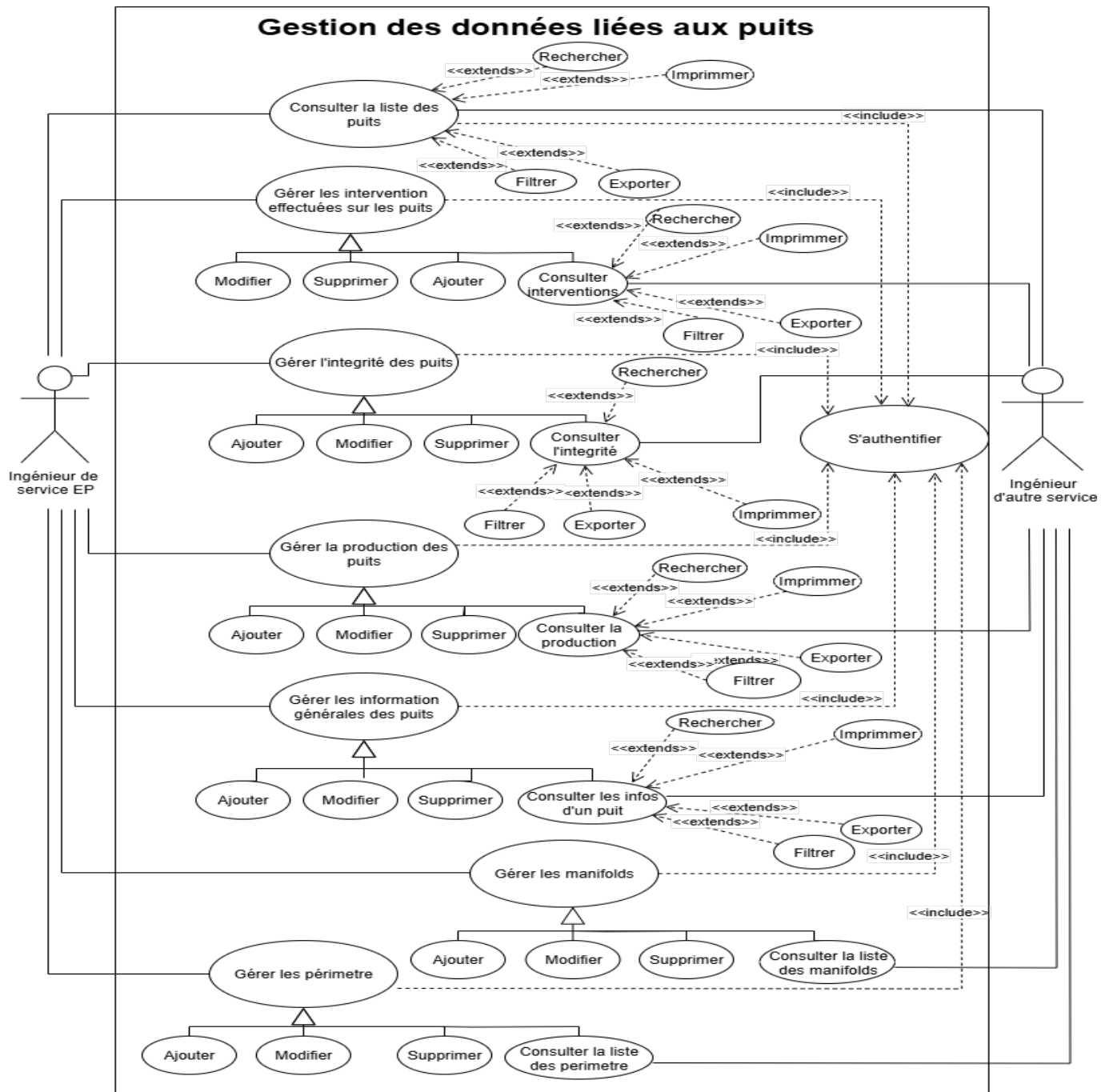


FIGURE 2.4 – Diagramme de cas d'utilisation "Gestion des puits"

### 2.4.2 Diagramme de Séquence "Rechercher un puit"

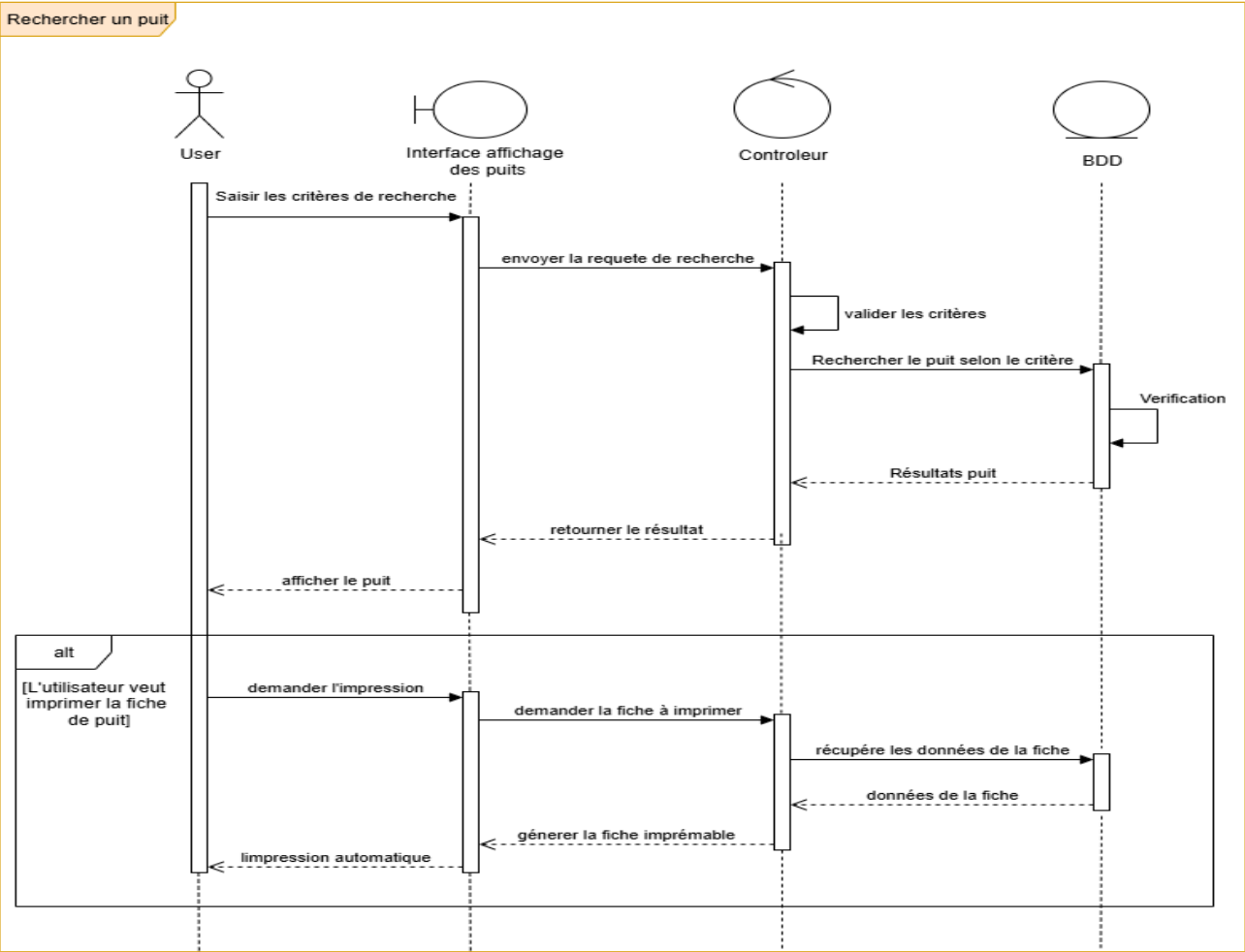


FIGURE 2.5 – Diagramme de Séquence "Rechercher un puit"

## 2.4.3 Diagramme de classe

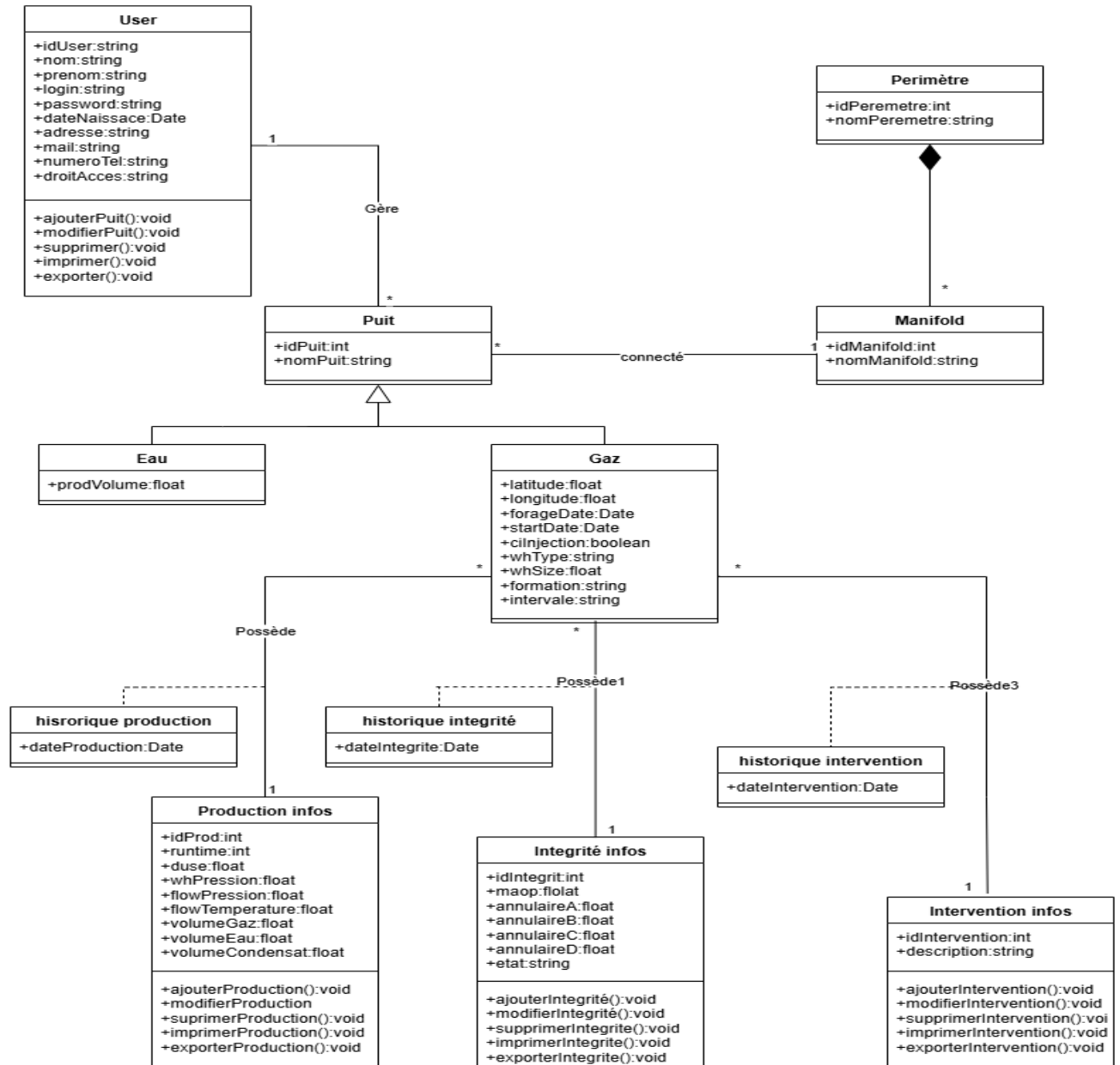


FIGURE 2.6 – Diagramme de classe "Gestion des puits"

## 2.5 Gestion de production

### 2.5.1 Diagramme de cas d'utilisation

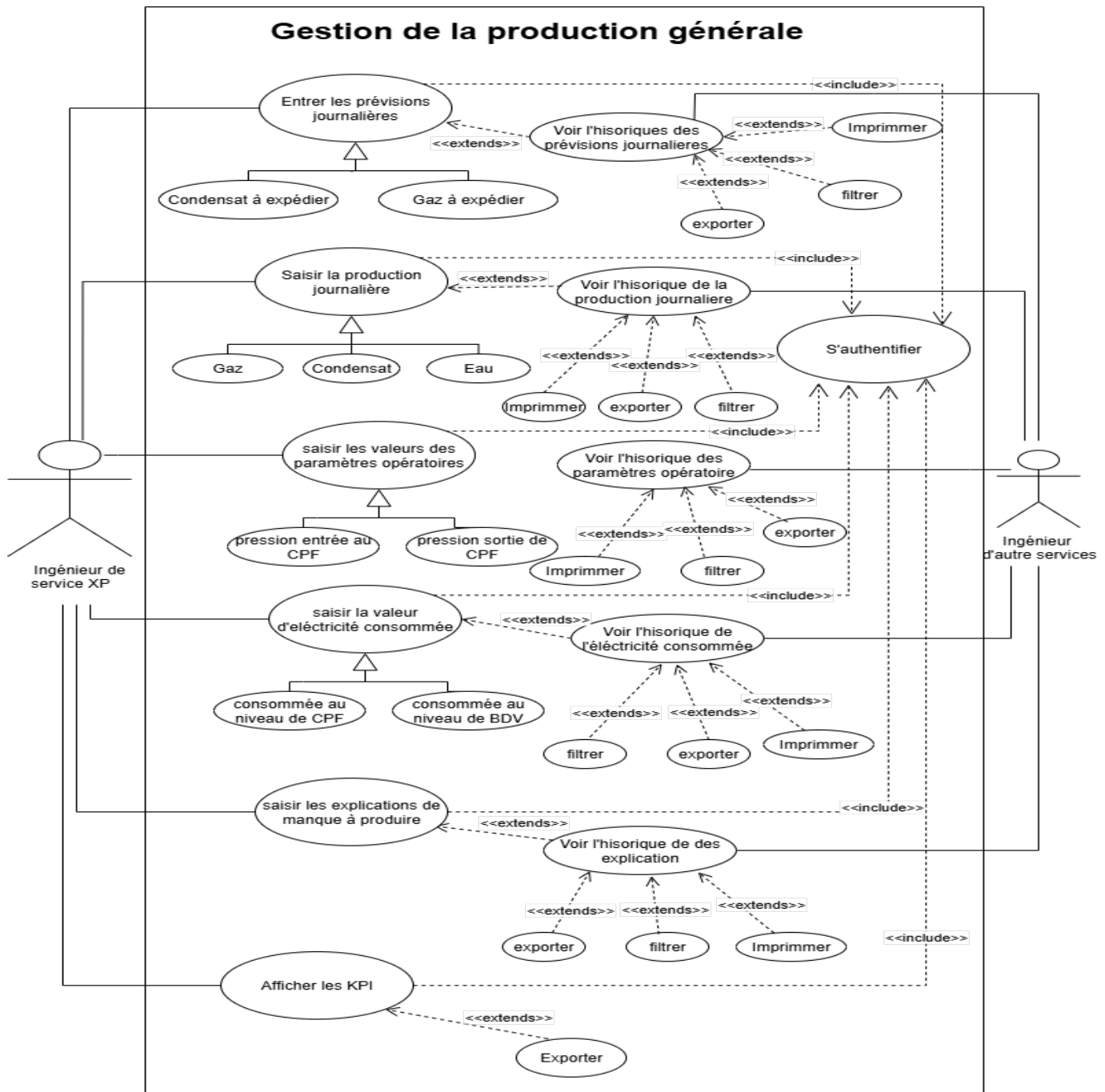


FIGURE 2.7 – Diagramme de cas d'utilisation "Gestion de production"

## 2.5.2 Diagramme de séquence "saisir une production"

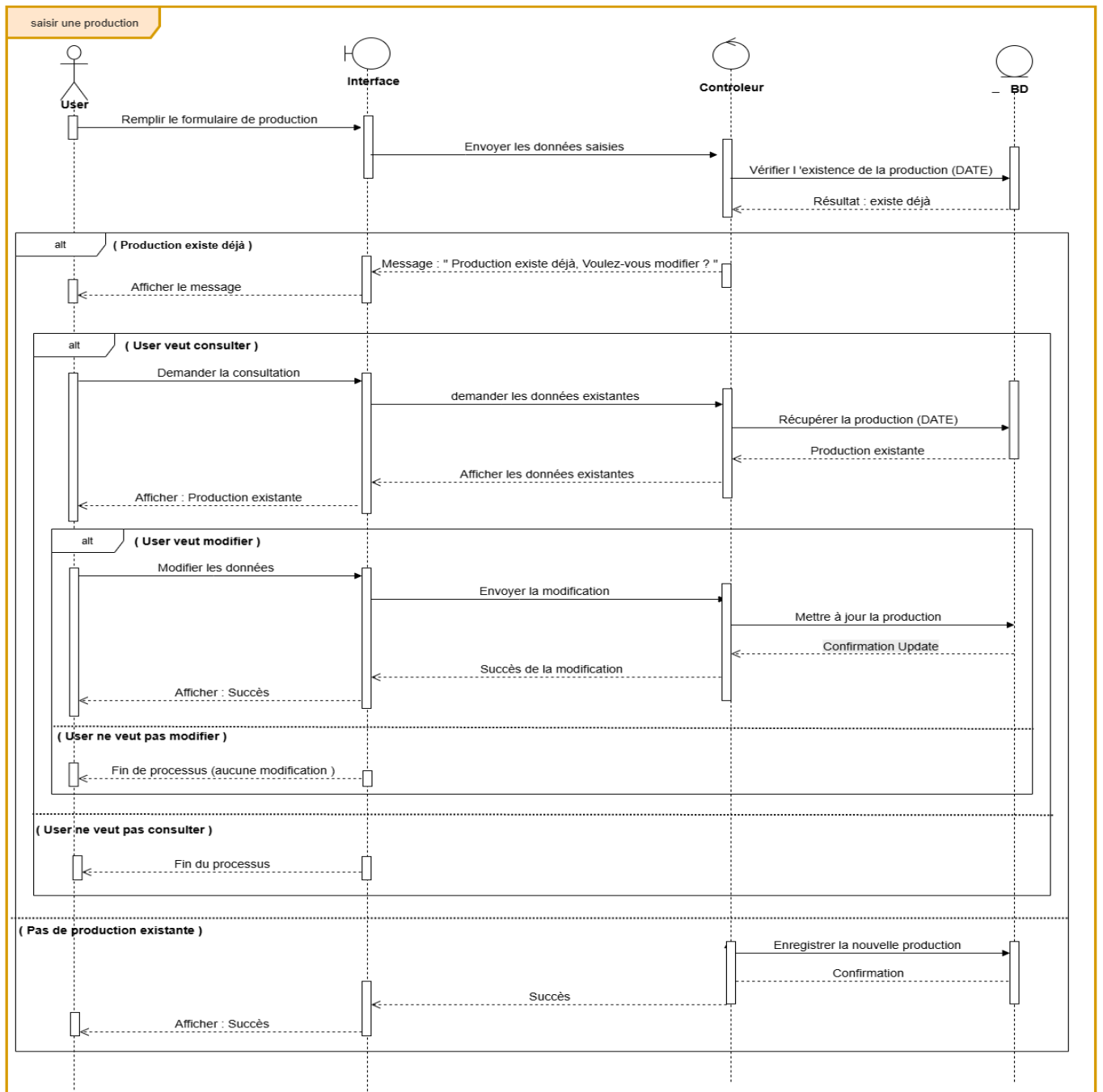


FIGURE 2.8 – Diagramme de séquence "Saisir une production"

### 2.5.3 Diagramme de classe

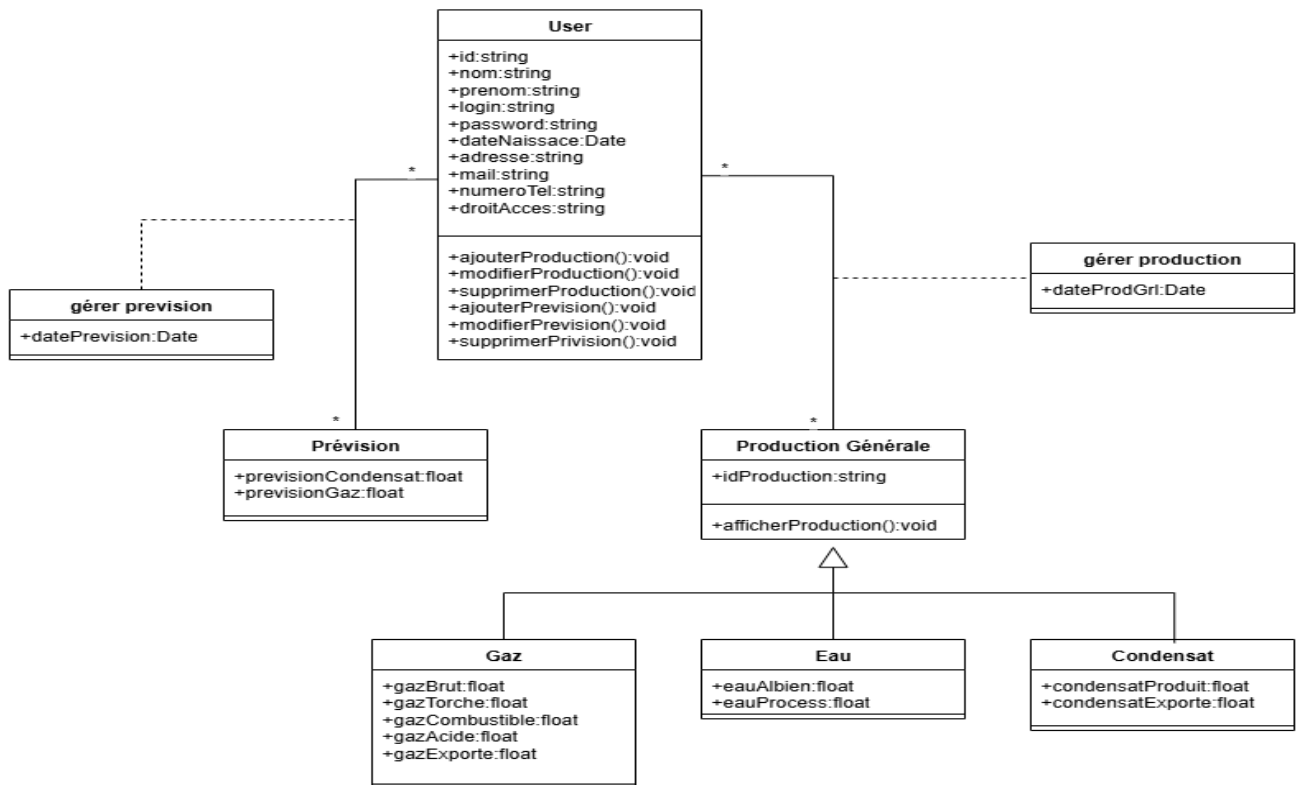


FIGURE 2.9 – Diagramme de classe "Gestion de production"

## 2.6 Gestion des analyses de gaz

### 2.6.1 Diagramme de cas d'utilisation

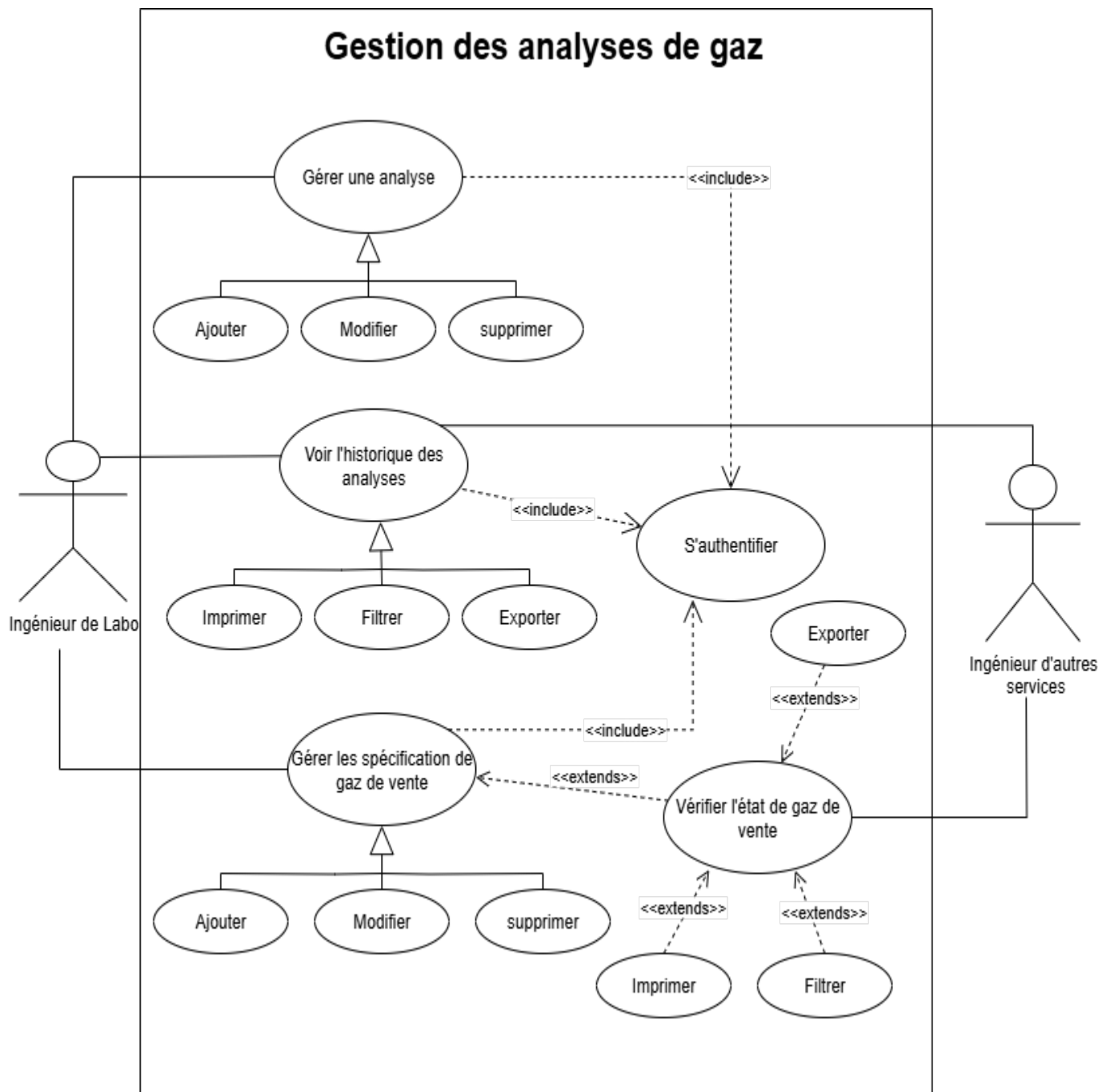


FIGURE 2.10 – Diagramme de cas d'utilisation "Gestion des analyses de gaz"

## 2.6.2 Diagramme de séquence

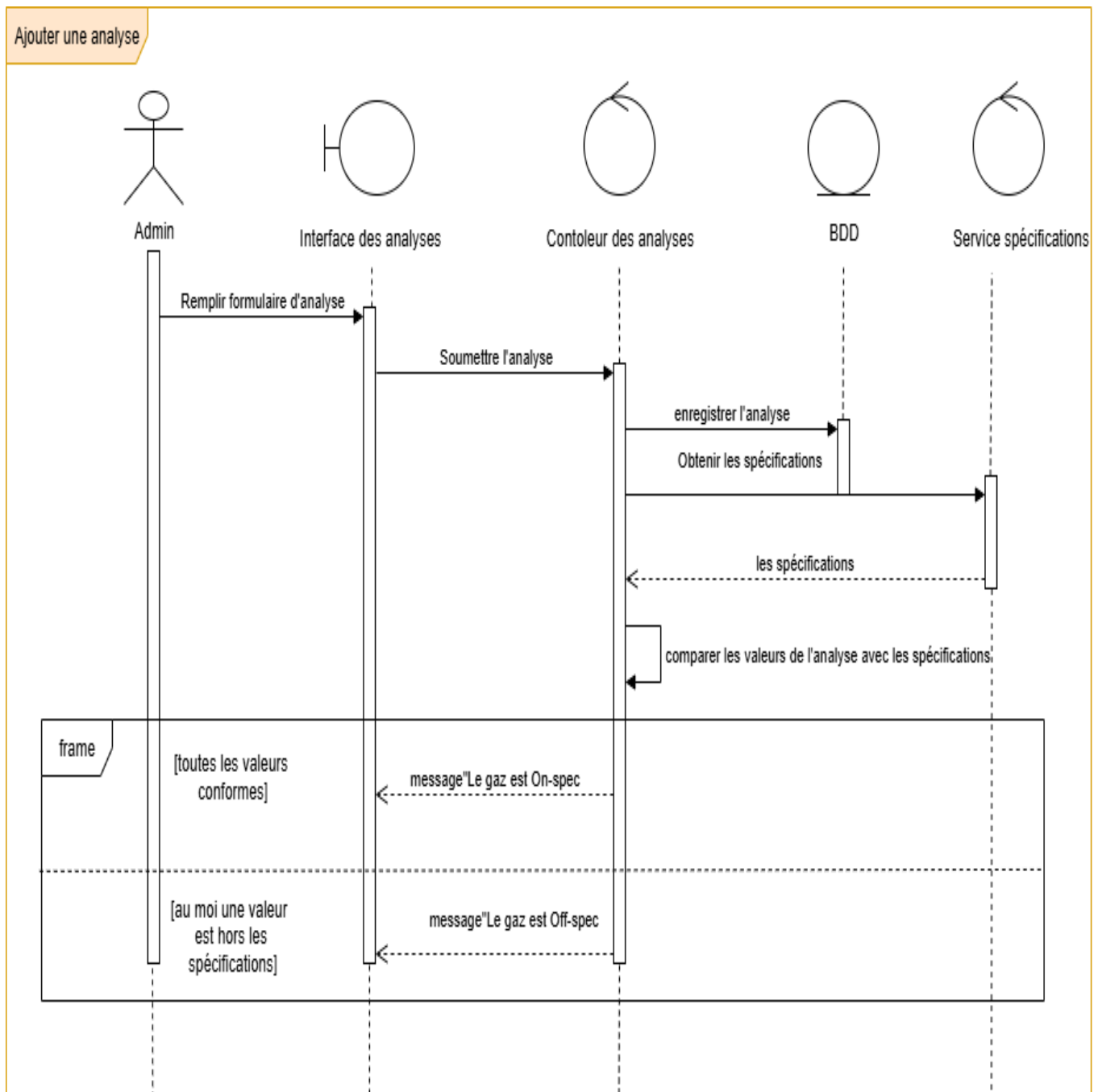


FIGURE 2.11 – Diagramme de séquence "Ajouter une analyse"

### 2.6.3 Diagramme de classe

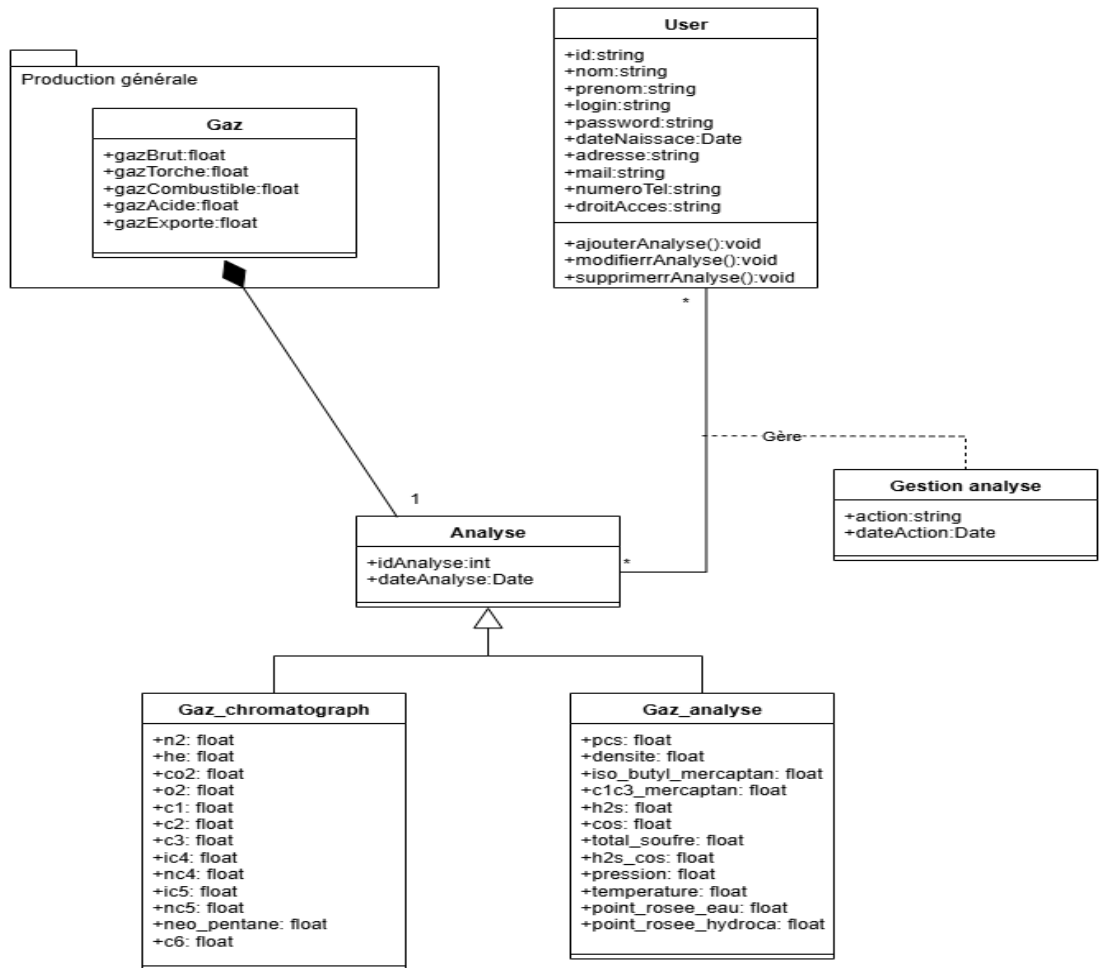


FIGURE 2.12 – Diagramme de classe "Gestion des analyses de gaz"

## 2.7 Le modèle relationnel

Avant de créer notre base de données, nous avons appliqué les règles de passage d'un diagramme de classes vers le modèle relationnel.

- `Departement(idDepartement, nomDepartement)`
- `Service(idService, nomService, #idDepartement)`

- User(idUser, nom, prenom, login, password, droitAccess, adresse, numeroTel, dateNaissance, mail, grade, rôle, #idService)
- Puit(idPuit, nomPuit, latitude, longitude, forageDate :Date, startDate, ciInjection, whType, whSize, formation, intervalle, TypePuit, prodVolume, dateProduction, dateIntegrite, dateIntervention, #idProd, #idIntegrit, #idIntervention, #idUser, #idManifold)
- Production\_infos(idProd, runtime, duse, whPression, flowPression, flowTemperature, volumeGaz, volumeEau, volumeCondensat)
- Integrité\_infos(idIntegrit, maop, annulaireA, annulaireB, annulaireC, annulaireD, etat)
- Intervention\_infos(idIntervention, description)
- Prévision(idPrévision, previsionCondensat, previsionGaz)
- Gérer\_Prévision(#idPrevision, #idUser, datePrevision)
- Gérer\_Production(#idProduction, #idUser, dateProduction)
- Production\_Générale(idProduction)
- Gaz(#idProduction, gazBrut, gazTorche, gazCombustible, gazAcide, gazExporte)
- Eau(#idProduction, eauAlbien, eauProcess)
- Condensat(#idProduction, condensatProduit, condensatExporte)
- Gérer\_Analyse(#idUser, #idAnalyse, action, dateAction)
- Analyse(idAnalyse, dateAnalyse, #idProduction)
- Gaz\_Chromatograph(#idAnalyse, n2, he, co2, o2, c1, c2, c3, ic4, nc4, ic5, nc5, neo\_pentane, c6)
- Gaz\_Analyse(#idAnalyse, pcs, densite, iso\_butyl\_mercaptan, c1c3\_mercaptan, h2s, cos, total\_soufre, h2s\_cos, pression, temperature, point\_rose\_eau, point\_rose\_hydroca)

## CONCLUSION

Au cours de ce chapitre, nous avons modélisé notre solution en suivant la méthode de conception UP avec le langage UML. Nous avons ainsi découpé notre projet en 6 packages, puis nous avons élaboré les diagrammes de cas d'utilisation, les diagrammes de séquence, les diagrammes de classes de conception et le modèle relationnel. Dans le prochain chapitre, nous passerons l'aspect pratique de l'implémentation à l'aide des outils logiciels qui seront introduits également dans cette prochaine partie.

# Chapitre 3

## Realisation de l'application

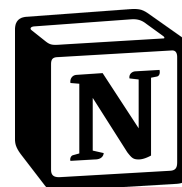
# Introduction

Ce chapitre est consacré à la réalisation et à la mise en œuvre de notre système de gestion et de suivi de la production gazière. Nous y présenterons l’environnement de développement adopté, les outils de collaboration utilisés, ainsi que les technologies logicielles retenues pour l’implémentation. Nous illustrerons également, les principales interfaces de notre application, en mettant en évidence les fonctionnalités développées pour répondre aux besoins des différents services impliqués dans la gestion des puits et du CPF.

## 3.1 Environnement et outils de développement

### 3.1.1 Outils de communication

**Notion** est un espace unique qui permet l’écriture et la planification. C’est une plateforme polyvalente qui permet la planification, la gestion de projets et même de diriger une entreprise entière. Notion offre la flexibilité nécessaire pour organiser les idées et les informations de la manière qui convient le mieux à chaque équipe. Nous avons utilisé Notion pour noter nos idées, nos interrogations et nos trouvailles ; il nous a permis également de stocker dans un endroit partagé la documentation des composants React.js que nous avons créés ainsi que les résumés et les bouts de code utiles des technologies que nous avons utilisées [21].



### Google meet

Google Meet est une plateforme de communication vidéo sécurisée développée par Google. Accessible à toute personne disposant d’un compte Google, elle permet d’organiser et de rejoindre des réunions en ligne, avec des fonctionnalités telles que les sous-titres instantanés, le partage d’écran, le cadrage automatique ou encore la suppression du bruit ambiant — notamment utile dans un cadre professionnel ou éducatif. Dans le cadre de ce projet, nous avons utilisé Google Meet pour assurer la coordination avec notre encadrant, organisant des points réguliers, en synchronisation avec Google Agenda, afin de suivre efficacement l’évolution des tâches [5].



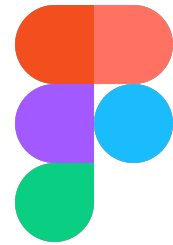
### 3.1.2 Outils de conception et prototypage

**Draw.io** est un outil puissant et flexible qui simplifie la création de diagrammes de manière collaborative en ligne. Sa gratuité, sa large gamme de fonctionnalités et sa compatibilité avec d’autres formats en font un



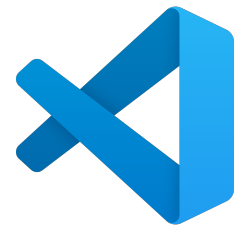
choix populaire parmi les professionnels de différents domaines. Nous avons choisi d'utiliser Draw.io via l'extension VSCode prévu à cet effet et le partage via GitHub, ce qui nous a permis de concevoir les diagrammes de cas d'utilisations, de séquences, de classes et le diagramme de déploiement. — Site officiel de Draw.io : <https://draw.io>

**Figma** est une plateforme collaborative pour l'édition de graphiques vectoriels et le prototypage. Elle permet de créer des design systèmes, facilite la conception de sites web et d'applications mobiles, et offre des fonctionnalités de design, de prototypage et de gestion de projet avancées. Figma est accessible via les navigateurs web, dispose d'une version bureau pour macOS et Windows, ainsi que d'une solution de visualisation des designs sur iOS et Android [1]. — Site officiel de Figma : <https://figma.com>



### 3.1.3 Environnement de codage

**Visual Studio Code** est un éditeur de code apprécié pour sa légèreté, sa rapidité et sa polyvalence. Il est largement utilisé et offre un excellent support intégré pour JavaScript et TypeScript, avec des fonctionnalités telles que la coloration syntaxique, l'autocomplétion et le débogage. De plus, son écosystème d'extensions nous a permis de personnaliser notre environnement de développement en fonction de nos besoins, notamment en utilisant les extensions Thunder Client, Live Share, Prettier, ESLint, etc. [19]

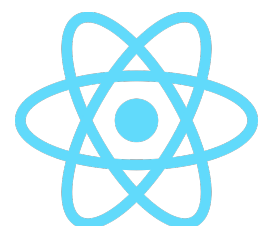


### 3.1.4 Languages et frameworks

**Django** est un framework web open source écrit en Python, conçu pour permettre le développement rapide d'applications web complexes et sécurisées. Il adopte une architecture **\*\*Model–Template–View (MTV)\*\***, favorisant la réutilisation des composants, la réduction du code redondant et la mise en place de fonctionnalités robustes telles que l'authentification, l'administration dynamique, la gestion des formulaires, et bien plus [4].



**React** est une bibliothèque JavaScript utilisée pour créer des interfaces utilisateur dans les applications web. Elle repose sur le concept de composants réutilisables, ce qui permet de simplifier le développement en évitant la répétition de code. Grâce à sa facilité d'apprentissage, sa documentation complète et sa communauté active, React offre de nombreuses opportunités professionnelles. De plus, il améliore les performances grâce à son DOM virtuel et à des bibliothèques telles



que React Router, qui permettent le chargement rapide de différentes pages sans rechargement. Enfin, React est hautement extensible, laissant aux développeurs la liberté de choisir les outils et bibliothèques complémentaires adaptés à leurs besoins [15].

**Tailwind CSS** est un framework CSS *utility-first*, open-source et très personnalisable, conçu pour permettre la construction rapide d'interfaces utilisateur sans jamais quitter le HTML. Il fournit un ensemble de classes utilitaires simples (taille, couleur, marges, flexbox, etc.) que l'on combine directement dans le markup pour créer des designs sur mesure sans styles prédéfinis ni surcharge. Cette approche encourage la modularité, la maintenabilité et le prototypage rapide, en offrant toute la flexibilité nécessaire aux développeurs modernes [30].



**i18next** est un framework d'internationalisation JavaScript complet, permettant de traduire facilement des applications web, mobiles ou serveurs en prenant en charge pluriels, interpolation, contexte, formatage, et chargement dynamique des traductions. Il s'intègre avec divers frameworks front-end (React, Angular, Vue.js, etc.) ainsi qu'avec Node.js ou Deno, offrant une solution modulaire, extensible et performante pour gérer l'internationalisation à l'échelle [11].



### 3.1.5 Bibliothèques

**React Toastify** est une bibliothèque JavaScript qui nous a permis d'ajouter facilement des notifications personnalisables, interactive et avec de jolies animations. — *Repository officiel de React Toastify* <https://github.com/fkhdra/react-toastify>

**react-country-flag** permet d'afficher aisément les drapeaux des pays sous forme de composants React. Utile dans les fonctionnalités liées à la langue ou à la région, elle permet d'enrichir l'interface utilisateur de manière visuelle et conviviale. — *Repository officiel de* <https://github.com/danalloway/react-country-flag>

**Print JS** est une petite bibliothèque JavaScript pour faciliter l'impression de fichier PDF ou autre à partir du Web. — *Documentation officielle de Print JS* <https://printjs.crabbly.com>

**@heroicons/react** est une bibliothèque d'icônes SVG de haute qualité conçue par les créateurs de Tailwind CSS, fournie sous licence MIT. Elle permet d'importer individuellement des icônes comme composants React, disponibles en styles *outline* ou *solid*, et dans plu-



sieurs tailles (16, 20 ou 24 px). Cette approche facilite leur stylisation (via classes Tailwind ou attributs CSS) et leur intégration harmonieuse dans les interfaces utilisateur développées avec React [29].

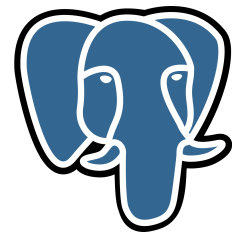
**La bibliothèque xlsx** (version 0.18.5) est une solution JavaScript open-source robuste, conçue pour l'analyse et la génération de feuilles de calcul au format XLSX. Elle permet d'extraire, de manipuler et de produire des fichiers Excel directement depuis le navigateur ou un environnement Node.js. Richesse fonctionnelle : lecture/écriture de données, conversion de tables HTML, export de fichiers Excel et compatibilité étendue avec des formats divers (read/write) [26].

**jsPDF** est une bibliothèque JavaScript permettant de générer des fichiers PDF directement côté client, sans nécessiter de serveur intermédiaire. Elle prend en charge la création de documents avec du texte, des images, des graphiques et diverses mises en page, offrant ainsi une grande flexibilité pour la production de rapports ou de factures dans les applications web. Grâce à son API simple et à sa compatibilité avec de nombreux navigateurs, **jsPDF** est largement adoptée pour des usages allant de l'export de données à la génération de documents interactifs [20].



### 3.1.6 Base de données

**PostgreSQL** est un système de gestion de base de données objet-relationnel (ORDBMS) open source, puissant et robuste. Il utilise et étend le langage SQL tout en offrant de nombreuses fonctionnalités qui permettent de stocker et faire évoluer en toute sécurité des charges de travail de données complexes. Les origines de PostgreSQL remontent à 1986, dans le cadre du projet POSTGRES mené à l'Université de Californie à Berkeley, et il bénéficie de plus de 35 années de développement actif [24].



**L'ORM** (Object-Relational Mapping) de Django est un composant intégré qui permet de manipuler la base de données à travers des objets Python, sans avoir à écrire directement des requêtes SQL. Chaque table est représentée par une classe (modèle), et chaque enregistrement devient une instance de cette classe. Ce mécanisme facilite la portabilité entre différents moteurs de bases de données, améliore la lisibilité du code et réduit les risques d'erreurs syntaxiques. Dans notre projet, l'ORM de Django a été utilisé pour interagir avec PostgreSQL de manière efficace et sécurisée [?].

## 3.2 Charte graphique

Dans cette section dédiée à la charte graphique, nous allons explorer l'univers visuel et identitaire de notre projet. La charte graphique joue un rôle essentiel car elle garantit la cohérence et l'attrait visuel de notre solution. Elle définit les éléments fondamentaux de l'identité visuelle, à savoir le nom, le slogan, les couleurs, la typographie et le logo, afin de construire une image claire et professionnelle autour de notre application **GasScope**.

### 3.2.1 Identité de marque

Le nom de notre projet est « **GasScope** », un terme composé de *Gas* (gaz) et *Scope* (vision, périmètre). Ce choix illustre parfaitement la vocation de notre solution : fournir une vision complète et centralisée sur les puits de gaz et leurs données de production.

En complément, nous avons retenu un slogan évocateur :

« *360° vision for gas wells* »

Ce slogan traduit l'idée de suivi global et de surveillance intégrée, permettant aux ingénieurs et aux techniciens d'avoir une vision à la fois exhaustive et détaillée de leurs opérations.

### 3.2.2 Couleurs

Pour construire l'identité visuelle de GasScope, nous avons défini deux couleurs principales, symbolisant à la fois la fiabilité, la modernité et l'innovation :

- **Couleur primaire** : Un *bleu pétrole* profond, code hexadécimal #1A3C63, représentant la stabilité, la confiance et la rigueur scientifique. Cette couleur est utilisée principalement pour le nom de l'application et les titres principaux.
- **Couleur secondaire** : Un *cyan lumineux*, code hexadécimal #21C4C0, incarnant la modernité, l'énergie et l'innovation technologique. Cette couleur est utilisée pour le slogan et certains éléments de mise en valeur dans l'interface.

L'utilisation cohérente de ces deux teintes permet de créer une identité visuelle harmonieuse et immédiatement reconnaissable.

### 3.2.3 Typographie

La typographie occupe une place centrale dans la charte graphique de GasScope. Elle a été choisie pour refléter à la fois la lisibilité, la modernité et le caractère professionnel du projet :

- **Texte principal (logo et interface)** : Montserrat (sans-serif), une police moderne et élégante qui offre un excellent équilibre entre sobriété et visibilité.

- **Texte secondaire (paragrapes, descriptions techniques)** : Roboto, une police claire et lisible sur tous les supports, notamment pour les interfaces numériques.

Cette combinaison garantit une cohérence visuelle, avec une hiérarchie typographique claire entre les titres, les sous-titres et les paragraphes.

### 3.2.4 Logo

Le logo **GasScope** se veut simple, professionnel et percutant. Il se compose uniquement du nom de l'application, écrit en bleu pétrole (#1A3C63), accompagné en dessous du slogan « *360° vision for gas wells* » en cyan (#21C4C0).

Ce choix minimaliste met l'accent sur la lisibilité et la mémorabilité de la marque. La disposition verticale (nom en haut, slogan en bas) offre un équilibre visuel qui renforce l'identité du projet.

Le logo conserve son impact quelle que soit la taille ou le support, grâce à la sobriété de sa conception. Il peut être utilisé en version agrandie pour des affiches et présentations, ou en version réduite pour des interfaces et icônes sans perte de lisibilité.



FIGURE 3.1 – Logo officiel de l'application GasScope

## 3.3 Présentation de la plateforme

Dans cette section, nous présentons les interfaces principales de notre application

### 3.3.1 L'interface utilisateur "Interface de connexion"

La figure 3.2 présente l'interface utilisateur "Interface de connexion". Elle illustre l'écran initial permettant à l'utilisateur de saisir ses identifiants afin d'accéder au système.

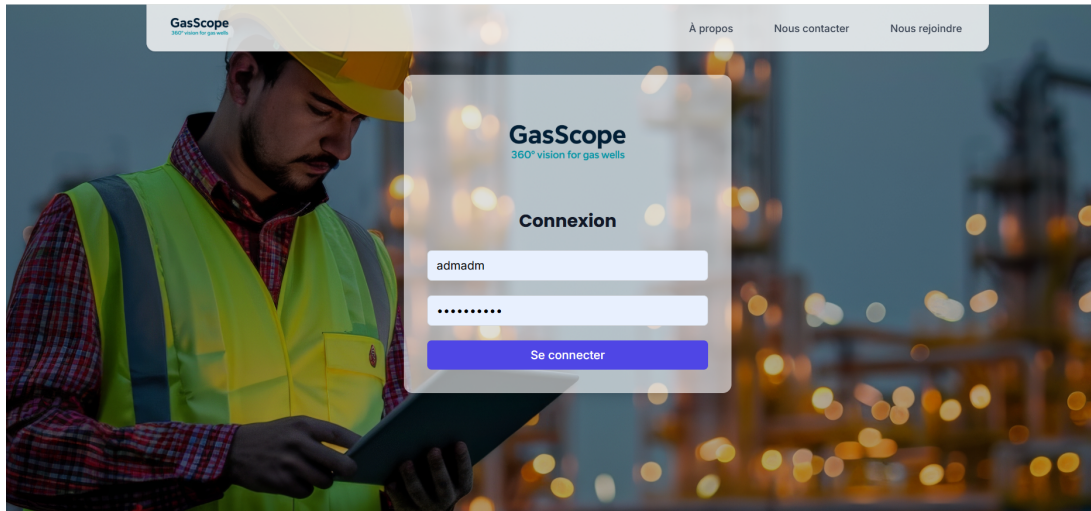


FIGURE 3.2 – Interface utilisateur "Interface de connexion"

### 3.3.2 L'interface utilisateur "Interface de gestion des utilisateurs"

La figure 3.3 présente l'interface utilisateur "Interface de connexion". Elle illustre l'écran initial permettant à l'utilisateur de saisir ses identifiants afin d'accéder au système.

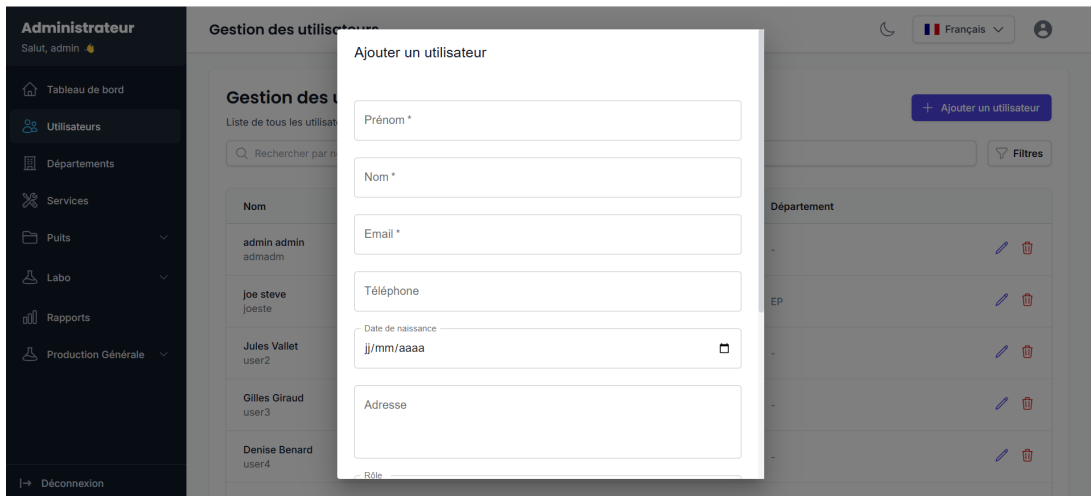


FIGURE 3.3 – Interface utilisateur "Interface de gestion des utilisateur"

### 3.3.3 L'interface utilisateur "Interface de saisie de production journalière"

La figure 3.4 et 3.5 présente l'interface utilisateur "Interface de saisie de production journalière". Elle offre la possibilité d'enregistrer les données quotidiennes relatives au gaz produit, au condensat, à l'eau, ainsi qu'à la consommation électrique. Elle permet également la saisie des principaux paramètres opératoires, tels que la pression d'entrée et de sortie au niveau du CPF.

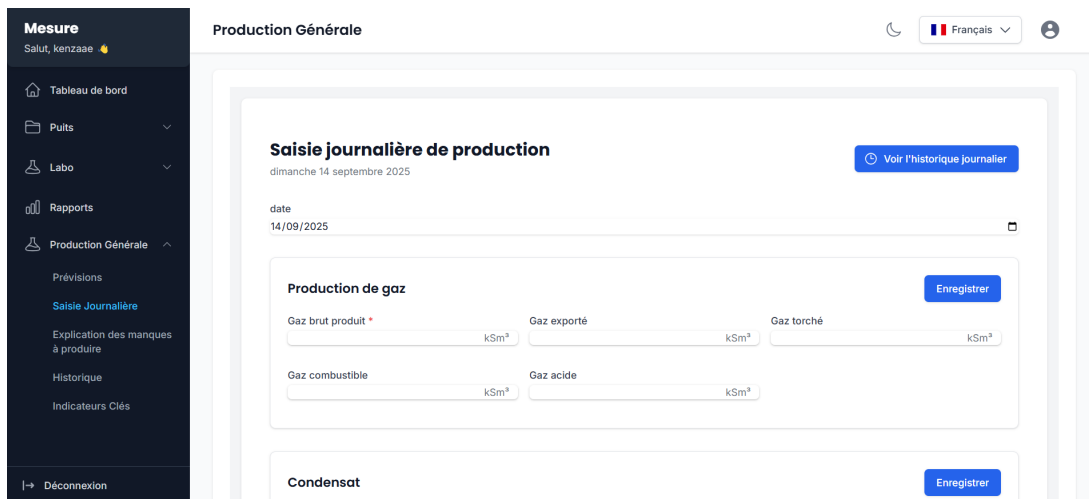


FIGURE 3.4 – Interface utilisateur "Interface de saisie de production journalière"

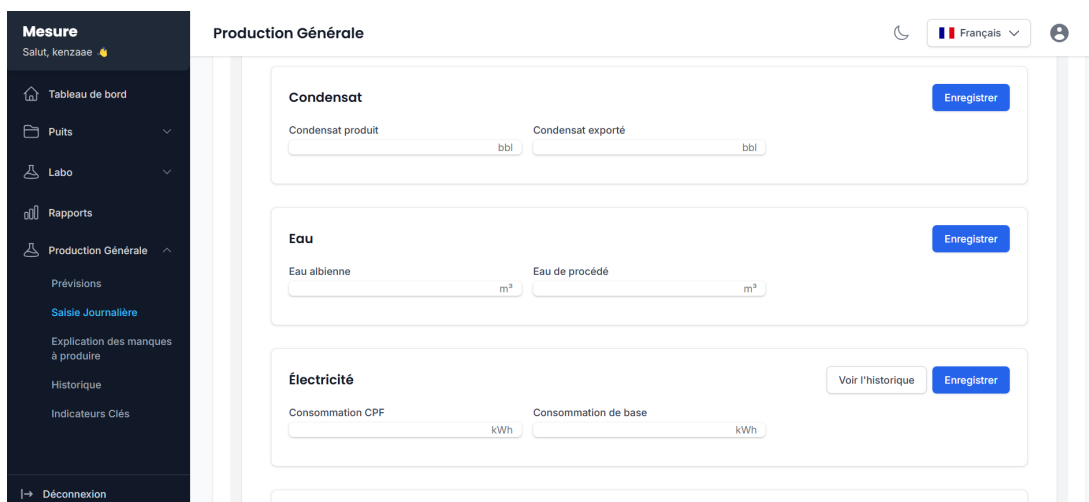


FIGURE 3.5 – Interface utilisateur "Interface de saisie de production journalière-suite"

### 3.3.4 L'interface utilisateur "Interface de saisie des prévisions journalière"

La figure 3.6 présente l'interface utilisateur "Interface de saisie des prévisions journalière". Elle permet de renseigner les prévisions quotidiennes concernant le volume de condensat à exporter ainsi que le volume de gaz à expédier. À partir de l'interface de saisie des prévisions journalières, l'utilisateur peut consulter l'historique des prévisions en cliquant sur le bouton « Historique des prévisions », situé en haut de l'écran. Cette action redirige vers l'interface illustrée dans la figure 3.7.

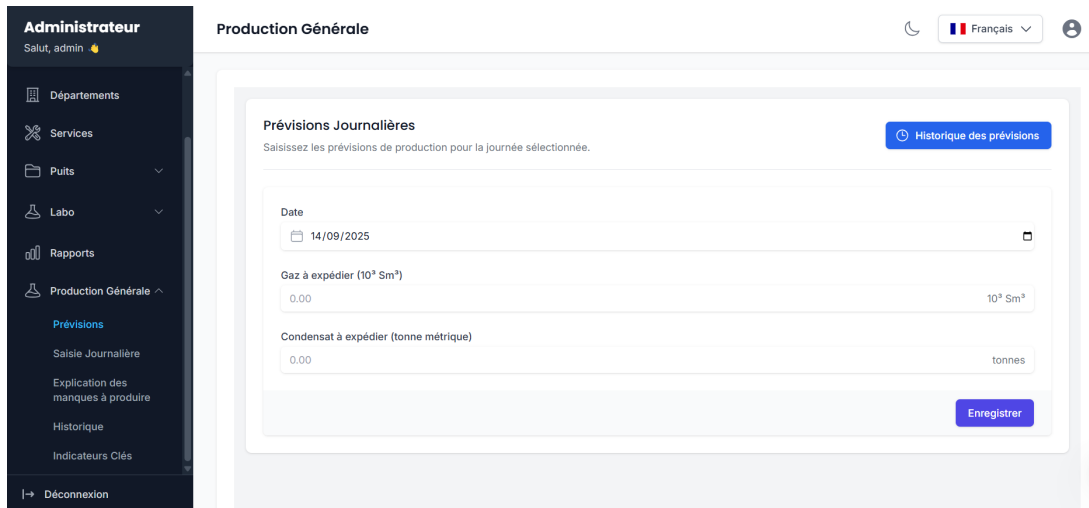


FIGURE 3.6 – Interface utilisateur "Interface de saisie des prévisions journalière"

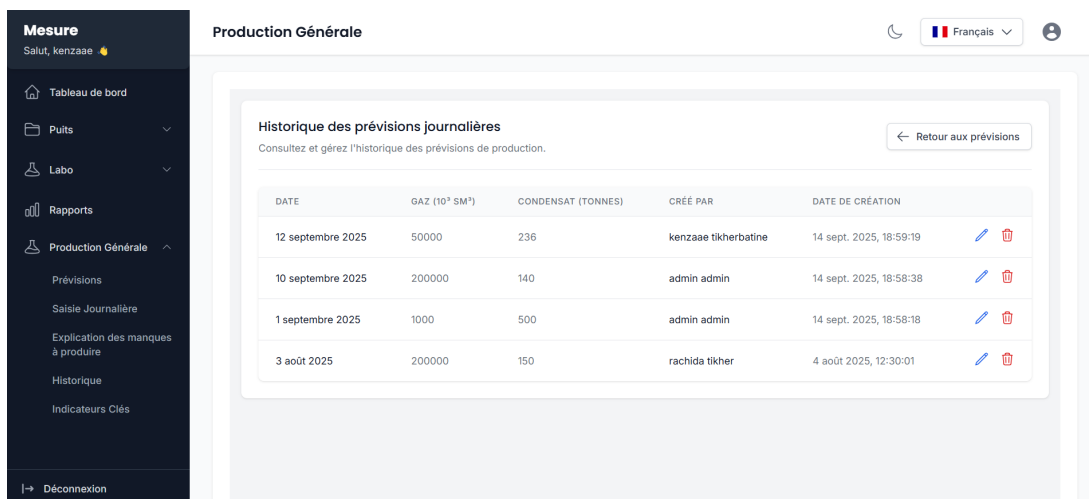


FIGURE 3.7 – Interface utilisateur "historique des prévisions journalière saisies"

### 3.3.5 L'interface utilisateur "Interface de visualisation des KPI"

La figure 3.8 illustre l'interface de visualisation des KPI liés aux différentes productions, notamment l'eau, le gaz et l'énergie électrique consommée.



FIGURE 3.8 – Interface utilisateur "Interface de visualisation des KPI"

### 3.3.6 L'interface utilisateur "Interface de saisir des analyses labo"

La figure 3.9 illustre l'interface saisie des analyses des échantillants du gaz produit.

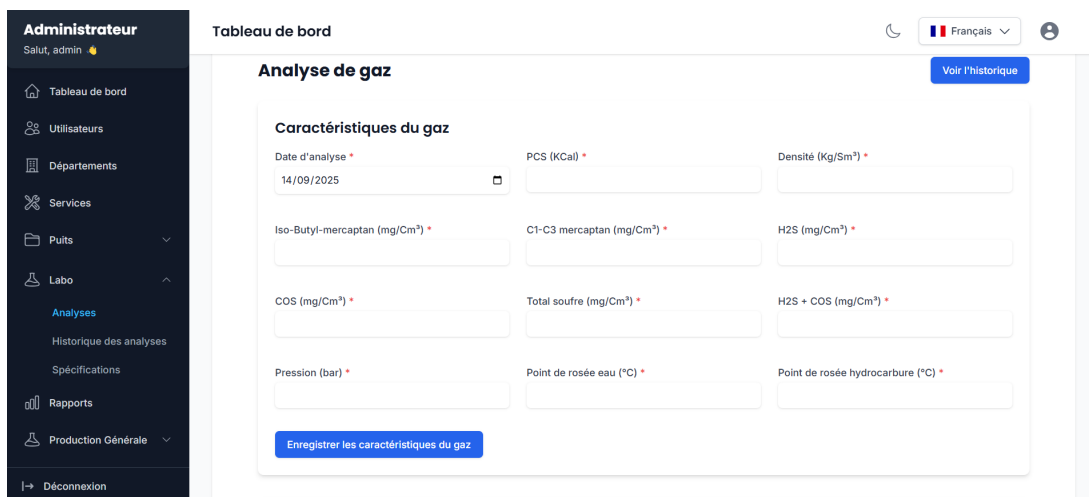


FIGURE 3.9 – Interface utilisateur "Interface de saisir des analyses labo"

## 3.4 Conclusion

Dans ce chapitre, nous avons introduit l'environnement de développement dans lequel le projet a été réalisé. Nous avons également présenté la charte graphique. Enfin, nous avons offert un aperçu concret de la plateforme à travers quelques captures d'écran.

# Conclusion Générale

En somme, ce mémoire met en lumière la nécessité cruciale de moderniser la gestion et l'analyse des données au sein des plateformes gazières, notamment au niveau du GRN. Les limites des systèmes existants, telles que le manque d'automatisation, les interfaces peu ergonomiques et les difficultés d'accès ou d'extraction des données, entravent significativement l'efficacité des ingénieurs ainsi que la rapidité de la prise de décision. La mise en œuvre d'une solution numérique performante apparaît donc indispensable pour optimiser les opérations et améliorer la qualité du suivi de la production. C'est dans cette perspective qu'a été conçue et réalisée **GasScope**, une plateforme visant à offrir une gestion intelligente et une analyse dynamique des données.

Tout au long du projet, des avancées significatives ont été réalisées. Le premier chapitre a permis une compréhension approfondie de l'environnement du GRN, une analyse des systèmes en place ainsi qu'une définition précise de la problématique et des objectifs de la solution. Le deuxième chapitre a établi une architecture robuste et détaillée pour **GasScope** à travers l'application de la méthodologie UML et UP, permettant de modéliser les besoins fonctionnels et non fonctionnels, d'identifier les acteurs et leurs interactions via divers diagrammes, ainsi que d'élaborer le modèle relationnel de la base de données. Enfin, le troisième chapitre a concrétisé cette conception en décrivant l'environnement de développement et les outils technologiques mobilisés, notamment Django pour le backend, React pour le frontend, Tailwind CSS pour le stylisme, ainsi que PostgreSQL pour la gestion des données.

Bien que les principaux axes du travail aient été menés à terme et que l'ensemble des objectifs fixés ait été atteint, certains aspects restent perfectibles. Ces éléments, qui n'affectent pas la réalisation globale de la plateforme, ont été identifiés et sont présentés dans les perspectives suivantes afin de proposer des améliorations et optimisations futures.

Pour les perspectives d'évolution, plusieurs pistes peuvent être envisagées afin d'enrichir davantage **GasScope** :

- **Finalisation et approfondissement de la présentation visuelle** : Achè-

vement de la charte graphique et élaboration d'une présentation exhaustive de l'application accompagnée de captures d'écran et d'explications détaillées.

- **Extension des fonctionnalités du service LABO** : Centralisation renforcée des données analytiques du laboratoire avec l'intégration d'outils de visualisation avancée et de reporting spécialisé.
- **Développement d'outils d'analyse prédictive** : Exploitation des données historiques du PHD pour implémenter des modèles de prédiction liés aux pannes d'équipement, aux fluctuations de production ou aux anomalies.
- **Intégration mobile** : Développement d'une application mobile destinée aux profils travaillant sur le terrain afin de faciliter l'accès aux informations essentielles et aux notifications.
- **Amélioration continue de l'ergonomie** : Affinement de l'interface utilisateur pour la rendre plus intuitive, plus fluide et adaptable selon les rôles et préférences des utilisateurs.

Ces perspectives permettront de renforcer progressivement l'efficacité de **GasScope** et de maximiser sa valeur ajoutée pour le Groupement Reggane Nord.

# Bibliographie

- [1] BDM (Blog du Modérateur). Figma – un outil de prototypage et de design collaboratif, 2025. Consulté le 10 juillet 2025. URL : <https://www.blogdumoderateur.com/tools/figma>.
- [2] Berkati.A and Ouaihouche.A. Conception et réalisation d’une application web pour le suivi et le traitement des agressions sur les ouvrages hydrocarbures de la sonatrach. Master’s thesis, Mémoire de fin d’étude, Master en Informatique – Option Génie Logiciel, 2023.
- [3] Cisco. What is a dmz in networking?, 2025. Consulté en 2025. URL : <https://www.cisco.com/>.
- [4] Django Software Foundation. Django – the web framework for perfectionists with deadlines, 2025. Consulté le 12 août 2025. URL : <https://www.djangoproject.com/>.
- [5] Google. Google meet – vidéoconférences sécurisées, 2025. Consulté le 11 juillet 2025. URL : <https://meet.google.com>.
- [6] Guemache.D and Hocine.R. Conception et réalisation d’une application web pour la gestion de l’étalonnage des instruments de mesures : Cas sonatrach. Master’s thesis, Mémoire de fin de cycle, Master Professionnel en Informatique – Option Génie Logiciel, 2022.
- [7] Habbache.M and Megri.H. Conception et réalisation d’une plateforme de services médicaux (don de sang, rdv, médicaments). Master’s thesis, Mémoire de Fin de Cycle, Master Informatique – Option Génie Logiciel, 2021. Examiné le 14 octobre 2021.
- [8] Honeywell. *Uniformance PHD User Manual*, 2024. Documentation officielle du PHD.
- [9] Honeywell. Distributed control system (dcs) overview, 2025. Consulté en 2025. URL : <https://www.honeywellprocess.com/>.
- [10] Honeywell. Uniformance process studio overview, 2025. Consulté en 2025. URL : <https://www.honeywell.com/>.
- [11] i18next Team. i18next – learn once, translate everywhere, 2025. Consulté le 12 août 2025. URL : <https://www.i18next.com/>.

- [12] IBM. Diagrammes de cas d'utilisation uml, 2025. Consulté le 25/03/2025. URL : <https://www.ibm.com/docs/fr/rhapsody/9.0.1?topic=uml-use-case-diagrams>.
- [13] IBM. Diagrammes de classes uml, 2025. Consulté le 25/03/2025. URL : <https://www.ibm.com/docs/fr/rhapsody/9.0.1?topic=uml-class-diagrams>.
- [14] IBM. Modèles uml, 2025. Consulté le 25/03/2025. URL : <https://www.ibm.com/docs/fr/dmrt/9.5.0>.
- [15] Kinsta. Un regard sur la bibliothèque javascript populaire, 2022. Consulté le 10 juin 2025. URL : <https://kinsta.com/fr/base-de-connaissances/qu-est-react-js/>.
- [16] L. Lahlou. *Conception et réalisation d'un portail web pour la gestion des stocks, cas d'étude magasin de la faculté des sciences exactes de l'université de Béjaïa*. Phd thesis, Université Abderrahmane Mira de Béjaïa.
- [17] Lekhel. Partenariat sonatrach, repsol et wintershall dea, 2025. Consulté en 2025.
- [18] Bernard Marr. *Key Performance Indicators : The 75+ Measures Every Manager Needs to Know*. Pearson Education, 2015.
- [19] Microsoft. Documentation officielle de vscode, 2025. Consulté le 9 août 2025. URL : <https://code.visualstudio.com/docs>.
- [20] MrRio. jspdf – bibliothèque javascript pour générer des fichiers pdf côté client, 2025. Consulté le 15 septembre 2025. URL : <https://github.com/parallax/jsPDF>.
- [21] Notion Labs. Notion – workspace collaboratif tout-en-un, 2025. Consulté le 10 juillet 2025. URL : <https://www.notion.so>.
- [22] Oulhaci.Y and Mouhous.S. Conception et réalisation d'un système d'information hospitalier (sih). Master's thesis, Mémoire de fin de cycle, Master Professionnel en Informatique – Option Génie Logiciel, 2023.
- [23] Peloton. Wellmaster integrity management system (wims), 2025. Consulté en 2025. URL : <https://www.peloton.com/>.
- [24] PostgreSQL Global Development Group. About — postgresql, 2025. Consulté le 12 août 2025. URL : <https://www.postgresql.org/about/>.
- [25] P. Roques and F. Vallée. *UML 2 en action : De l'analyse des besoins à la conception*. Eyrolles, 4th edition, 2007.
- [26] SheetJS. xlsx – bibliothèque javascript pour manipulation de fichiers excel, 2025. Consulté le 15 septembre 2025. URL : <https://github.com/SheetJS/sheetjs>.
- [27] SLB. Avocet production operations software, 2025. Consulté en 2025. URL : <https://www.slb.com/>.

- [28] Taguilsa.Rafik and Tighidet.M. Conception et réalisation d'une application desktop de gestion pour la section relex de sonatrach – béjaia. Master's thesis, Mémoire de Fin de Cycle, Master Professionnel en Informatique – Option Génie Logiciel, 2023.
- [29] Tailwind Labs. @heroicons/react – bibliothèque d'icônes svg pour react, 2025. Consulté le 15 septembre 2025. URL : <https://github.com/tailwindlabs/heroicons>.
- [30] Tailwind Labs. Tailwind css – utility-first css framework, 2025. Consulté le 12 août 2025. URL : <https://tailwindcss.com/>.
- [31] B. TIDJET and K. TIDJET. “conception et réalisation d'une plateforme e-santé”. Master's thesis, Université A. MIRA – Béjaïa, Béjaïa, Algérie, Juin 2023.
- [32] Automation World. Définition et rôle d'un plc, 2025. Consulté en 2025. URL : <https://www.automationworld.com/>.

# ANNEXE

## Mon Expérience de Stage au sein de GRN

### **De l'Aéroport au Site de GRN ( Une Entrée dans un Nouvel Univers)**

Tout a commencé à l'aéroport international Houari Boumediène, où j'ai pris mon vol pour Adrar. Durant ce voyage d'une heure et quarante-cinq minutes, j'ai profité de la vue aérienne magnifique de mon pays, sans vraiment savoir ce qui m'attendait une fois sur place. À mon arrivée à Adrar, j'ai été accueillie par un employé de GRN, qui s'est assuré que moi et les deux autres stagiaires, Amira et Dalia, embarquions pour la base via un jet affrété. L'atterrissage sur la piste privée du site marquait le début de cette aventure professionnelle. Après un contrôle de sécurité rigoureux dans la frontière de la "green zone", nous avons été conduits à la base de vie de GRN, où un accueil chaleureux nous attendait. L'installation était impressionnante, une chambre tout confort digne d'un hôtel, un service de restauration de qualité, et une organisation impeccable. Nous avons ensuite été dirigés vers l'administration pour les formalités d'usage. Monsieur Omar Omari, responsable de la formation, nous a présenté les différentes étapes de notre intégration et Il nous a remis nos tenues de travail. S'en est suivie une induction HSE (Hygiène, Sécurité, Environnement) encadrée par Monsieur Walid Nekkache, nous initiant aux réflexes à adopter en cas de danger. Après une visite médicale, la journée s'est achevée sur une rencontre avec Monsieur Bilal, ingénieur en instrumentation, qui nous a offert une visite guidée de la base et des conseils précieux sur le bon déroulement du stage.

### **Plongée dans l'Univers du CPF**

Le lendemain, j'ai entamé mon expérience au CPF (Central Processing Facility). Après une induction HSE spécifique au site, j'ai effectué les formalités administratives pour obtenir mon badge d'accès.

Mon immersion a débuté au sein du département EP avec Monsieur Berkat Mohamed, ingénieur mesure. Au départ, il n'avait pas saisi l'objectif de mon projet, mais en approfondissant nos échanges, j'ai identifié un besoin crucial : disposer d'une interface synthétisant les informations essentielles, évitant ainsi une navigation fastidieuse à travers les multiples onglets du logiciel AVOCET. Après une analyse approfondie, j'ai proposé une

interface intuitive qui répond à son besoin. J'ai ensuite approfondi mes connaissances sur l'intégrité des puits en rencontrant Monsieur Laasekri Belaid, ingénieur Well Integrity, qui m'a présenté le système WIMS, indépendant d'AVOCET. Une fois les maquettes validées, il me restait à collecter les données nécessaires.

Une sortie terrain à Reggane était prévue pour les stagiaires en instrumentation, et j'ai eu la chance d'y participer. Cette expérience était enrichissante à plusieurs niveaux : j'ai observé l'architecture d'un puits de gaz, le fonctionnement des capteurs et des automates, et le réseau de communication des données.

De retour, j'ai commencé à travailler avec Monsieur Abdelhamid Tabi, ingénieur exploitation. Cette fois-ci, j'ai adopté une approche différente, tirant les leçons de mon expérience avec le département EP. J'ai affiné mon analyse des besoins, en appliquant les méthodes d'analyse apprises durant mon cursus, et en mettant en pratique mes connaissances théoriques sur le traitement du gaz.

J'ai également eu l'opportunité de rencontrer un ingénieur système, qui m'a expliqué le processus de stockage des données et l'architecture des serveurs, enrichissant ainsi ma compréhension globale du système d'information de GRN. Quelques heures avant la fin de mon stage, j'ai visité le laboratoire afin d'identifier leurs besoins. J'ai relevé la nécessité d'une centralisation des données analytiques, un aspect que je pourrais considérer dans de futurs développements. Mon rapport de stage a été validé avec une note de 17, et je l'ai déposé auprès du département ITC.

### **Conclusion**

Mon stage au sein de GRN a été une opportunité cruciale ce n'est pas seulement résumé à une expérience académique. Il a changé ma perception du monde professionnel et renforcé ma motivation à innover. J'ai compris que les difficultés ne sont pas des obstacles, mais des opportunités d'apprentissage. J'ai découvert le rôle concret d'un ingénieur, ses responsabilités et sa capacité à faire face aux défis quotidiens. Enfin, cette expérience m'a fait réaliser l'immense potentiel de notre pays. L'Algérie est une force en devenir, et c'est à nous, jeunes ingénieurs et futurs entrepreneurs, de créer les moyens de son développement, plutôt que de nous plaindre du manque de ressources. L'avenir est entre nos mains.

# Résumé

Ce mémoire de fin d'études, réalisé dans le cadre de l'obtention du diplôme de Master en Génie Logiciel, porte sur la conception et la réalisation d'une plateforme de gestion et d'analyse des données de production gazière. La problématique a été identifiée au sein du Groupement Reggane Nord (GRN), où les systèmes existants présentent des limites importantes : absence d'automatisation des indicateurs de performance (KPI), extractions manuelles fastidieuses et interfaces peu ergonomiques.

L'objectif principal du projet est de proposer une solution moderne et intuitive, nommée **GasScope**, permettant d'automatiser le calcul des KPI, de visualiser les données de manière claire et dynamique, et de faciliter la prise de décision des ingénieurs. Pour y parvenir, nous avons adopté le **Processus Unifié (UP)** et le langage **UML** pour modéliser les besoins, les cas d'utilisation et l'architecture du système. La réalisation s'appuie sur **Django** pour le backend, **React** et **Tailwind CSS** pour le frontend, et **PostgreSQL** pour la base de données.

**Mots-clés** : Gestion des données, Plateformes gazières, KPI, UML, UP, Django, React, PostgreSQL.

# Abstract

This Master's thesis focuses on the design and implementation of a data management and analysis platform for gas production facilities. The study was conducted within the context of the **Reggane Nord Group (GRN)**, where existing systems show significant limitations, such as the lack of automated performance indicators (KPIs), time-consuming manual data extraction, and non-intuitive interfaces.

The main objective of this project is to provide a modern and user-friendly solution, named **GasScope**, designed to automate KPI calculation, ensure clear and dynamic data visualization, and support fast decision-making for engineers. To achieve this, we adopted the **Unified Process (UP)** and the **Unified Modeling Language (UML)** to specify requirements, use cases, and system architecture. The implementation relies on **Django** for the backend, **React** and **Tailwind CSS** for the frontend, and **PostgreSQL** as the database system.

**Keywords** : Data management, Gas production platforms, KPI, UML, UP, Django, React, PostgreSQL.