

**République Algérienne Démocratique et Populaire**  
**Ministère de l'Enseignement Supérieur et de la Recherche Scientifique**  
**Université A. MIRA - Bejaïa**

**Faculté des sciences exactes**  
**Département informatique**  
**Systeme LMD**



***Mémoire de Fin de Cycle***

En vue de l'obtention du diplôme d'un Master professionnel

Option : Génie Logiciel

***Thème***

Optimisation de la Planification Industrielle par la  
Prédiction à l'Aide de Techniques Avancées ( Cas : Sonelgaz  
Distribution Direction de Béjaïa)

**Réalisé et Présenté par :**

➤ M<sup>elle</sup> Drias Mira

**Devant le jury composé de :**

- |                    |                                  |            |                   |
|--------------------|----------------------------------|------------|-------------------|
| • <b>Président</b> | M <sup>r</sup> Achroufene Achour | <b>MCA</b> | Université A/Mira |
| • <b>Examineur</b> | M <sup>r</sup> Bedjou Khaled     | <b>MCD</b> | Université A/Mira |
| • <b>Encadrant</b> | M <sup>r</sup> Mir Foudil        | <b>MAA</b> | Université A/Mira |

**Année universitaire : 2023/ 2024**

# Remerciements

Je tiens avant tout à exprimer ma gratitude à Allah Tout-Puissant. Grâce à Dieu, j'ai pu atteindre mes objectifs et réaliser ce travail.

Tout d'abord, ce travail ne serait pas aussi riche et n'aurait pas pu voir le jour sans l'aide et l'encadrement de **M. MIR Foudil** et **M. HADJOUT Dalil**. Je les remercie pour la qualité de leur encadrement exceptionnel, pour leur patience, leur rigueur et leur disponibilité durant ma préparation de ce mémoire.

Je remercie les membres du jury d'avoir pris la peine de lire et de juger ce travail.

Mes remerciements s'adressent également à tous mes professeurs de la spécialité pour leur aide.

Mes profonds remerciements vont également à toutes les personnes qui m'ont aidé et soutenu de près ou de loin.

# Dédicaces

À tous ceux qui ont soutenu mon parcours, je dédie ce projet de fin d'études à toutes les âmes ayant joué un rôle exceptionnel dans ma vie. Votre soutien, vos encouragements et votre amour ont été essentiels à ma réussite.

À mes très chers parents, aujourd'hui, alors que j'achève mon mémoire de fin d'études, je tiens à vous dédier ces quelques lignes.

Maman, tu es la source de ma force et de ma détermination. Tu m'as appris à persévérer dans les moments difficiles, à croire en moi-même et à ne jamais abandonner mes rêves. Ta présence bienveillante et tes conseils avisés ont été essentiels pour surmonter les obstacles qui se dressaient sur mon chemin. Ton absence cette année a été douloureuse, mais ton souvenir m'a donné la force nécessaire pour continuer. Que tu reposes en paix, toujours dans mon cœur.

Papa, merci d'avoir toujours été à mes côtés et de m'avoir constamment encouragé. Ta présence bienveillante et ta capacité à me faire sentir capable de tout ont été mes plus grandes motivations..

À mon petit frère Amine et ma petite sœur Anies merci pour votre soutien constant et votre affection. Votre présence a été d'une importance inestimable tout au long de ce parcours.

À ma copine Hinan, Un merci immense pour ton aide précieuse et ton soutien constant. Tu as été un véritable pilier pour moi durant cette période. Aussi à mes copines Sissa et Samira, Merci pour votre amitié et votre soutien indéfectible. Votre présence a rendu ce chemin plus agréable.

À mes cousines Kahina et Kamila Merci pour vos encouragements et votre soutien tout au long de ce projet. Vous avez toujours cru en moi et cela m'a donné la force de persévérer.

Que cette dédicace leur parvienne, empreinte de ma profonde gratitude et de mon respect éternel. Que mes paroles reflètent l'amour et la reconnaissance qui habitent mon cœur envers chacun d'entre eux.

# Table des matières

<b>Table des figures</b>	<b>i</b>
<b>Liste des tableaux</b>	<b>iii</b>
<b>Liste des abréviations</b>	<b>v</b>
<b>Introduction générale</b>	<b>1</b>
<b>1 ETAT DE L'ART</b>	<b>3</b>
1.1 Introduction . . . . .	3
1.2 Séries Temporelles . . . . .	3
1.2.1 Définition des Séries Temporelles . . . . .	3
1.2.2 Décomposition des Séries Temporelles . . . . .	4
1.3 Modèles de Décomposition d'une série temporelle . . . . .	5
1.3.1 Modèle de Décomposition Additive . . . . .	5
1.3.2 Modèle de Décomposition Multiplicative . . . . .	5
1.4 Méthodes de prévision . . . . .	6
1.4.1 Moyenne Mobile (Moving Average) . . . . .	6
1.4.1.1 Définition . . . . .	6
1.4.1.2 Types de Moyennes Mobiles . . . . .	7
1.4.1.3 Avantages de la Moyenne Mobile . . . . .	7
1.4.1.4 Limitations de la Moyenne Mobile . . . . .	8
1.4.1.5 Représentation graphique des moyens mobiles . . . . .	8
1.4.2 Régression Linéaire . . . . .	8
1.4.2.1 Définition . . . . .	8
1.4.2.2 Avantages de la Régression Linéaire . . . . .	9
1.4.2.3 Limitations de la Régression Linéaire . . . . .	9
1.4.2.4 Représentation graphique de régression linière . . . . .	9
1.4.3 Méthode Exponentielle . . . . .	10
1.4.3.1 Définition . . . . .	10
1.4.3.2 Avantages des Méthodes Exponentielles . . . . .	11
1.4.3.3 Limitations des Méthodes Exponentielles . . . . .	11
1.4.3.4 Représentation graphique de méthode exponentielle . . . . .	11
1.4.4 ARIMA (AutoRegressive Integrated Moving Average) . . . . .	11
1.4.4.1 Definition . . . . .	11
1.4.4.2 Composantes du Modèle ARIMA . . . . .	12

1.4.4.3	Avantages de ARIMA . . . . .	12
1.4.5	Limitations de ARIMA . . . . .	12
1.4.6	SARIMA (Seasonal AutoRegressive Integrated Moving Average) . . . . .	13
1.4.6.1	Définition . . . . .	13
1.4.6.2	Avantages de SARIMA . . . . .	13
1.4.6.3	Limitations de SARIMA . . . . .	13
1.5	Modèles du Machine Learning . . . . .	14
1.5.1	Avantages des Méthodes basées sur l'apprentissage automatique . . . . .	15
1.5.2	Limitations des Méthodes basées sur l'apprentissage automatique . . . . .	16
1.6	Le Deep Learning . . . . .	16
1.6.1	Définition . . . . .	16
1.6.2	Algorithmes d'optimisation de Deep Learning . . . . .	16
1.6.3	Réseaux de neurones . . . . .	19
1.6.3.1	Définition . . . . .	19
1.6.3.2	Structure des Réseaux de Neurones . . . . .	19
1.6.3.3	Les types de réseaux de neurones . . . . .	20
1.7	Comment choisir la méthode à utiliser ? . . . . .	23
1.8	Conclusion . . . . .	25
<b>2</b>	<b>Les Transformers pour la Prédiction des Séries Temporelles</b>	<b>26</b>
2.1	Introduction aux Transformers . . . . .	26
2.1.1	Contexte et Motivation . . . . .	26
2.1.2	Comparaison avec les Méthodes Traditionnelles . . . . .	27
2.2	Architecture des Transformers . . . . .	28
2.2.1	Structure de Base . . . . .	28
2.2.2	Couches d'Encodage et de Décodage . . . . .	29
2.2.3	Modèles Pré-entraînés et Fine-tuning . . . . .	31
2.2.4	Optimisation et Algorithmes d'Entraînement . . . . .	31
2.3	Méthodes de Calcul d'Erreur . . . . .	32
2.3.1	Mean Squared Error (MSE) . . . . .	32
2.4	Conclusion . . . . .	32
<b>3</b>	<b>Analyse des besoins et Conception</b>	<b>33</b>
3.1	Introduction . . . . .	33
3.2	Présentation du sonalgaz . . . . .	33
3.3	Transformation de SONALGAZ en groupe . . . . .	34
3.4	Les différents types de client de sonalgaz : . . . . .	35
3.5	Présentation de l'organisme d'accueil (Sonalgaz Distribution Direction de Bejaia) . . . . .	35
3.5.1	Organisation de la Sonalgaz Distribution Direction de BÉJAIA . . . . .	36
3.5.1.1	Division Relations Commerciales . . . . .	36
3.6	Problématique . . . . .	38
3.7	Solutions proposées . . . . .	38
3.8	Gestion de projet . . . . .	39

3.8.1	Approches Traditionnelle et Agile pour une Gestion Efficace . . . . .	40
3.8.2	Scrum . . . . .	41
3.8.2.1	Les Trois Piliers de Scrum . . . . .	41
3.8.2.2	Les Rôles dans la Méthode Scrum . . . . .	41
3.8.2.3	Grandes Étapes de la Méthode Scrum . . . . .	42
3.9	Délimitation du domaine d'étude . . . . .	42
3.9.1	Répartition des rôles . . . . .	43
3.9.2	Identification des acteurs . . . . .	44
3.9.3	Diagramme de contexte . . . . .	44
3.10	Pilotage de notre projet avec SCRUM . . . . .	46
3.10.1	Product backlog . . . . .	46
3.10.2	Planification des sprints . . . . .	49
3.11	Diagramme de cas d'utilisation . . . . .	49
3.12	Description des cas d'utilisation . . . . .	51
3.13	Diagrammes de séquences . . . . .	51
3.13.1	Cas d'utilisation « S'authentifier» . . . . .	52
3.13.2	Cas d'utilisation « Visualiser les consommations» . . . . .	54
3.13.3	Cas d'utilisation « Prédire la consommation » . . . . .	56
3.13.4	Cas d'utilisation « Comparaison de la consommation » . . . . .	58
3.13.5	Cas utilisation “ prédire la consommation d'un client” . . . . .	60
3.14	Diagramme de séquence détaillé . . . . .	62
3.14.1	Diagramme de séquence détaillé du cas d'utilisation ”S'authentifier” : . . . . .	62
3.14.2	Diagramme de séquence détaillé du cas d'utilisation ”Visualiser les consommations” . . . . .	63
3.14.3	Diagramme de séquence détaillé du cas d'utilisation ”Prédire les consommations ” . . . . .	64
3.14.4	Diagramme de séquence détaillé du cas d'utilisation ”Comparaison des Consommations prédite ” . . . . .	66
3.15	Diagramme de navigation . . . . .	67
3.16	Diagrammes de Classe . . . . .	69
3.16.1	Passage Relationnel . . . . .	70
3.17	Conclusion . . . . .	70
<b>4</b>	<b>Réalisation</b> . . . . .	<b>71</b>
4.1	Introduction . . . . .	71
4.2	Environnement de développement de l'application . . . . .	71
4.2.1	Architectures . . . . .	71
4.2.2	Matériel . . . . .	72
4.2.3	Outils et logiciels utilisés . . . . .	72
4.2.4	Langages utilisés . . . . .	73
4.2.5	Frameworks . . . . .	73
4.2.6	Librairies Python utilisé . . . . .	74
4.2.7	Modèle utilisé . . . . .	75
4.3	Modélisation des méthodes de prédiction utilisé . . . . .	75

4.3.1	Méthode LSTM . . . . .	75
4.3.2	Méthode régression linéaire : . . . . .	76
4.3.3	Méthode transformers . . . . .	78
4.4	Présentation de l'application . . . . .	78
4.4.1	Application Web . . . . .	78
4.4.2	Interfaces Graphiques . . . . .	79
4.4.2.1	Interface d'Authentification . . . . .	79
4.4.2.2	Interface Inscription . . . . .	80
4.4.2.3	Interface du Tableau de Bord . . . . .	80
4.4.2.4	Interface de Visualisation Générale des Consommations . . . . .	81
4.4.2.5	Interface Visualisation des Consommations par Client . . . . .	82
4.4.2.6	Interface Localisation des Clients . . . . .	83
4.4.2.7	Interface Prédiction des Consommations . . . . .	83
4.4.2.8	Interfaces de Comparaison . . . . .	88
4.4.2.9	Interface d'Ajout et de Suppression d'un Client . . . . .	90
4.4.2.10	Interface de Tableau de Bord de Client . . . . .	91
4.4.2.11	Interface de Prédiction pour le Client . . . . .	93
4.5	Comparaison entre les Méthodes de Régression Linéaire, LSTM et Transformers	95
4.6	Conclusion . . . . .	97
	<b>Conclusion et perspectives</b>	<b>98</b>
	<b>Bibliographie</b>	<b>101</b>

# Table des figures

1.1	Série temporelle [21]. . . . .	4
1.2	shéma additif et multiplicatif [10] . . . . .	6
1.3	Représentation graphique des moyens mobiles . . . . .	8
1.4	Représentation graphique de régression linière . . . . .	10
1.5	Représentation graphique de méthode exponentielle . . . . .	11
1.6	Machines à Vecteurs de Support [34] . . . . .	14
1.7	les Forêts Aléatoires [17] . . . . .	15
1.8	La descente de gradient [29]. . . . .	17
1.9	Les réseaux de neurones [12] . . . . .	19
1.10	la Structure des Réseaux de Neurones [12] . . . . .	20
1.11	Cellule Lstm [12] . . . . .	23
2.1	l'architecture de base des Transformers[36] . . . . .	29
2.2	la structure des encodeurs et des décodeurs dans un Transformer[37]. . . . .	30
3.1	Organigramme de la Sonelgaz Direction Distribution de BÉJAIA[5]. . . . .	36
3.2	Diagramme de contexte . . . . .	44
3.3	Diagramme de cas d'utilisation . . . . .	50
3.4	Diagramme de cas d'utilisation "S'Authentifier" . . . . .	53
3.5	Diagramme de cas d'utilisation "visualiser les consommationsr" . . . . .	55
3.6	Diagramme de cas d'utilisation "prédire la consommation" . . . . .	57
3.7	Diagramme de cas d'utilisation "Comparaison de la consommation" . . . . .	59
3.8	Diagramme de cas d'utilisation "prédire la consommation d'un client" . . . . .	61
3.13	Diagramme de Navigation . . . . .	68
3.14	Diagramme de Classe . . . . .	69
3.15	Diagramme de Classe . . . . .	70
4.1	Interface d'Authentification . . . . .	80
4.2	Interface Inscription . . . . .	80
4.3	Interface Tableau de Bord . . . . .	81
4.4	Interface de Visualisation Générale des Consommations Tableau . . . . .	81
4.5	Interface de Visualisation Générale des Consommations Graphe . . . . .	82
4.6	Interface Visualisation des Consommations par Client - Tableau . . . . .	82
4.7	Interface Visualisation des Consommations par Client - Graphe . . . . .	83
4.8	Interface de Visualisation de la Localisation des Grands Clients . . . . .	83
4.9	Interface Prédiction des Consommations . . . . .	84



4.10	Interface Prédiction avec Régression - Tableau . . . . .	84
4.11	Capture d'un Exemple de la Consommation Prédite en Format PDF . . . . .	85
4.12	Interface Prédiction avec Régression Linéaire - Graphe . . . . .	85
4.13	Interface Prédiction avec Transformers - Tableau . . . . .	86
4.14	Interface Prédiction avec Transformers - Graphe . . . . .	86
4.15	Interface Prédiction avec LSTM - Tableau . . . . .	87
4.16	Interface Prédiction avec LSTM - Graphe . . . . .	87
4.17	Interface de Comparaison de Prédiction par Client . . . . .	88
4.18	Interface de Comparaison de Prédiction par Client - Tableau . . . . .	88
4.19	Interface de Comparaison de Prédiction par Client - Graphe . . . . .	89
4.20	Interface d'Ajout d'un Client . . . . .	90
4.21	Interface de Suppression d'un Client . . . . .	90
4.22	Interface de Tableau de Bord de Client - Tableau . . . . .	91
4.23	Interface de Tableau de Bord de Client - Graphe . . . . .	91
4.24	Interface de Tableau de Bord de Client - Consommation durant les 4 ans . . . . .	92
4.25	Interface de Prédiction pour le Client - Tableau . . . . .	93
4.26	Interface de Prédiction avec Régression Linéaire - Graphe . . . . .	93
4.27	Interface de Prédiction avec Transformers - Graphe . . . . .	94
4.28	Interface de Prédiction avec LSTM - Graphe . . . . .	94
4.29	Comparaison des taux d'erreur entre la Régression Linéaire, LSTM et Transformers . . . . .	96

# Liste des tableaux

2.1	Comparaison Synthétique [35]. . . . .	28
3.1	Messages échangés entre le système et les acteurs . . . . .	45
3.2	Product Backlog . . . . .	48
3.3	Planification des Sprints . . . . .	49
3.4	Formalisme de description des cas d'utilisation . . . . .	51
3.5	description des cas d'utilisation "S'authentifier " . . . . .	52
3.6	description des cas d'utilisation " Visualiser les consommations" . . . . .	54
3.7	description des cas d'utilisation " Prédire la consommation " . . . . .	56
3.8	description des cas d'utilisation " Comparaison de la consommation " . . . .	58
3.9	description des cas d'utilisation "prédire la consommation d'un client " . . .	60

# Liste des abréviations

<b>AIC</b>	Akaike Information Criterion.
<b>AR</b>	AutoRégressive.
<b>ARIMA</b>	AutoRegressive Integrated Moving Average.
<b>ARMA</b>	AutoRegressive Moving Average.
<b>ADF</b>	Dickey-Fuller augmenté.
<b>ACF</b>	Fonction Auto-Corrélation.
<b>BDD</b>	Base de données.
<b>CSS</b>	Cascading Style Sheets.
<b>CVS</b>	Série Corrigée de Variations Saisonnières.
<b>DL</b>	Deep Learning.
<b>RNN</b>	Recurrent Neural Networks.
<b>DNN</b>	Deep Neural Network.
<b>CNN</b>	Convolutional Neural Networks.
<b>GRU</b>	Gated Recurrent Unit .
<b>IA</b>	Intelligence Artificielle.
<b>LSTM</b>	Long Short Term Memory.
<b>ML</b>	Machine Learning.
<b>PACF</b>	Fonction Auto-Corrélation Partielle.
<b>RAD</b>	Rapide Application Development.
<b>SC</b>	Série Chronologique.
<b>SVM</b>	Support Vector Machine.
<b>UML</b>	Unified Modeling Language.
<b>XP</b>	Exterme Programming.

# Introduction générale

L'anticipation de la consommation électrique revêt une importance cruciale de nos jours, tant d'un point de vue économique qu'environnemental. Étant donné que l'électricité ne peut être stockée, il est essentiel de répondre à la demande actuelle en ajustant la production de manière proportionnelle. Cela permet d'atteindre un équilibre entre la production et la consommation, réduisant ainsi les pertes et assurant un approvisionnement continu pour les consommateurs. Grâce à l'utilisation d'outils de prédiction, basés sur des méthodes telles que la régression linéaire, les Transformers et les LSTM (Long Short Term Memory) du domaine de l'apprentissage profond, il est possible de prévoir avec précision les besoins énergétiques pour la production.

Notre projet vise à développer un système novateur pour la Sonelgaz Distribution Direction de BEJAIA, dédié à l'analyse et à la prévision des ventes d'électricité, afin d'améliorer l'efficacité du processus d'analyse des statistiques de ventes et de prédire leur consommation électrique. En exploitant des méthodes telles que la régression linéaire, les Transformers et les LSTM dans le domaine de l'apprentissage profond, notre objectif est d'offrir une solution automatisée et plus efficace pour répondre aux besoins de notre client. Pour la gestion de notre projet, nous avons opté pour une méthodologie agile afin d'assurer une gestion efficace et flexible. Notre travail est divisé en quatre chapitres, organisés de la manière suivante :

**Le premier chapitre :** Ce chapitre offre un aperçu des technologies utilisées dans notre projet. Nous explorons la régression linéaire et d'autres méthodes classiques d'apprentissage, ainsi que les techniques de deep learning telles que les Transformers et les LSTM (Long Short Term Memory). En outre, nous discutons de l'analyse et de la prévision des séries temporelles. Nous détaillons également le processus d'analyse prédictive et introduisons la gestion de projet, en mettant particulièrement l'accent sur les méthodes agiles, notamment la méthode Scrum qui sera mise en œuvre dans notre projet.

**Le deuxième chapitre :** Nous plongeons dans l'historique des méthodes Transformers, mettant en lumière leur évolution et leur importance croissante dans le domaine de l'apprentissage automatique. Nous explorons également en détail comment utiliser efficacement les Transformers dans notre contexte spécifique, en mettant en avant les bibliothèques, telles que TensorFlow et PyTorch, pour implémenter ces méthodes de manière pratique et performante dans l'analyse et la prévision des séries temporelles.

**Le troisième et le dernier chapitre :** Nous examinons en détail les étapes de construction des trois modèles de prédiction. Nous décrivons également l'architecture de

l'application et expliquons son fonctionnement à travers les interfaces graphiques.

En conclusion, nous résumons les principaux éléments discutés dans ce rapport et proposons des pistes pour de futures recherches et développements.

# Chapitre 1

## ETAT DE L'ART

### 1.1 Introduction

L'Intelligence Artificielle (IA) est essentielle pour l'analyse et la prévision des séries temporelles, nécessitant une maîtrise de ses fondamentaux et une compréhension des diverses approches pour minimiser les risques associés. Dans le domaine de l'électricité, le machine learning et le deep learning jouent un rôle crucial dans la prévision de la demande et l'optimisation de la production d'énergie, contribuant ainsi à la croissance économique. Ce chapitre examine les technologies employées dans notre projet d'IA, incluant le machine learning, le deep learning, l'analyse des séries temporelles.

### 1.2 Séries Temporelles

#### 1.2.1 Définition des Séries Temporelles

Une série temporelle est une séquence de données observées à différents points dans le temps, généralement à intervalles réguliers. Elle est souvent utilisée pour analyser et modéliser des phénomènes qui évoluent au fil du temps, tels que les prix des actions, les températures météorologiques, les ventes de produits, etc[6].

La Figure 1.1 représente un exemple d'une série temporelle

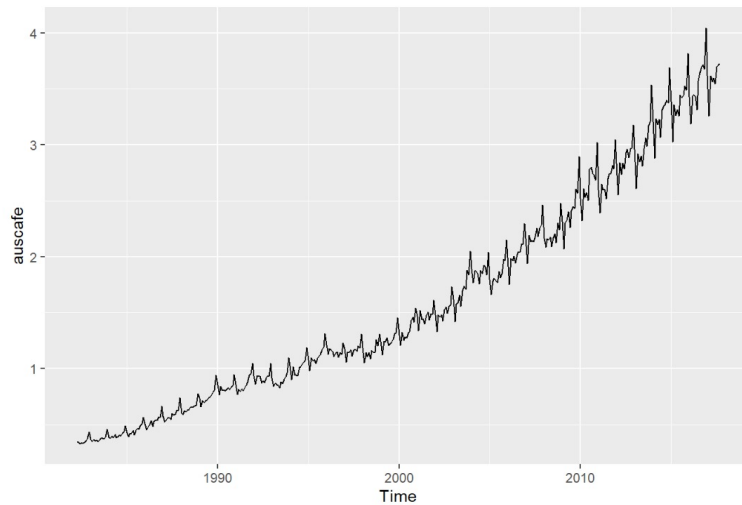


FIGURE 1.1 – Série temporelle [21].

Mathématiquement, une série temporelle peut être représentée par une fonction  $X(t)$ , où  $t$  est le temps. Si nous avons  $n$  observations, la série temporelle peut être représentée comme suit :

$$X(t_1), X(t_2), X(t_3), \dots, X(t_n) \quad (1.1)$$

Où  $t_1, t_2, t_3, \dots, t_n$  sont les instants auxquels les observations ont été faites.

## 1.2.2 Décomposition des Séries Temporelles

La décomposition d'une série temporelle est un processus analytique qui consiste à séparer une série temporelle observée en différentes composantes afin d'identifier les tendances, les saisons et les irrégularités qui la composent. Cette décomposition est souvent utilisée pour mieux comprendre la structure sous-jacente des données temporelles et pour faciliter la modélisation et la prévision [10].

Une décomposition typique d'une série temporelle peut être réalisée en trois composantes principales :

**1. Tendance (T) :** La composante de tendance représente la direction générale du comportement de la série temporelle sur une période de temps plus longue. Elle peut être croissante, décroissante ou stable. La tendance capture les variations à long terme de la série temporelle, souvent en utilisant des techniques telles que la moyenne mobile ou la régression [10].

**2. Saisonnalité (S) :** La composante saisonnière représente les fluctuations périodiques ou récurrentes de la série temporelle, généralement associées à des effets saisonniers, tels que les variations saisonnières des ventes de produits, les fluctuations météorologiques saisonnières, etc. Cette composante capture les variations régulières et prévisibles qui se répètent à des intervalles fixes [10].

**3. Irrégularité ou résidu (I) :** La composante d'irrégularité, également appelée résidu, représente les variations aléatoires ou irrégulières de la série temporelle qui ne peuvent pas être attribuées à la tendance ou à la saisonnalité. Elle capture les fluctuations non systématiques ou les chocs aléatoires qui peuvent affecter la série temporelle [10].

Mathématiquement, la décomposition d'une série temporelle  $Y(t)$  peut être représentée comme suit :

$$Y(t) = T(t) + S(t) + I(t) \quad (1.2)$$

Où : .

- $Y(t)$  est la série temporelle observée à un moment donné  $t$ .
- $T(t)$  est la composante de tendance.
- $S(t)$  est la composante saisonnière .
- $I(t)$  est la composante d'irrégularité ou résiduelle.

Une fois que la série temporelle a été décomposée en ces différentes composantes, chaque composante peut être analysée séparément, ce qui peut faciliter la modélisation et la prévision à des fins spécifiques [10] .

## 1.3 Modèles de Décomposition d'une série temporelle

### 1.3.1 Modèle de Décomposition Additive

La décomposition additive d'une série temporelle suppose que la série est la somme de ses composantes de tendance, saisonnière et irrégulière. Voici sa représentation mathématique :

$$Y(t) = T(t) + S(t) + I(t) \quad (1.3)$$

où : .

- $Y(t)$  est la série temporelle observée à un moment  $t$  . .
- $T(t)$  est la composante de tendance. .
- $S(t)$  est la composante saisonnière. .
- $I(t)$  est la composante d'irrégularité ou résiduelle. .

Dans ce modèle, les variations saisonnières sont considérées comme constantes, quelle que soit la valeur de la tendance.

### 1.3.2 Modèle de Décomposition Multiplicative

La décomposition multiplicative d'une série temporelle suppose que la série est le produit de ses composantes de tendance, saisonnière et irrégulière[10] . Voici sa représentation mathématique :

$$Y(t) = T(t) \times S(t) \times I(t) \quad (1.4)$$



où les termes sont définis de la même manière que dans le modèle additif .

Dans ce modèle, les variations saisonnières augmentent ou diminuent en fonction de la tendance de la série temporelle. Si la tendance est à la hausse, les variations saisonnières auront tendance à augmenter également, et vice versa [10] .

Ces modèles sont utilisés pour décomposer une série temporelle en ses différentes composantes afin d'analyser et de modéliser ses variations.

Figure 1.2 présente un exemple de schéma additif et multiplicatif.

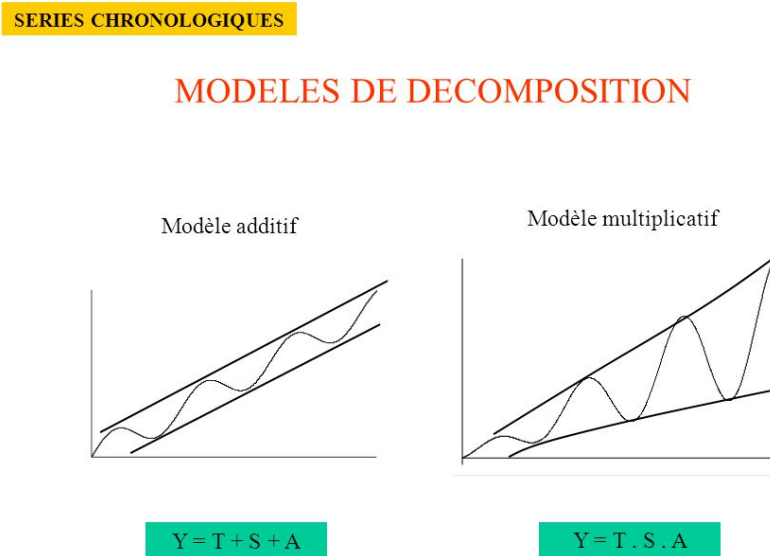


FIGURE 1.2 – schéma additif et multiplicatif [10] .

## 1.4 Méthodes de prévision

### 1.4.1 Moyenne Mobile (Moving Average)

#### 1.4.1.1 Définition

La méthode de la moyenne mobile est une technique simple de prévision qui consiste à calculer la moyenne des valeurs passées d'une série temporelle sur une période spécifiée, puis à utiliser cette moyenne pour prédire les valeurs futures. Cette méthode est souvent utilisée pour modéliser les variations à court terme dans les données temporelles [11].

Soit  $y_t$  la valeur de la série temporelle à l'instant  $t$ , et  $k$  la période de la moyenne mobile. La prédiction de la moyenne mobile  $y_{t+1}$  pour le prochain instant de temps  $t + 1$  est calculée comme suit :

$$y_{t+1} = \frac{1}{k} \sum_{i=t-k+1}^t y_i \quad (1.5)$$

### 1.4.1.2 Types de Moyennes Mobiles

1. **Moyenne Mobile Simple (SMA)** : Dans la SMA, chaque observation est pondérée uniformément, ce qui signifie que toutes les observations ont le même poids dans le calcul de la moyenne.
2. **Moyenne Mobile Pondérée (WMA)** : Dans la WMA, différentes pondérations sont attribuées à chaque observation, généralement avec des pondérations décroissantes à mesure que les observations s'éloignent dans le temps [11].

La formule de la moyenne mobile pondérée est calculée en attribuant des poids à chaque observation :

$$\text{WMA}(t) = \sum_{i=t-k+1}^t w_i \cdot y_i \quad (1.6)$$

- $\text{WMA}(t)$  représente la moyenne mobile pondérée à l'instant  $t$ .
- $w_i$  sont les poids attribués à chaque observation.
- $y_i$  sont les valeurs de la série temporelle pour les instants  $t - k + 1$  à  $t$ .

Les poids  $w_i$  peuvent être attribués de différentes manières, par exemple avec une pondération décroissante à mesure que les observations s'éloignent dans le temps.

3. **Moyenne Mobile Exponentielle (EMA)** : L'EMA accorde davantage de poids aux observations les plus récentes, ce qui les rend plus réactives aux changements récents dans les données [11].

La formule de la moyenne mobile exponentielle est calculée en attribuant davantage de poids aux observations les plus récentes :

$$\text{EMA}(t) = \alpha \cdot y_t + (1 - \alpha) \cdot \text{EMA}(t - 1) \quad (1.7)$$

- $\text{EMA}(t)$  représente la moyenne mobile exponentielle à l'instant  $t$ .
- $\alpha$  est le facteur de lissage, généralement compris entre 0 et 1.
- $y_t$  est la valeur de la série temporelle à l'instant  $t$ .
- $\text{EMA}(t - 1)$  est la moyenne mobile exponentielle calculée à l'instant précédent.

La moyenne mobile exponentielle est mise à jour à chaque nouvelle observation en combinant la valeur actuelle avec la moyenne mobile exponentielle précédente, avec un poids attribué à chaque observation.

### 1.4.1.3 Avantages de la Moyenne Mobile

La méthode de la moyenne mobile est simple à calculer et à utiliser, ce qui la rend accessible même aux utilisateurs novices en analyse de séries temporelles[27].

- Fréquences de variations dans les données, ce qui la rend adaptable à diverses situations.
- En prenant une moyenne sur plusieurs observations, la méthode de la moyenne mobile peut atténuer les effets de données bruitées ou erratiques.

#### 1.4.1.4 Limitations de la Moyenne Mobile

- Retard dans la détection des changements de tendance .
- Le choix de la période de la moyenne mobile peut influencer considérablement les prévisions. Une période trop courte peut entraîner une instabilité, tandis qu'une période trop longue peut entraîner un lissage excessif des données.

#### 1.4.1.5 Représentation graphique des moyens mobiles

La représentation graphique de la méthode de la moyenne mobile peut être réalisée à l'aide d'un graphique de la série temporelle d'origine ainsi que d'une courbe représentant la moyenne mobile.

La représentation graphique de la méthode de la moyenne mobile peut être réalisée à l'aide d'un graphique de la série temporelle d'origine ainsi que d'une courbe représentant la moyenne mobile. Comme illustré dans la Figure 1.3, cette figure montre une série temporelle aléatoire avec la moyenne mobile calculée sur une fenêtre de taille 5 [28].

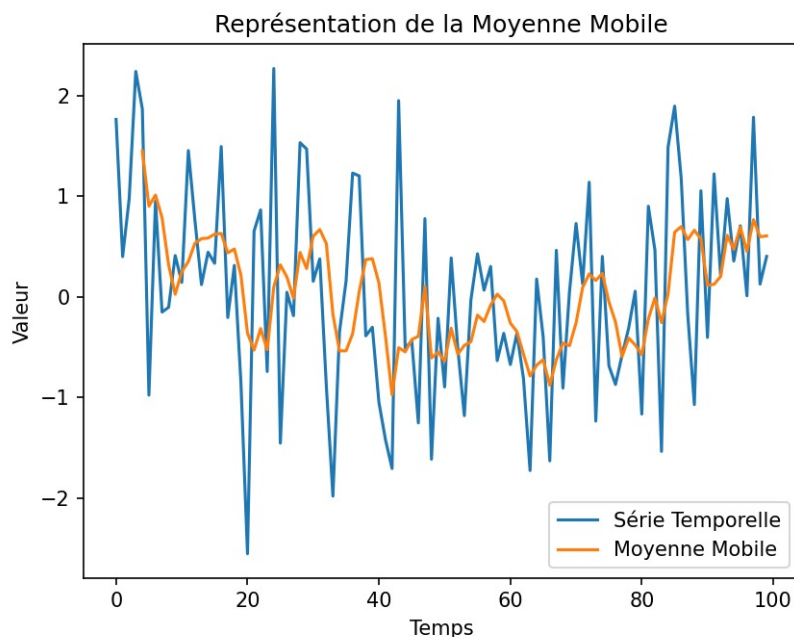


FIGURE 1.3 – Représentation graphique des moyens mobiles

La méthode de la moyenne mobile est un outil utile pour la prévision des séries temporelles, en particulier pour modéliser les variations à court terme. Cependant, elle doit être utilisée avec prudence et en tenant compte de ses limitations, notamment en ce qui concerne le choix de la période et la détection des changements de tendance [27] .

### 1.4.2 Régression Linéaire

#### 1.4.2.1 Définition

La régression linéaire est une méthode statistique qui consiste à ajuster une ligne droite aux données historiques de la série temporelle afin de modéliser la tendance. Cette méthode

est utile lorsque la série temporelle présente une tendance linéaire claire [11]. Le modèle de régression linéaire peut être représenté mathématiquement par l'équation d'une droite :

$$y_t = \beta_0 + \beta_1 \cdot t + \varepsilon_t \quad (1.8)$$

- $y_t$  représente la valeur de la série temporelle à l'instant  $t$ . .
- $\beta_0$  et  $\beta_1$  sont les coefficients de régression, où  $\beta_0$  est l'ordonnée à l'origine et  $\beta_1$  est la pente de la droite. .
- $t$  est la variable explicative représentant le temps. .
- $\varepsilon_t$  est le terme d'erreur aléatoire. .

Les coefficients de régression  $\beta_0$  et  $\beta_1$  sont estimés à partir des données historiques à l'aide de la méthode des moindres carrés ordinaires (MCO), qui minimise la somme des carrés des écarts entre les valeurs observées et les valeurs prédites par le modèle.[28]

#### 1.4.2.2 Avantages de la Régression Linéaire

- La régression linéaire fournit une relation linéaire claire entre la variable explicative (temps) et la variable cible (série temporelle), ce qui permet une interprétation intuitive des résultats
- Adaptabilité aux tendances linéaires

#### 1.4.2.3 Limitations de la Régression Linéaire

- Sensible aux anomalies
- Incapacité à capturer des relations complexes

#### 1.4.2.4 Représentation graphique de régression linéaire

Cette figure présente une série temporelle fictive comprenant une tendance linéaire et du bruit, sur laquelle un modèle de régression linéaire est appliqué pour ajuster une ligne droite aux données. Comme illustré dans la Figure 1.4, cette représentation inclut à la fois les données d'origine et la ligne de régression tracées sur un graphique.

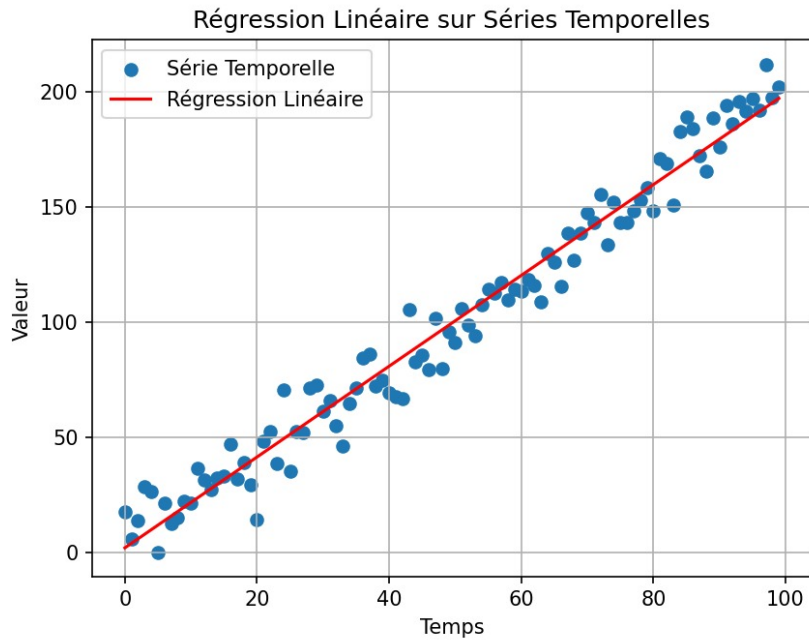


FIGURE 1.4 – Représentation graphique de régression linière

La régression linéaire est une méthode utile pour modéliser et prévoir les tendances linéaires dans les séries temporelles. Elle offre une approche interprétable et facile à utiliser pour l'analyse de données temporelles, bien qu'elle puisse être limitée dans sa capacité à capturer des relations plus complexes.

### 1.4.3 Méthode Exponentielle

#### 1.4.3.1 Définition

Les méthodes exponentielles sont des techniques de prévision qui attribuent un poids décroissant aux observations passées en fonction de leur âge. Elles sont particulièrement adaptées pour modéliser les tendances et les variations saisonnières dans les données temporelles. L'EMA est l'une des méthodes exponentielles les plus couramment utilisées. Elle calcule une moyenne pondérée des valeurs passées, en accordant davantage de poids aux observations les plus récentes[11]. L'EMA est définie récursivement comme suit :

$$\text{EMA}(t) = \alpha \cdot y_t + (1 - \alpha) \cdot \text{EMA}(t - 1) \quad (1.9)$$

- $\text{EMA}(t)$  représente la moyenne mobile exponentielle à l'instant  $t$  .
- $\alpha$  est le facteur de lissage, généralement compris entre 0 et 1 .
- $y_t$  est la valeur de la série temporelle à l'instant  $t$  .
- $\text{EMA}(t1)$  est la moyenne mobile exponentielle calculée à l'instant précédent .

Le facteur  $\alpha$  de lissage contrôle la réactivité de l'EMA aux changements dans les données. Une valeur plus élevée de  $\alpha$  donne plus de poids aux observations récentes, rendant l'EMA plus réactive aux changements récents, mais aussi plus volatile.

### 1.4.3.2 Avantages des Méthodes Exponentielles

- Adaptabilité aux variations temporaires
- Facile d'interprétation

### 1.4.3.3 Limitations des Méthodes Exponentielles

- Sensible au choix du facteur de lissage
- Incapacité à capturer des motifs complexes

### 1.4.3.4 Représentation graphique de méthode exponentielle

Cette figure présente une série temporelle fictive ainsi que les valeurs calculées en utilisant la formule du lissage exponentiel. Comme illustré dans la Figure 1.5, cette représentation inclut à la fois les données d'origine et les données lissées tracées sur un graphique .

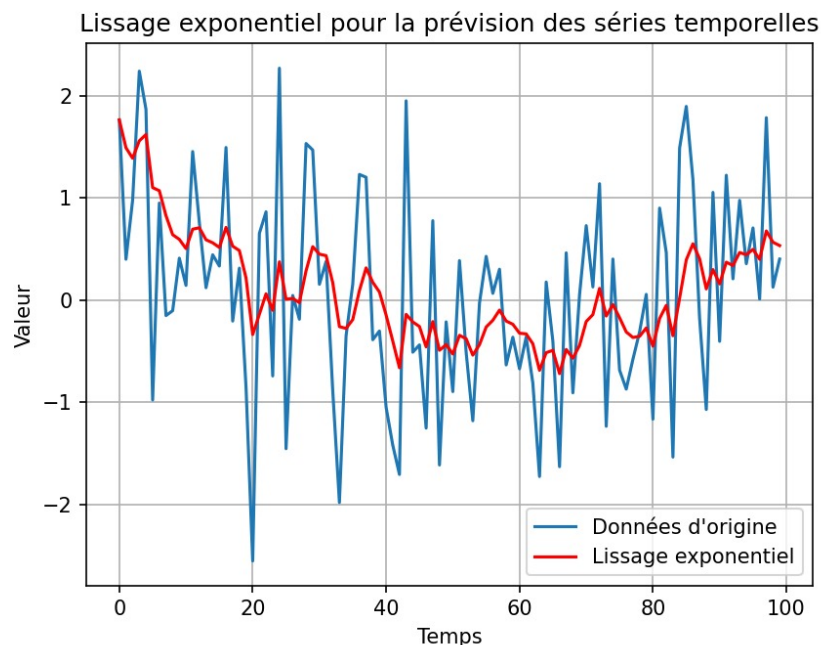


FIGURE 1.5 – Représentation graphique de méthode exponentielle

Les méthodes exponentielles offrent une approche efficace pour modéliser et prévoir les séries temporelles, en accordant une attention particulière aux variations récentes dans les données. Elles sont largement utilisées dans de nombreux domaines pour leur simplicité et leur capacité à fournir des prévisions rapides et réactives.

## 1.4.4 ARIMA (AutoRegressive Integrated Moving Average)

### 1.4.4.1 Définition

ARIMA est un modèle statistique couramment utilisé pour modéliser les séries temporelles. Il combine les concepts de régression autorégressive (AR), de moyenne mobile intégrée

(I) et de moyenne mobile (MA) pour capturer les différentes composantes des séries temporelles, y compris la tendance, la saisonnalité et les fluctuations aléatoires [11].

#### 1.4.4.2 Composantes du Modèle ARIMA

- **Composante Auto-Régressive (AR)** : La composante AR capture la dépendance linéaire entre les observations successives dans la série temporelle. Un modèle AR d'ordre  $p$  utilise  $p$  observations précédentes pour prédire la valeur actuelle [11].
- **Composante Intégrée (I)** : La composante I représente le nombre de différenciations nécessaires pour rendre la série temporelle stationnaire, c'est-à-dire en éliminant les tendances et les cycles. Cette étape est cruciale pour appliquer les modèles ARIMA [11].
- **Composante Moyenne Mobile (MA)** : La composante MA modélise les erreurs de prévision en utilisant une moyenne mobile des erreurs précédentes. Un modèle MA d'ordre  $q$  utilise  $q$  erreurs précédentes pour prédire la valeur actuelle [11].

Formule Mathématique du Modèle ARIMA :

$$y_t = c + \phi_1 y_{t-1} + \phi_2 y_{t-2} + \dots + \phi_p y_{t-p} + \theta_1 \varepsilon_{t-1} + \theta_2 \varepsilon_{t-2} + \dots + \theta_q \varepsilon_{t-q} + \varepsilon_t \quad (1.10)$$

- $y_t$  représente la valeur observée de la série temporelle à l'instant  $t$ .
- $c$  est la constante du modèle.
- $\phi_1, \phi_2, \dots, \phi_p$  sont les coefficients de régression auto-régressive.
- $\theta_1, \theta_2, \dots, \theta_q$  sont les coefficients de régression de la moyenne mobile.
- $\varepsilon_t$  est le terme d'erreur aléatoire à l'instant  $t$ .

#### 1.4.4.3 Avantages de ARIMA

- ARIMA est flexible car il peut modéliser efficacement une large gamme de comportements temporels[11].
- Ce modèle est robuste face aux données manquantes et aux valeurs aberrantes, grâce à ses méthodes d'ajustement qui permettent de maintenir la précision malgré les perturbations dans les séries temporelles [11].
- Les modèles ARIMA sont facilement interprétables en raison de leur formulation intuitive basée sur l'autorégression [11].

#### 1.4.5 Limitations de ARIMA

- ARIMA peut devenir complexe à ajuster et à interpréter lorsque les séries temporelles présentent des modèles non linéaires ou des structures complexes[11].
- Ce modèle est sensible au choix des paramètres ( $p, d, q$ ) qui définissent l'ordre de l'autorégression, l'intégration et la moyenne mobile [11].

ARIMA est une méthode puissante et flexible pour modéliser et prévoir les séries temporelles, en capturant efficacement les tendances, les saisons et les fluctuations aléatoires dans les données. Son utilisation nécessite cependant une expertise dans l'identification des composantes ARIMA appropriées pour chaque série temporelle spécifique.

## 1.4.6 SARIMA (Seasonal AutoRegressive Integrated Moving Average)

### 1.4.6.1 Definition

SARIMA est un modèle de séries temporelles capable de modéliser à la fois les tendances et les saisonnalités, en plus des composantes autorégressives et à moyenne mobile. Il est largement utilisé pour la prévision des séries temporelles avec des motifs saisonniers clairs.[11] SARIMA est spécifié par trois termes principaux :  $(p,d,q)$  pour les composantes non saisonnières et  $(P,D,Q,m)$  pour les composantes saisonnières, où :

- $p$  est l'ordre du terme autoregressif non saisonnier.
- $d$  est le degré de différenciation non saisonnier.
- $q$  est l'ordre du terme de moyenne mobile non saisonnier.
- $P$  est l'ordre du terme autoregressif saisonnier.
- $D$  est le degré de différenciation saisonnier.
- $Q$  est l'ordre du terme de moyenne mobile saisonnier.
- $m$  est la période saisonnière.

La structure générale d'un modèle SARIMA est donc :

$$\text{SARIMA}(p, d, q)(P, D, Q, m) \quad (1.11)$$

### 1.4.6.2 Avantages de SARIMA

- Modélisation des tendances et des saisons
- Adaptabilité aux données saisonnières
- Flexibilité dans la spécification du modèle

### 1.4.6.3 Limitations de SARIMA

- Complexité de la modélisation
- Sensibilité aux données

SARIMA est une méthode puissante et largement utilisée pour la prévision des séries temporelles saisonnières. En spécifiant correctement les termes non saisonniers et saisonniers, SARIMA peut fournir des prévisions précises et fiables pour une grande variété de données temporelles.



## 1.5 Modèles du Machine Learning

En plus des méthodes classiques telles que ARIMA, il existe toute une gamme de techniques basées sur l'apprentissage automatique pour la prévision des séries temporelles. Ces méthodes exploitent des algorithmes sophistiqués pour modéliser et prévoir les tendances et les variations dans les données temporelles. [26]. Voici quelques-uns des modèles les plus populaires :

### — Machines à Vecteurs de Support (SVM)

Les machines à vecteurs de support sont des algorithmes d'apprentissage supervisé utilisés pour la classification et la régression. Elles peuvent également être appliquées à la prévision des séries temporelles en formulant le problème comme une tâche de régression, où les valeurs passées sont utilisées pour prédire les valeurs futures.

Les SVM visent à trouver un hyperplan dans un espace de dimension supérieure qui sépare de manière optimale les différentes classes. Pour les problèmes de classification binaire, cet hyperplan est choisi de manière à maximiser la marge, qui est la distance entre l'hyperplan et les points de données les plus proches de chaque classe, appelés vecteurs de support. Comme la Figure 1.7 le montre, cet hyperplan permet une séparation claire entre les classes [34]. Les SVM peuvent être linéaires ou non

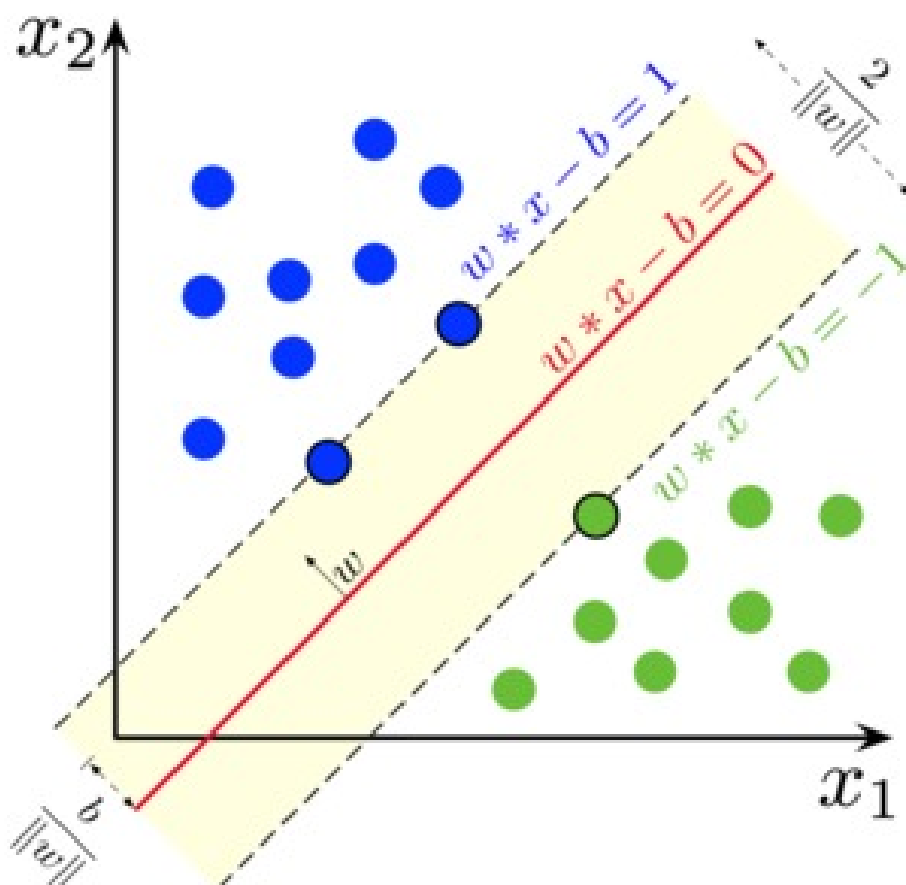


FIGURE 1.6 – Machines à Vecteurs de Support [34]

linéaires. Dans les cas où les données ne sont pas linéairement séparables, les SVM utilisent des fonctions noyau pour projeter les données dans un espace de dimension supérieure où elles peuvent être séparées linéairement.

## — Forêts Aléatoires (Random Forests)

Les Forêts Aléatoires (Random Forests) sont une méthode d'apprentissage supervisé utilisée pour la classification, la régression et d'autres tâches d'apprentissage automatique. Elles sont largement utilisées dans divers domaines tels que la biomédecine, la finance, la bioinformatique, la vision par ordinateur et bien d'autres [17]. Voici quelques informations clés sur les Forêts Aléatoires : comme illustré dans l'exemple de la Figure 1.8 ci-dessous.

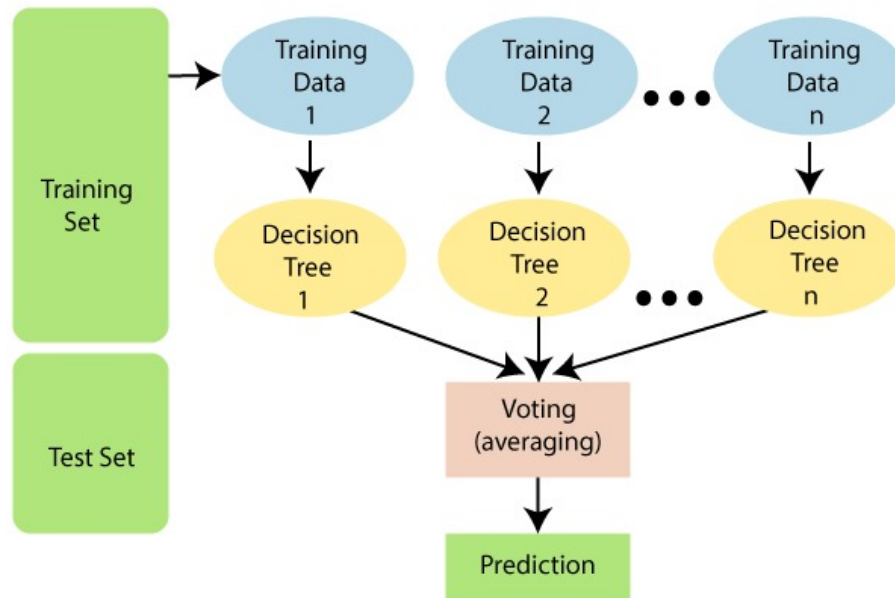


FIGURE 1.7 – les Forêts Aléatoires [17]

Une Forêt Aléatoire est un ensemble d'arbres de décision, où chaque arbre est entraîné sur un sous-ensemble aléatoire des données d'entraînement et des caractéristiques (variables explicatives).

Pendant l'apprentissage, chaque arbre est formé de manière indépendante, généralement en utilisant une technique de bagging (bootstrap aggregating), ce qui signifie qu'il est entraîné sur un échantillon aléatoire avec remplacement des données d'entraînement. Lors de la prédiction, la Forêt Aléatoire agrège les prédictions de chaque arbre pour obtenir une prédiction finale. Pour la classification, la prédiction est généralement basée sur un vote majoritaire, tandis que pour la régression, la prédiction est généralement la moyenne des prédictions de chaque arbre [17].

Bien sûr ! Voici une expansion de chaque point avec une phrase pour chaque avantage :

### 1.5.1 Avantages des Méthodes basées sur l'apprentissage automatique

- **La capacité à capturer des relations complexes** : Les méthodes d'apprentissage automatique peuvent identifier et modéliser des relations non linéaires et des interactions complexes entre les variables, souvent indétectables par des techniques traditionnelles [17].
- **La performance élevée** : Ces méthodes peuvent fournir des prédictions très précises

et fiables grâce à leur capacité à apprendre et à s'améliorer à partir de grandes quantités de données, dépassant souvent les approches classiques en termes de précision et de généralisation [17].

- **La flexibilité** : Les modèles d'apprentissage automatique peuvent être appliqués à une large gamme de problèmes et de types de données, qu'il s'agisse de données structurées, non structurées, textuelles ou visuelles, offrant ainsi une grande polyvalence dans leur application [17].

## 1.5.2 Limitations des Méthodes basées sur l'apprentissage automatique

:

- La complexité de l'entraînement et de l'interprétation
- Les Exigences en matière de données
- La sensibilité aux hyperparamètres

Les méthodes basées sur l'apprentissage automatique offrent une approche puissante et flexible pour la prévision des séries temporelles, en particulier dans les cas où les données présentent des structures complexes ou non linéaires. Elles sont largement utilisées dans de nombreux domaines pour leur capacité à capturer des relations subtiles dans les données et à fournir des prévisions précises et fiables[17].

## 1.6 Le Deep Learning

### 1.6.1 Définition

Le Deep Learning, ou apprentissage profond, est une branche de l'intelligence artificielle qui utilise des réseaux de neurones artificiels profonds pour apprendre et extraire des modèles complexes à partir de données [29]. Dans le domaine des séries temporelles, le Deep Learning peut être défini comme une méthode d'analyse et de modélisation des données temporelles qui exploite des architectures de réseaux de neurones profonds pour découvrir des motifs et des structures temporelles complexes..

### 1.6.2 Algorithmes d'optimisation de Deep Learning

L'optimisation joue un rôle central dans l'apprentissage automatique, car la performance d'un modèle dépend directement de la qualité des paramètres qu'il apprend à partir des données d'entraînement. En minimisant la fonction de coût [32]. Voici quelques-uns des algorithmes d'optimisation les plus couramment utilisés dans ce contexte :

- **La descente de gradient**

Tire son nom du concept mathématique du gradient, qui représente la direction de la plus forte augmentation d'une fonction. Dans le contexte de l'apprentissage automatique, la fonction à minimiser est souvent une fonction de coût qui mesure l'écart entre les prédictions du modèle et les véritables valeurs cibles. L'objectif est de trouver les

valeurs optimales des paramètres du modèle qui minimisent cette fonction de coût. Comme illustré dans la Figure 1.6, la descente de gradient, avec ses variantes telles que le Gradient Descent classique, le Stochastic Gradient Descent et le Mini-Batch Gradient Descent, constitue une famille d'algorithmes fondamentaux en optimisation et en apprentissage automatique [32].

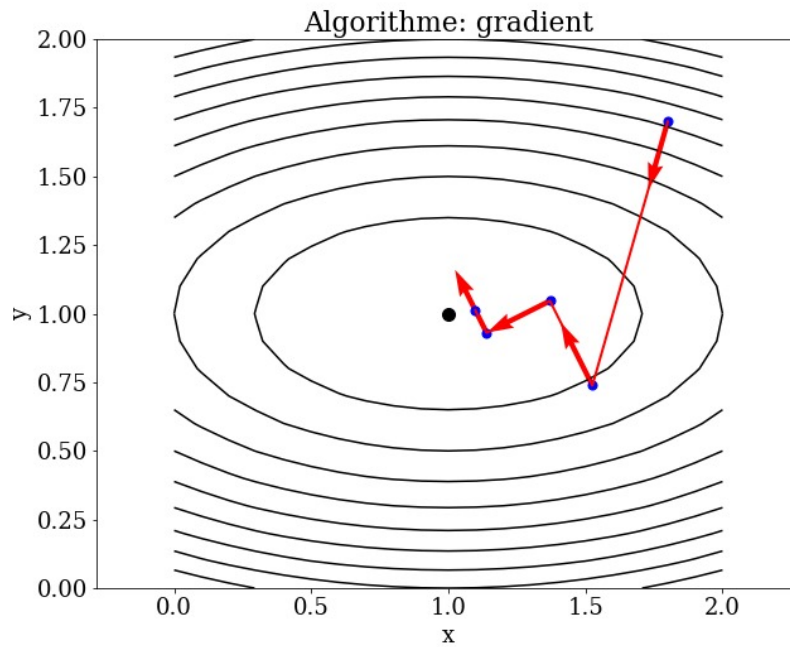


FIGURE 1.8 – La descente de gradient [29].

## 1. Gradient Descent classique .

Le Gradient Descent classique est l'un des algorithmes d'optimisation les plus fondamentaux et largement utilisés dans le domaine de l'apprentissage automatique et de l'optimisation. Son objectif principal est de minimiser une fonction de coût en ajustant itérativement les paramètres du modèle [32]. Formule mathématique La mise à jour des paramètres  $\theta$  à chaque itération  $t$  est formulée comme suit :

$$\theta(t + 1) = \theta(t) - \alpha \nabla J(\theta(t)) \quad (1.12)$$

Où :

- $\theta(t)$  : représente le vecteur des paramètres à l'itération  $t$ .
- $\alpha$  : est le taux d'apprentissage, contrôlant la taille des pas de mise à jour.
- $\nabla J(\theta(t))$  : est le gradient de la fonction de coût  $J(\theta)$  par rapport aux paramètres  $\theta$  à l'itération  $t$ .
- **Avantages**
  - Facile à comprendre et à mettre en oeuvre [32].
  - Fonctionne bien sur de nombreux types de fonctions de coût.
  - Peut être utilisé avec de grands ensembles de données et des modèles complexes[32].
- **Inconvénients**
  - Sensible au choix du taux d'apprentissage[32].
  - Requiert des données entièrement chargées [32].

Le Gradient Descent classique est un pilier fondamental de l'optimisation et forme la base de nombreux autres algorithmes d'optimisation, y compris ses variantes :

### a) Le Stochastic Gradient Descent (SGD) .

Le Stochastic Gradient Descent (SGD) est couramment utilisée dans l'apprentissage automatique, en particulier pour l'entraînement de modèles sur de grands ensembles de données. Contrairement au Gradient Descent classique, qui calcule le gradient de la fonction de coût sur l'ensemble des données d'entraînement à chaque itération, le SGD calcule le gradient et met à jour les paramètres du modèle en utilisant un seul exemple d'entraînement (ou un petit lot, appelé mini-batch) à la fois. [31, 32]

- **Avantages**
  - Le SGD est souvent plus rapide que le Gradient Descent classique car il effectue des mises à jour de paramètres plus fréquentes[32].
  - Le SGD est adaptatif aux données et peut être utilisé efficacement sur de grands ensembles de données[32].
- **Inconvénients**
  - Les mises à jour basées sur un seul exemple d'entraînement peuvent être plus bruitées et moins stables que celles basées sur l'ensemble des données.
  - Le SGD peut converger vers un minimum local plutôt que global, ce qui peut parfois entraîner une solution sous-optimale.

### b) Le Mini-Batch Gradient Descent (MBGD) .

Le Mini-Batch Gradient Descent (MBGD) est une autre variante du Gradient Descent classique, qui se situe entre le Gradient Descent classique et le Stochastic Gradient Descent

(SGD). Contrairement au Gradient Descent classique qui utilise l'ensemble des données d'entraînement pour calculer le gradient à chaque itération, et au SGD qui utilise un seul exemple d'entraînement à chaque itération, le MBGD utilise un petit lot (mini-batch) d'exemples d'entraînement.[31]

— **Avantages**

- Les mises à jour des paramètres sont moins bruitées et plus stables que celles du SGD, car elles sont basées sur un petit lot d'exemples d'entraînement[32].
- Le MBGD est souvent plus rapide que le Gradient Descent classique car il effectue des mises à jour plus fréquentes que ce dernier [32].

— **Inconvénients**

- Requiert du réglage des hyperparamètres
- Le MBGD nécessite de stocker en mémoire plusieurs mini-batches, ce qui peut être coûteux en termes de mémoire pour de grands ensembles de données.

### 1.6.3 Réseaux de neurones

#### 1.6.3.1 Définition

Les réseaux de neurones sont les éléments fondamentaux du deep learning. Ils sont utilisés pour apprendre des représentations complexes à partir de données et sont capables de modéliser des relations non linéaires entre les entrées et les sorties [12]. Comme illustré dans la Figure 1.9, ces réseaux de neurones permettent de capturer et d'exploiter ces relations complexes de manière efficace.

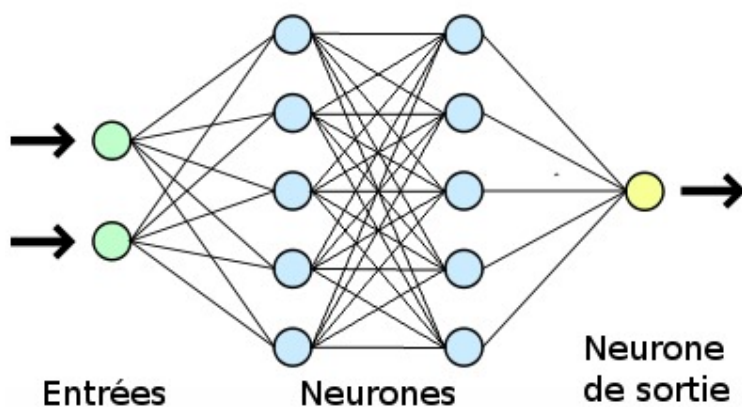


FIGURE 1.9 – Les réseaux de neurones [12]

#### 1.6.3.2 Structure des Réseaux de Neurones

La structure des réseaux de neurones fait référence à l'organisation des neurones et des connexions entre ces neurones dans le réseau. Voici une explication plus détaillée de la structure des réseaux de neurones :

— **Neurones et Couches**

— **Neurones**

Un neurone est une unité de traitement de base d'un réseau de neurones. Chaque

neurone prend des entrées, effectue une opération de calcul sur ces entrées pondérées par des poids, puis applique une fonction d'activation pour produire sa sortie.

— **Couches**

• **Couche d'Entrée**

Cette couche reçoit les entrées brutes du système et transmet ces informations aux couches cachées du réseau.

• **Couches Cachées**

Ces couches sont situées entre la couche d'entrée et la couche de sortie. Elles effectuent des calculs complexes sur les entrées pour apprendre des représentations hiérarchiques des données.

• **Couche de Sortie**

Cette couche produit les sorties finales du réseau. Le nombre de neurones dans cette couche dépend du type de tâche que le réseau doit effectuer (classification, régression, etc.). Comme illustré dans la Figure 1.10, celle-ci représente la structure des Réseaux de Neurones.

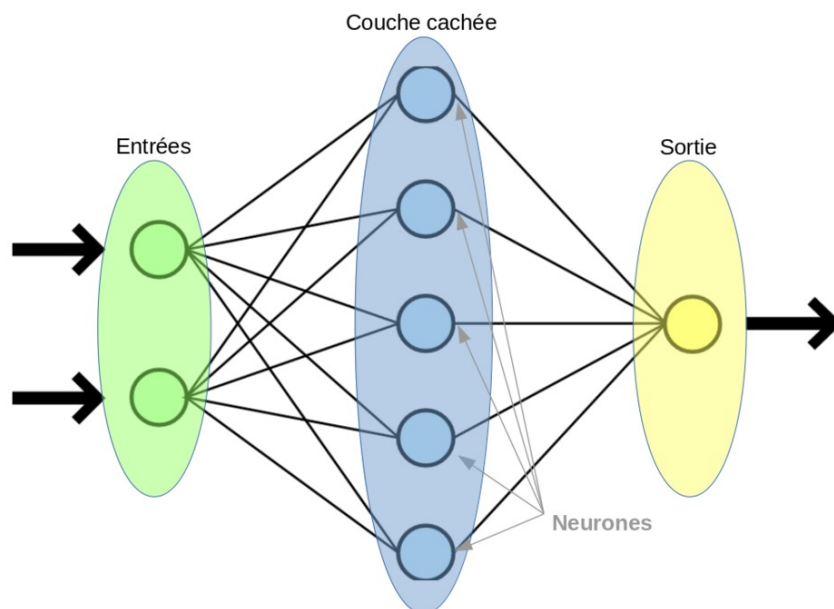


FIGURE 1.10 – la Structure des Réseaux de Neurones [12]

### 1.6.3.3 Les types de réseaux de neurones

Dans le domaine du Deep Learning, plusieurs types de réseaux de neurones sont couramment utilisés en fonction des tâches spécifiques à accomplir [12]. Voici quelques-uns des types de réseaux de neurones les plus utilisés :

#### a) Les Réseaux de Neurones Convolutionnels (CNN)

Les Réseaux de Neurones Convolutionnels (CNN), Les Réseaux de Neurones Convolutionnels (CNN), ou ConvNets, sont des réseaux de neurones profonds qui ont révolutionné la vision par ordinateur et trouvé des applications dans la reconnaissance vocale, la bioinfor-

matique, la finance, et bien d'autres domaines. Leur capacité à capturer des motifs spatiaux à partir d'images les rend particulièrement efficaces pour traiter des données complexes.

Les CNN utilisent des couches de convolution pour appliquer des filtres sur les images d'entrée, extrayant ainsi des caractéristiques pertinentes inspirées du cortex visuel humain. Ces filtres sont appris à partir des données via la rétropropagation du gradient, permettant au réseau de détecter automatiquement des motifs et des caractéristiques discriminatives. Les couches de convolution sont souvent suivies de couches de pooling (comme le max pooling ou l'average pooling) pour réduire la dimensionnalité spatiale tout en conservant les informations essentielles.[12]

Parmi les architectures populaires de CNN, on trouve LeNet, AlexNet, VGG, GoogLeNet (Inception) et ResNet. GoogLeNet a introduit les modules Inception, qui combinent des filtres de différentes tailles pour capturer des caractéristiques à différentes échelles, tandis que ResNet est célèbre pour ses connexions résiduelles qui facilitent l'entraînement de réseaux très profonds en atténuant les problèmes de gradients.

Les applications des CNN sont vastes : ils sont utilisés pour la classification d'images, la détection d'objets, la segmentation sémantique, la reconnaissance faciale, la reconnaissance de texte, la traduction d'images en texte, la synthèse d'images, et bien d'autres tâches. En résumé, les CNN ont révolutionné le traitement d'images en permettant des avancées significatives grâce à leur capacité à apprendre des caractéristiques hiérarchiques à partir des données brutes, et ils continuent d'évoluer pour offrir des applications encore plus puissantes à l'avenir. [12].

### **b) Les Réseaux de Neurones Récurrents (RNN)**

Les Réseaux de Neurones Récurrents (RNN) sont une classe spécifique de réseaux de neurones adaptée au traitement de données séquentielles, où l'ordre et la dépendance temporelle sont essentiels. Contrairement aux réseaux de neurones traditionnels, les RNN intègrent des connexions récurrentes qui leur permettent de conserver une mémoire interne des étapes précédentes. Cette capacité leur permet de modéliser efficacement des séquences de données de longueur variable et de capturer des relations complexes et des dépendances à long terme entre les éléments de la séquence.[12]

Les RNN sont couramment utilisés dans la prédiction de séries temporelles pour anticiper les tendances futures en fonction des données passées. Leur capacité à gérer la mémoire à long terme est particulièrement bénéfique pour des applications telles que la traduction automatique, où ils peuvent analyser et générer du texte en prenant en compte le contexte antérieur des mots et des phrases. En outre, les RNN trouvent des applications dans des domaines tels que la reconnaissance de la parole, la génération de musique, et la modélisation de séquences biologiques, illustrant leur polyvalence et leur importance dans le domaine de l'apprentissage profond.[12]

### **c) Les Réseaux de Neurones Profonds (DNN)**

Les Réseaux de Neurones Profonds (DNN), aussi appelés réseaux de neurones à propagation avant, représentent une classe avancée de modèles d'apprentissage profond composés de plusieurs couches de neurones. Conçus pour apprendre de manière autonome des représentations complexes à partir des données, ils sont particulièrement efficaces pour



résoudre des tâches complexes telles que la classification d'images, la reconnaissance vocale, la traduction automatique, et bien d'autres encore.[30]

L'architecture des DNN comprend généralement une couche d'entrée où les données sont introduites, plusieurs couches cachées où les transformations non linéaires sont appliquées aux données, et une couche de sortie qui génère les prédictions finales. Chaque neurone dans les couches cachées utilise des fonctions d'activation comme ReLU ou tanh pour introduire de la non-linéarité, permettant au réseau d'apprendre des représentations de plus en plus abstraites à mesure que les données passent à travers les différentes couches.[30]

Les applications des DNN sont diverses et incluent des domaines tels que la vision par ordinateur, où ils sont utilisés pour la détection d'objets, la segmentation d'images, et la reconnaissance faciale. En traitement du langage naturel, les DNN sont cruciaux pour la compréhension du langage, la génération de texte, et la traduction automatique. Leur capacité à modéliser des données complexes et à extraire des caractéristiques significatives en fait des outils essentiels pour de nombreuses applications modernes de l'intelligence artificielle et de l'informatique.[30]

#### **d) Les Réseaux de Neurones Long Short-Term Memory (LSTM)**

Les Réseaux de Neurones Long Short-Term Memory (LSTM) sont une variante avancée des réseaux de neurones récurrents (RNN), conçue pour capturer efficacement les dépendances à long terme dans les données séquentielles. Contrairement aux RNN traditionnels, les LSTM intègrent des mécanismes de portes (ou gates) qui régulent le flux d'informations à travers les cellules neuronales, permettant au réseau de conserver et d'utiliser des informations sur de longues périodes [30].

L'architecture des LSTM comprend des cellules spéciales avec des portes d'oubli, d'entrée et de sortie, comme illustré dans la Figure 1.11. Ces portes contrôlent la manière dont les informations sont ajoutées ou supprimées de la mémoire cellulaire à chaque étape temporelle, permettant ainsi au réseau de traiter efficacement les séquences de données avec des dépendances temporelles complexes.

Les LSTM sont largement utilisés dans des applications telles que la prédiction de séries temporelles, la traduction automatique, la reconnaissance de la parole, et la génération de texte. Leur capacité à gérer la mémoire à long terme et à modéliser des séquences avec des dépendances à plusieurs niveaux en fait une méthode puissante pour de nombreux problèmes de traitement séquentiel de données..[12]

#### **e) Les Réseaux de Neurones Transformer**

Les Réseaux de Neurones Transformers ont révolutionné l'apprentissage profond depuis leur introduction par Vaswani et al. en 2017, notamment dans le traitement du langage naturel (NLP) et la prédiction énergétique. Ils exploitent efficacement les mécanismes d'attention et les réseaux de neurones positionnels pour capturer les dépendances complexes et à long terme dans les séries temporelles. Contrairement aux approches traditionnelles, les Transformers utilisent un mécanisme d'attention multi-tête pour évaluer simultanément les relations entre tous les éléments d'une séquence, permettant ainsi de modéliser des séquences de longueur variable sans dépendre de connexions récurrentes ou convolutives. En intégrant des informations sur la position relative des éléments, les réseaux de neurones positionnels garan-

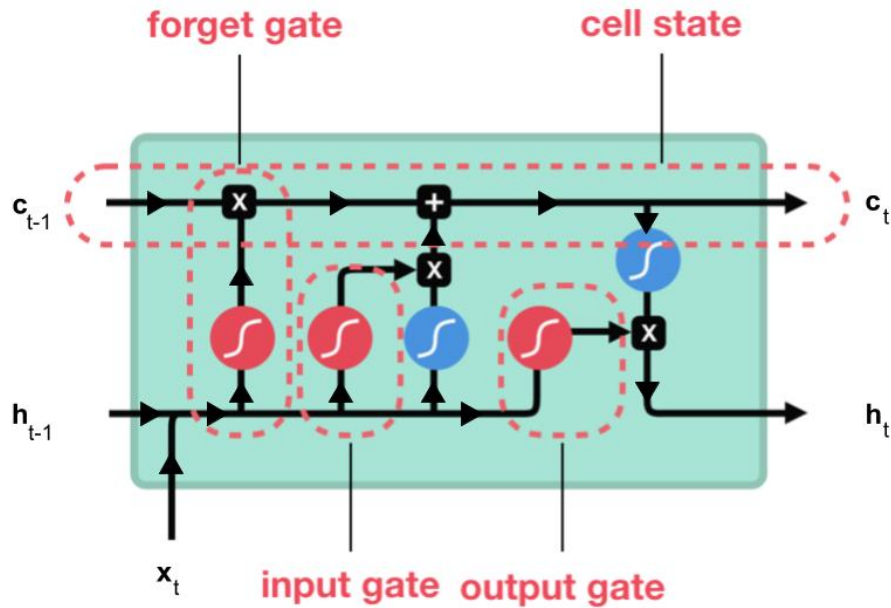


FIGURE 1.11 – Cellule Lstm [12]

tissent une compréhension précise de l'ordre temporel des événements. Dans le domaine de la prédiction énergétique, les Transformers se distinguent par leur capacité à apprendre des modèles complexes et à fournir des prévisions robustes, essentielles pour la gestion efficace des réseaux électriques et la planification de la production d'énergie [33, 16].

## 1.7 Comment choisir la méthode a utilisé ?

Pour choisir la méthode la plus adaptée lors de l'application du Deep Learning ou de méthodes classiques dans le domaine des séries temporelles, plusieurs considérations doivent être prises en compte. Tout d'abord, il est essentiel d'examiner attentivement la nature des données temporelles, en tenant compte de leur fréquence, de la présence de saisonnalité, de tendances et d'anomalies. Ensuite, déterminer clairement l'objectif de la prédiction est crucial, que ce soit pour anticiper des tendances à long terme, détecter des modèles saisonniers ou capturer des variations à court terme.

Pour les méthodes classiques telles que les modèles ARIMA, les moyennes mobiles et les techniques de régression linéaire, elles sont souvent privilégiées dans des scénarios où les données sont relativement simples et où une interprétation claire des résultats est requise. Ces approches sont bien établies et peuvent être efficaces lorsque les relations temporelles sont linéaires ou modérément non linéaires.

En revanche, le Deep Learning, avec des modèles comme les Réseaux de Neurones Convolutionnels (CNN), les Réseaux de Neurones Récurrents (RNN) et les Réseaux de Neurones Transformers, offre une flexibilité et une capacité à capturer des dépendances temporelles complexes et des structures non linéaires dans les données. Ces modèles sont particulièrement puissants dans des contextes où les relations entre les données sont hautement non linéaires ou où des patterns abstraits doivent être identifiés, par exemple dans des données sensorielles ou textuelles.

La décision entre le Deep Learning et les méthodes classiques dépend donc de la complexité des données, de la nécessité d'interprétabilité des résultats, ainsi que des ressources disponibles pour l'entraînement et le déploiement des modèles. En fonction de ces facteurs, il est possible de sélectionner la méthode la plus appropriée pour optimiser à la fois la précision des prédictions et l'efficacité des ressources utilisées dans l'analyse des séries temporelles.

## 1.8 Conclusion

Ce premier chapitre nous a permis d'explorer les concepts fondamentaux des séries temporelles en présentant une diversité de modèles et de méthodes existants. Nous avons examiné des approches traditionnelles telles que les modèles ARIMA, les moyennes mobiles, ainsi que les techniques de régression linéaire, qui ont été largement utilisées dans l'analyse et la prédiction des séries temporelles. De plus, nous avons abordé l'importance croissante de l'intelligence artificielle, en particulier des techniques de Deep Learning, dans ce domaine. Ces avancées, notamment les Réseaux de Neurones Convolutionnels (CNN), les Réseaux de Neurones Récurrents (RNN) et les Réseaux de Neurones Transformers, ont considérablement amélioré les capacités de modélisation et de prédiction des séries temporelles. En synthétisant ces concepts, nous avons jeté les bases nécessaires pour explorer plus en profondeur les méthodes modernes de prédiction des séries temporelles et leur application dans divers domaines. Le chapitre suivant se concentrera spécifiquement sur une exploration détaillée de la méthode des Transformers.

# Chapitre 2

## Les Transformers pour la Prédiction des Séries Temporelles

### 2.1 Introduction aux Transformers

#### 2.1.1 Contexte et Motivation

##### Historique des Transformers

Les Transformers ont été introduits par Vaswani et al. en 2017 dans l'article "Attention is All You Need". Avant cette innovation, les modèles séquentiels traditionnels, tels que les Réseaux de Neurones Récurrents (RNN) et les Long Short-Term Memory (LSTM), dominaient les tâches de traitement des séquences. Cependant, ces modèles étaient souvent confrontés à des limitations significatives, notamment des problèmes de vanishing gradients et des difficultés à capturer des dépendances à long terme dans les séquences [36].

Les Transformers ont révolutionné le traitement des séquences en remplaçant les mécanismes récurrents par des mécanismes d'attention, permettant ainsi une parallélisation plus efficace et une capacité à capturer des relations complexes à différentes échelles. Leur architecture repose sur l'utilisation de couches d'attention multi-têtes, qui permettent au modèle de se concentrer simultanément sur différentes parties de la séquence d'entrée. Cette approche a conduit à des avancées majeures dans diverses applications, notamment le traitement du langage naturel (NLP), où les modèles basés sur les Transformers, tels que BERT et GPT, ont établi de nouveaux standards de performance [16].

Les Transformers ont également été étendus à d'autres domaines tels que la prédiction des séries temporelles. Dans ce contexte, les Transformers exploitent efficacement les mécanismes d'attention pour capturer les dépendances complexes et à long terme dans les données temporelles. Cette capacité est cruciale pour des applications comme la prédiction de la consommation d'énergie future, où une compréhension précise des tendances et des variations temporelles est essentielle pour la planification et la gestion efficace des ressources énergétiques..

##### Pourquoi les Transformers sont-ils adaptés aux séries temporelles ?

Les séries temporelles présentent des caractéristiques particulières qui nécessitent des modèles capables de capturer des dépendances temporelles à diverses échelles. Les Transfor-

mers, grâce à leur architecture basée sur l'attention, sont particulièrement bien adaptés à cette tâche pour plusieurs raisons tel que la parallélisation, Flexibilité et Adaptabilité ... [16].

## 2.1.2 Comparaison avec les Méthodes Traditionnelles

### Limites des Méthodes Classiques (ARIMA, Moyennes Mobiles)

Les méthodes classiques de prédiction des séries temporelles, telles que les modèles ARIMA (AutoRegressive Integrated Moving Average) et les moyennes mobiles, ont été largement utilisées en raison de leur simplicité et de leur capacité à modéliser des séries temporelles stationnaires. Cependant, ces méthodes présentent plusieurs limites :

- **Relations Linéaires :** ARIMA et les moyennes mobiles sont principalement conçus pour capturer des relations linéaires dans les données. Cela limite leur capacité à modéliser des relations non linéaires complexes qui sont courantes dans les séries temporelles réelles [36].
- **Capacité à Long Terme :** Les méthodes comme ARIMA ont des capacités limitées à capturer des dépendances à long terme. Elles sont efficaces pour les relations de court terme, mais peuvent échouer à modéliser correctement les dépendances à longue échéance [36].
- **Paramétrage et Complexité :** L'identification des paramètres optimaux pour les modèles ARIMA ( $p, d, q$ ) peut être complexe et nécessite souvent une expertise en analyse de séries temporelles. De plus, ces modèles peuvent devenir difficiles à ajuster et à interpréter lorsque les séries temporelles présentent des structures complexes.

### Comparaison Synthétique :

Le tableau suivant compare trois approches différentes pour l'analyse des séries temporelles : les méthodes classiques comme ARIMA et les moyennes mobiles, les réseaux de neurones récurrents (RNN) ou LSTM, et les méthodes basées sur les transformées.

Les méthodes classiques comme ARIMA sont efficaces pour traiter des séries temporelles stationnaires et modéliser des relations linéaires, mais elles ont des limitations dans la capture des dépendances à long terme et la représentation complexe des données. En revanche, les RNN/LSTM sont excellents pour capturer les dépendances à long terme grâce à leur capacité à mémoriser des séquences, mais ils sont moins adaptés à la modélisation linéaire et peuvent être difficiles à paralléliser. Les méthodes basées sur les transformées offrent une représentation riche des données et une bonne scalabilité, en plus de faciliter l'intégration de variables externes, mais elles peuvent être moins efficaces pour traiter la non-stationnarité et modéliser les relations linéaires.

Le choix entre ces approches dépend des caractéristiques spécifiques des données et des objectifs d'analyse : les méthodes classiques pour la simplicité et la linéarité, les RNN/LSTM pour les dépendances à long terme, et les méthodes basées sur les transformées pour la représentation riche et la scalabilité.

Caractéristique	Méthodes Classiques (ARIMA, Moyennes Mobiles)	RNN/LSTM	Transforme
Stationnarité	Oui	Non	Non
Modélisation des Relations Linéaires	Oui	Non	Non
Capturer des Dépendances à Long Terme	Limité	Modéré	Excellent
Parallélisation	Non	Non	Oui
Intégration de Variables Exogènes	Difficile	Difficile	Facile
Richesse des Représentations	Faible	Modérée	Élevée
Scalabilité	Limité	Modéré	Élevé

TABLE 2.1 – Comparaison Synthétique [35].

## 2.2 Architecture des Transformers

### 2.2.1 Structure de Base

#### Description Générale des Composants

Les Transformers sont constitués de plusieurs composants clés, chacun jouant un rôle crucial dans le traitement et la transformation des séquences d'entrée [37]. Voici une description générale de ces composants :

- **Embeddings** : Les données d'entrée, telles que des mots ou des valeurs de séries temporelles, sont converties en vecteurs denses à l'aide de couches d'embeddings. Ces vecteurs représentent chaque élément de la séquence dans un espace de dimension plus élevée[36].
- **Attention Multi-têtes** : Ce mécanisme permet au modèle de se concentrer sur différentes parties de la séquence simultanément. Chaque tête d'attention apprend à capturer différents aspects des relations entre les éléments de la séquence [36].
- **Feedforward Networks** : Après le mécanisme d'attention, les données passent par des réseaux de neurones feedforward entièrement connectés, appliqués indépendamment à chaque position de la séquence.

La figure ci-dessous montre l'architecture de base des Transformers

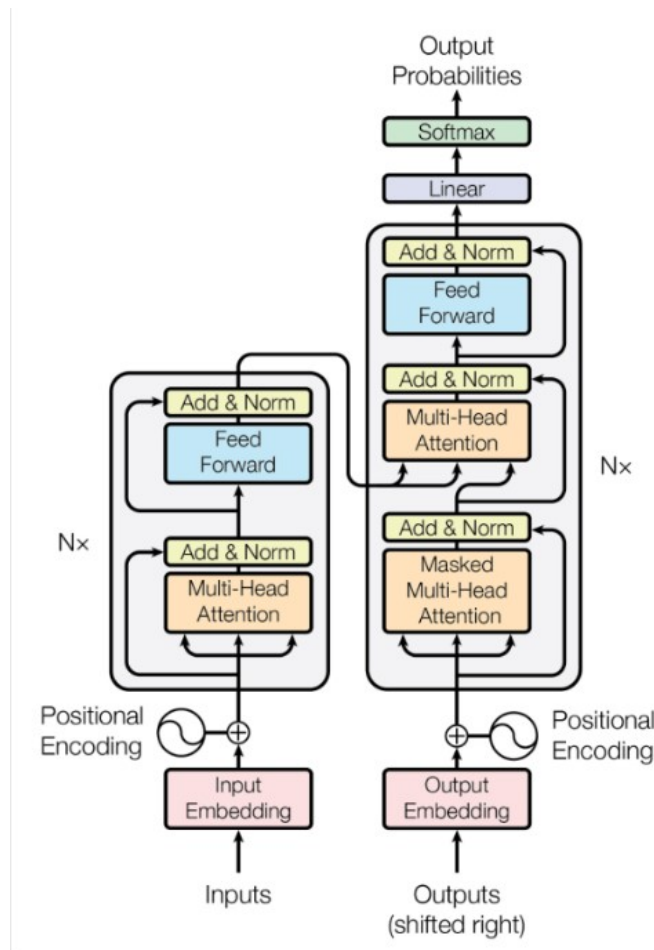


FIGURE 2.1 – l'architecture de base des Transformers[36]

## 2.2.2 Couches d'Encodage et de Décodage

### Fonctionnement des Encodeurs et des Décodeurs

**Encodeurs** : Un encodeur dans un Transformer est composé de plusieurs couches identiques empilées [37]. Chaque couche d'encodeur a deux sous-couches principales :

- **Self-Attention Multi-têtes** : Cette sous-couche permet à chaque élément de la séquence d'attentionner toutes les autres positions de la séquence pour capturer les dépendances globales[35].
- **Feedforward Network** : Un réseau de neurones entièrement connecté qui traite les sorties de l'attention [35].

Chaque sous-couche est entourée de résidus et de normalisation de couche pour stabiliser et accélérer l'entraînement.

**Décodeurs** : Un décodeur est également composé de plusieurs couches identiques empilées, mais il comporte une sous-couche d'attention supplémentaire qui regarde les sorties de l'encodeur (attention encodage-décodage) :

- **Masked Self-Attention** : Similaire à la self-attention de l'encodeur, mais avec un masque pour empêcher l'accès aux positions futures dans la séquence de sortie.
- **Attention Encodage-Décodage** : Cette sous-couche permet au décodeur de focaliser



sur les représentations encodées par l'encodeur.

- **Feedforward Network** : Un réseau de neurones entièrement connecté similaire à celui de l'encodeur [35].

Chaque sous-couche dans le décodeur est également entourée de résidus et de normalisation de couche.

La figure ci-dessous montre la structure des encodeurs et des décodeurs dans un Transformer :

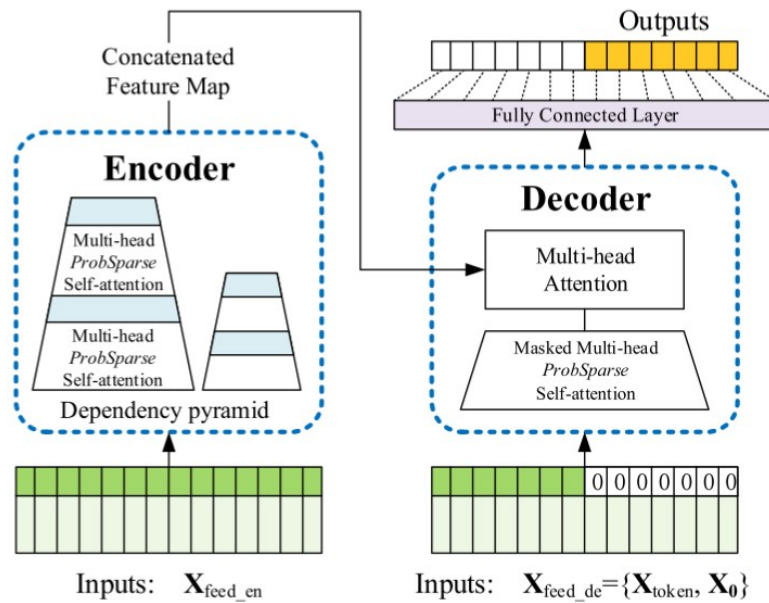


FIGURE 2.2 – la structure des encodeurs et des décodeurs dans un Transformer[37].

## 2.2.3 Modèles Pré-entraînés et Fine-tuning

### Introduction à BERT et Autres Modèles Pertinents

BERT (Bidirectional Encoder Representations from Transformers) est un modèle pré-entraîné révolutionnaire initialement conçu pour les tâches de traitement du langage naturel (NLP). Cependant, son architecture basée sur les Transformers s'est avérée être polyvalente et peut être adaptée avec succès à d'autres domaines, y compris la prédiction des séries temporelles [2].

- **BERT** : BERT utilise une architecture d'encodeur bidirectionnel de Transformers, ce qui lui permet de capturer les dépendances contextuelles dans les deux directions. Cette capacité est extrêmement bénéfique pour les séries temporelles, où les dépendances à la fois passées et futures sont importantes pour des prédictions précises [2].
- **TFT (Temporal Fusion Transformer)** : Contrairement à BERT, TFT est spécialement conçu pour les séries temporelles. Il intègre des mécanismes d'attention pour fusionner des informations temporelles et contextuelles, ce qui en fait un choix puissant pour les tâches de prédiction temporelle [2].

## 2.2.4 Optimisation et Algorithmes d'Entraînement

### Description de l'Optimisateur AdamW

AdamW est une variante de l'optimiseur Adam, qui est largement utilisé dans l'entraînement des réseaux de neurones profonds. AdamW modifie la méthode de régularisation de l'optimiseur Adam pour mieux contrôler la norme des poids du modèle pendant l'entraînement [7]. Voici les caractéristiques clés de l'optimisateur AdamW :

- **Adaptativité des Taux d'Apprentissage** : Comme Adam, AdamW ajuste automatiquement les taux d'apprentissage pour chaque paramètre du modèle en fonction de l'historique des gradients. Cela permet une convergence plus rapide et une meilleure généralisation [7].
- **Correction de la Décroissance des Poids** : Contrairement à Adam, AdamW applique une correction de décroissance des poids pendant la mise à jour des poids du modèle. Cela contribue à prévenir le surajustement en stabilisant la norme des poids [7].
- **Régularisation L2** : AdamW inclut une régularisation L2 directement dans la mise à jour des poids, ce qui favorise des solutions plus régulières et moins susceptibles de surajuster.

## 2.3 Méthodes de Calcul d'Erreur

### 2.3.1 Mean Squared Error (MSE)

Le Mean Squared Error (MSE) est l'une des métriques les plus couramment utilisées pour évaluer la performance des modèles de régression, y compris dans le contexte de la prédiction des séries temporelles. Définition et Formule Le MSE mesure la moyenne des carrés des écarts entre les valeurs prédites par le modèle ( $\hat{Y}_i$ ) et les valeurs réelles ( $Y_i$ ) pour un ensemble de données [37]. Sa formule mathématique est la suivante :

$$\text{MSE} = \frac{1}{n_1} \sum_{i=1}^n (\hat{y}^i - y_i)^2 \quad (2.1)$$

- $n$  est le nombre d'échantillons dans l'ensemble de données.
- ( $\hat{Y}_i$ ) est la valeur prédite par le modèle pour l'échantillon  $i$ .
- ( $Y_i$ ) est la valeur réelle pour l'échantillon  $i$ .

#### — Autres Applications Possibles des Transformers

Outre la prédiction des séries temporelles, les Transformers ont un large éventail d'applications potentielles dans divers domaines. Par exemple, ils peuvent être utilisés pour la traduction automatique, la génération de texte, la reconnaissance d'entités nommées, la modélisation du langage naturel et bien plus encore. Leur architecture flexible et leur capacité à capturer des dépendances à long terme en font des candidats idéaux pour une gamme diversifiée de tâches de traitement de données. Malgré quelques limitations, les Transformers présentent un grand potentiel pour la prédiction des séries temporelles et d'autres applications de traitement de données. Leur évolution continue et leur adaptation à des tâches spécifiques promettent d'ouvrir de nouvelles perspectives passionnantes dans le domaine de l'apprentissage automatique et de l'intelligence artificielle. [37]

## 2.4 Conclusion

Enfin, ce chapitre offre une discussion approfondie sur les performances des Transformers dans la prédiction des séries temporelles, ainsi que sur les perspectives futures et les possibilités d'amélioration. Il souligne l'importance croissante des Transformers dans le domaine de l'apprentissage automatique et de l'intelligence artificielle, et leur potentiel à ouvrir de nouvelles voies pour la modélisation et la prédiction des phénomènes temporels complexes. Le chapitre suivant se concentrera sur l'analyse et la conception.

# Chapitre 3

## Analyse des besoins et Conception

### 3.1 Introduction

Ce chapitre se concentre sur deux aspects essentiels du développement de logiciels : l'analyse des besoins et la conception de la solution. Il explore l'identification, l'analyse et la documentation des exigences du projet en collaboration avec les parties prenantes, en utilisant diverses techniques de collecte et de validation des besoins. La transformation de ces exigences en une solution logicielle est ensuite détaillée, avec une attention particulière à l'architecture système, la spécification des composants et la modélisation des fonctionnalités. Enfin, le chapitre aborde la validation et la vérification pour garantir que la solution répond aux besoins des utilisateurs, en mettant en avant les processus de qualité et de conformité.

### 3.2 Présentation du sonalgaz

Sonalgaz est l'opérateur historique de la fourniture d'énergie électrique et gazière en Algérie. Fondée en 1969, Sonalgaz a servi le citoyen algérien pendant plus d'un demi-siècle, fournissant une source d'énergie essentielle à la vie quotidienne. Avec la promulgation de la loi sur l'électricité et la distribution de gaz par canalisations, Sonalgaz est passée d'une entreprise verticalement intégrée à une holding pilotant un groupe industriel composé de multiples sociétés et métiers.

Sonalgaz a toujours joué un rôle crucial dans le développement économique et social du pays.

Sa contribution à la politique énergétique nationale est illustrée par les importants programmes réalisés en électrification rurale et distribution publique de gaz, permettant d'atteindre un taux de couverture en électricité de 98 % pour 10 983 538 clients et un taux de pénétration du gaz de 65% pour 6 886 407 clients. Aujourd'hui, le groupe Sonalgaz se compose de 14 sociétés filiales gérées directement par la holding et de 12 sociétés en participation avec des tiers [5].

### 3.3 Transformation de SONALGAZ en groupe

À partir de 2004, le processus de transformation de Sonelgaz, lié à la loi n° 02-01 du 5 février 2002, a permis non seulement l'ouverture de la production d'électricité à la concurrence, mais aussi la séparation des fonctions de production, de transport et de distribution de l'électricité et du gaz en filiales distinctes sous forme de sociétés par actions (SPA). Ce processus a conduit à la création de nouvelles filiales à partir de 2004 [5].

En 2004, les nouvelles entités créées incluent :

- Une société de transport du gaz : le gestionnaire du réseau de transport du gaz (GRTG).
- Une société de production regroupant les centrales existantes : Sonelgaz Production de l'Électricité (SPE).
- Une société de transport de l'électricité, gestionnaire transitoire du système de production-transport : le gestionnaire du réseau de transport de l'électricité (GRTE).[5]
- La première modification des statuts du fonds des œuvres sociales et culturelles (FOSC).

En 2005, plusieurs entités et réformes importantes ont vu le jour :

- Création de la Commission de Régulation de l'Électricité et du Gaz (CREG), un organisme indépendant et autonome doté de la personnalité juridique, avec trois missions principales : réaliser et contrôler le service public, conseiller les pouvoirs publics sur l'organisation et le fonctionnement des marchés de l'électricité et du gaz, et surveiller le respect des lois et règlements relatifs à ces marchés [5].
- Création de la Société Civile de Médecine du Travail (SMT).
- Création d'une société de recherche et développement de l'électricité et du gaz (CREDEG).
- Fusion des quatre sociétés de maintenance et de prestations de véhicules (MPV) en une seule entité.
- Fusion des trois sociétés de maintenance des transformateurs en une seule société (SMT).
- Préparation à la filialisation de la distribution avec la création de quatre directions générales régionales [5].

En 2006, quatre nouvelles sociétés de distribution ont été créées :

1. Sonelgaz Distribution Alger (SDA)
2. Sonelgaz Distribution Centre (SDC)
3. Sonelgaz Distribution Est (SDE)
4. Sonelgaz Distribution Ouest (SDO)

En 2006 :

- Mise en place de la société opérateur système production-transport : OS, SPT.

- Nouvelle modification des statuts du fonds des œuvres sociales et culturelles.
- Réintégration des cinq sociétés de travaux : KAHRAKIB, KANAGAZ, KAHRIF, ETTERKIB et INERGA [5].

En 2007 :

- Création de l'Institut de formation en électricité et Gaz (IFEG).
- Réorganisation de la SPE en quatre pôles nationaux de production d'électricité, dont un pôle sud .

En 2009 :

- Création de la Société de Gestion du Patrimoine Immobilier (SOPIG).
- Création de la Compagnie d'Engineering de l'Électricité et du Gaz (CEEG).
- Création de la Société des Systèmes (ELIT).

### 3.4 Les différents types de clients de sonalgaz :

Organisation de la Sonelgaz Distribution Direction de Bejaia Les types de clients sont définis en fonction du niveau de tension (basse tension ou haute tension) qui leur est attribué [5]. On distingue parmi eux :

- **Clients HTB (Haute Tension B) :** Ces clients utilisent des lignes HTB qui constituent le réseau de répartition ou d'administration régionale, permettant le transport de l'électricité à l'échelle régionale ou locale. Ces lignes, avec une tension de 63 kV ou 90 kV, desservent les industries lourdes et les grands consommateurs électriques tels que les transports ferroviaires, et connectent avec le réseau secondaire.
- **Clients HTA (Haute Tension A) :** Ces clients bénéficient des lignes HTA qui transportent l'électricité à l'échelle locale vers les petites industries, PME et commerces. Ces lignes ont une tension comprise entre 15 kV et 30 kV.
- **Clients BT (Basse Tension) :** Ces clients utilisent les lignes BT, les plus petites du réseau, avec une tension de 230V ou 400V. Ces lignes alimentent les ménages et les artisans, fournissant l'énergie nécessaire pour les appareils ménagers et autres usages quotidiens.

### 3.5 Présentation de l'organisme d'accueil (Sonelgaz Distribution Direction de Bejaia)

La direction de distribution de Bejaia est rattachée à la Société Algérienne de Distribution de l'Électricité et du Gaz de l'Est (SDE) [19]. Dont le siège est situé à Constantine. Cette direction est composée des unités suivantes :

- Le secrétariat

- Les assistants du Directeur de Distribution
- Le chargé des affaires juridiques
- Le chargé de la communication
- Le chargé de la sécurité
- 9 divisions et 10 agences commerciales (Bejaia Cité Tobal, Bejaia 4 Chemins, El Kseur, Amizour, Sidi Aich, Seddouk, Akbou, Tazmalt, Aokas, Kherrata)
- 5 districts (Bejaia, Akbou, Sidi Aich, Amizour, Kherrata)

### 3.5.1 Organisation de la Sonelgaz Distribution Direction de BÉJAIA

La structure organisationnelle des différentes divisions de la Direction de Sonelgaz à Bejaia est présentée dans la figure 3.1 .

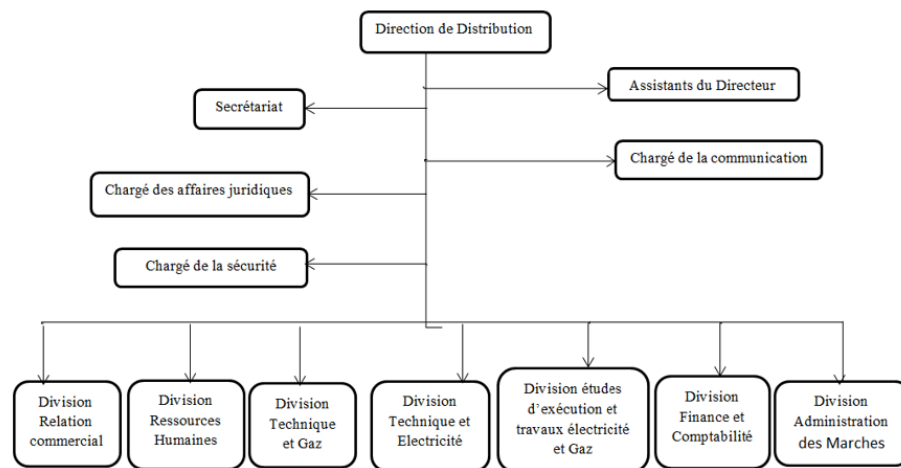


FIGURE 3.1 – Organigramme de la Sonelgaz Direction Distribution de BÉJAIA[5].

#### 3.5.1.1 Division Relations Commerciales

Cette division joue un rôle crucial au sein de la Direction de Distribution [5]. Avec pour principales missions :

- La vente d'électricité et de gaz.
- La gestion des relations clients.
- La facturation sur mémoire (FSM).
- L'analyse statistique (taux de pertes).
- L'assurance de la qualité de service et la satisfaction client.

Elle est subdivisée en trois services :

1. Service Développement des Ventes (RCN) :
  - Prise en charge de toutes les demandes de raccordements des clients en électricité.
  - Orientation et conseil des clients MT/MP concernant les modes de raccordement et la détermination du niveau de tension/pression.

- Facturation des devis RCN et de toutes les prestations.
  - Établissement des ordres d'exécution des travaux après paiement des devis par les clients.
2. Service Recouvrement :
- Gestion des comptes clients IIT/IIP, MT/MP, BT/BP.
  - Respect des délais de facturation pour les groupes en HT/IIP, MT/MP, BT/BP.
  - Avertissement des clients en cas de retard de paiement des factures (ordres de coupure, mise en demeure, etc.).
  - Contrôle et archivage de toutes les pièces comptables.
  - Suivi des clients grands comptes.
  - Suivi de la facturation et du recouvrement FSM (facturation sur mémoire).
3. Service Grands Comptes
- Gestion, télérelève et facturation mensuelle des clients MT/MP, HT/HP.
  - Établissement de contrats de 5 ans à signer avant la mise en service d'un poste client (livraison) MT/MP.
  - Veille à la qualité de service.



## 3.6 Problématique

Sonalgaz rencontre plusieurs défis majeurs, notamment des problèmes de production excessive et insuffisante d'électricité. Cette variation de production pose des défis en termes de gestion efficace des ressources énergétiques. De plus, l'entreprise fait face à des difficultés significatives dans la détection des fraudes, aggravées par des systèmes de comptage d'énergie défectueux chez certains clients. L'intégration de l'intelligence artificielle pourrait être bénéfique pour les entreprises du secteur de l'électricité, les aidant à prendre des décisions plus efficaces en termes de production, de gestion de la fraude et d'analyse des ventes. En utilisant des données historiques et en temps réel, l'analyse prédictive pourrait aider ces entreprises à anticiper les besoins en électricité de leurs clients, à détecter les cas de fraude et à optimiser leurs processus de production. L'électricité représente un élément crucial pour le développement économique mondial, et son importance croît avec les progrès technologiques, l'industrialisation et l'augmentation de la production, ce qui expose les entreprises à de nombreuses décisions complexes, notamment en matière de production d'électricité.

Au cours de mon stage, on a identifié plusieurs problèmes majeurs rencontrés par l'entreprise. Tout d'abord, lorsque la production d'énergie est insuffisante, elle ne satisfait pas les besoins réels des clients. En revanche, en produisant un excès d'électricité, l'entreprise est confrontée à un nouveau défi : la perte d'énergie, car l'électricité ne peut pas être stockée. Ensuite, l'entreprise éprouve des difficultés à détecter les fraudes. En cas de dysfonctionnement du système de comptage d'énergie électrique chez un client, il devient difficile d'évaluer la quantité d'énergie consommée et donc de facturer correctement le client. Enfin, l'entreprise utilise des méthodes d'analyse de ventes traditionnelles et ne dispose pas de systèmes d'aide à la décision sophistiqués.

En intégrant l'intelligence artificielle, ces entreprises pourraient améliorer leur performance dans ces domaines clés. Les systèmes d'analyse prédictive pourraient aider à anticiper la demande en électricité et ainsi optimiser la production pour répondre aux besoins des clients. La détection de fraudes pourrait également être renforcée grâce à des systèmes de surveillance sophistiqués basés sur l'analyse de données. L'utilisation de l'intelligence artificielle pour l'analyse des ventes permettrait d'obtenir des informations précises et en temps réel sur les habitudes de consommation des clients, facilitant ainsi la prise de décisions éclairées..

## 3.7 Solutions proposées

Face aux défis liés à la prise de décision en matière de ventes, il est essentiel pour les entreprises d'adopter des méthodes avancées d'analyse et de prévision des ventes. L'essor de l'intelligence artificielle, en particulier le machine learning et le deep learning, a conduit de nombreuses entreprises à intégrer ces technologies pour améliorer leur gestion et relever les défis pratiques.

Il est indéniable que l'intelligence artificielle, et plus particulièrement le machine learning et le deep learning, est cruciale pour développer des méthodes d'analyse et de prévision des ventes, afin de surmonter les défis associés à la prise de décision. La

création d'une application informatique basée sur des systèmes intelligents pour l'analyse et la prévision des ventes énergétiques répondra aux besoins de la Direction de Bejaia de Sonelgaz Distribution, en améliorant la gestion des ventes et en fournissant une meilleure compréhension des habitudes de consommation des clients.

Dans cette optique, la Direction de Bejaia de Sonelgaz Distribution souhaite instaurer un système intelligent pour analyser les ventes des clients, afin de résoudre les problématiques de prise de décision. Mon objectif est de répondre à cette demande en développant une application informatique dédiée à l'analyse et à la prédiction des ventes énergétiques des clients IITA.

Le principal objectif de ce projet est de fournir à l'entreprise un système capable de prévoir la consommation future des clients, afin de minimiser les pertes d'énergie et de détecter les fraudes. Ce système offrira également aux analystes de Sonelgaz une meilleure visualisation de la consommation des clients, une compréhension approfondie de l'évolution de la consommation électrique et la capacité d'identifier les périodes de forte demande. Les principales fonctionnalités de l'application seront les suivantes :

1. Authentification et inscription des utilisateurs.
2. Visualisation des ventes électriques des différents clients.
3. Prédiction des consommations électriques à l'aide de modèles statistiques LSTM.
4. Prédiction des consommations électriques à l'aide de modèles statistiques regression linear .
5. Prédiction des consommations électriques à l'aide d'un modèle statistique utilisé dans le deep learning (méthode transformers).
6. Analyse comparative des clients en fonction de leur consommation et de la prédiction de leur consommation.
7. Localisation des clients sur une carte géographique.
8. Impression de les données au format PDF ou CSV

En résumé, Notre application fournira à Sonelgaz une solution efficace pour résoudre les problèmes de prise de décision liés aux ventes. En utilisant les dernières avancées en matière d'intelligence artificielle, elle permettra une meilleure gestion des ventes et une compréhension plus approfondie des habitudes de consommation des clients.

### **3.8 Gestion de projet**

La gestion d'un projet informatique se déroule en quatre phases : initiation, planification, exécution, et clôture. L'initiation établit les objectifs et identifie les parties prenantes. La planification détaille les tâches, ressources, délais et budgets. L'exécution voit le travail effectif de l'équipe, surveillé de près par le gestionnaire de projet. En parallèle, surveillance et contrôle sont assurés pour détecter les déviations et ajuster si nécessaire. Enfin, la clôture inclut une évaluation finale et la documentation des leçons apprises. Les méthodes traditionnelles privilégient la prévention des risques dès le début, tandis que les méthodes agiles favorisent l'adaptabilité face aux imprévus.[9]

### 3.8.1 Approches Traditionnelle et Agile pour une Gestion Efficace

#### — La méthode traditionnelle

La méthode traditionnelle de gestion de projet, souvent associée au modèle en cascade, repose sur une approche séquentielle et planifiée. Dans cette méthode, chaque phase du projet est complétée avant de passer à la suivante, avec un accent particulier mis sur la planification exhaustive dès le départ. Les exigences du projet sont généralement définies en amont et figées avant le début du développement. Cette approche vise à minimiser les risques en identifiant et en traitant les problèmes potentiels dès le début, ce qui peut offrir une certaine prévisibilité et stabilité tout au long du projet. Cependant, cela peut également rendre le processus moins flexible pour s'adapter aux changements de dernière minute ou aux nouvelles informations en cours de route.

[9]

## — La méthode Agile

La méthodologie Agile, en contraste avec l'approche traditionnelle, favorise une flexibilité et une réactivité accrues aux changements tout au long du cycle de développement. Des cadres de travail tels que Scrum, Extreme Programming (XP) et Rapid Application Development (RAD) sont souvent utilisés dans les projets Agile pour faciliter cette approche.

[9] La méthodologie Agile, en contraste avec l'approche traditionnelle, favorise une flexibilité et une réactivité accrues aux changements tout au long du cycle de développement. Des cadres de travail tels que Scrum, Extreme Programming (XP) et Rapid Application Development (RAD) sont souvent utilisés dans les projets Agile pour faciliter cette approche.[9]

### 3.8.2 Scrum

Scrum est une méthodologie Agile populaire utilisée pour gérer et développer des projets logiciels complexes. Elle repose sur des cycles de développement itératifs et incrémentaux, appelés "sprints", qui permettent à l'équipe de livrer régulièrement des fonctionnalités utilisables. [9]

#### 3.8.2.1 Les Trois Piliers de Scrum

1. **Transparence** : Tous les aspects du processus doivent être visibles et compréhensibles pour toutes les parties prenantes. Cela favorise la confiance et permet à l'équipe de prendre des décisions éclairées.[9]
2. **Inspection** : Les progrès sont régulièrement inspectés tout au long du processus pour identifier les obstacles, les problèmes et les opportunités d'amélioration. Cette inspection régulière permet à l'équipe de s'adapter rapidement aux changements.[9]
3. **Adaptation** : Une fois que les problèmes sont identifiés, l'équipe s'adapte en conséquence pour résoudre les problèmes et améliorer le processus. Cette capacité d'adaptation continue permet à l'équipe de répondre efficacement aux changements des exigences ou de l'environnement.[9]

#### 3.8.2.2 Les Rôles dans la Méthode Scrum

1. **Product Owner** : Le Product Owner est responsable de maximiser la valeur du produit et de définir les fonctionnalités à développer. Il est chargé de maintenir le carnet de produit, de prioriser les éléments du carnet et de clarifier les exigences pour l'équipe de développement.[9]
2. **Scrum Master** : Le Scrum Master est le chef de file de l'équipe Scrum, chargé de veiller à ce que Scrum soit compris et mis en œuvre correctement. Il aide l'équipe à surmonter les obstacles, facilite les réunions Scrum et encourage l'amélioration continue.[9]

3. **Équipe de Développement** : L'équipe de développement est responsable de la réalisation des éléments du carnet de produit. Elle est auto-organisée et interfonctionnelle, ce qui signifie qu'elle possède toutes les compétences nécessaires pour livrer un produit fonctionnel à la fin de chaque sprint.[9]

### 3.8.2.3 Grandes Étapes de la Méthode Scrum

1. **Planification du Sprint** :

Lors de cette étape, le Product Owner collabore étroitement avec l'équipe de développement pour définir les objectifs du Sprint. Il sélectionne les éléments les plus pertinents du carnet de produit, en priorisant ceux qui apportent le plus de valeur au client. Cette planification repose sur une compréhension approfondie des besoins du client et des exigences du projet, afin d'orienter efficacement le travail de l'équipe pendant le Sprint à venir.[19]

2. **Développement du Sprint** :

Pendant le Sprint, l'équipe de développement s'engage dans l'exécution des tâches définies lors de la planification. Le Product Owner reste disponible pour répondre aux questions et clarifier les points d'ambiguïté sur les éléments du carnet de produit. Il est également chargé de valider les fonctionnalités développées au fur et à mesure qu'elles sont achevées, s'assurant ainsi que le travail réalisé correspond aux attentes et aux besoins du client. [19]

3. **Revue de Sprint** :

À la fin du Sprint, le Product Owner participe à la réunion de revue de Sprint, où l'équipe de développement présente les fonctionnalités achevées. C'est l'occasion pour le Product Owner de voir concrètement le progrès réalisé et d'évaluer si les livrables répondent aux critères de succès définis. Il peut alors fournir des feedbacks précieux sur les fonctionnalités livrées et ajuster les priorités du carnet de produit en conséquence, en fonction des besoins et des retours d'expérience des parties prenantes.[19]

4. **Rétrospectives de Sprint** :

Enfin, le Product Owner participe également à la réunion de rétrospective de Sprint, où l'équipe de développement réfléchit sur son propre processus de travail. Il partage ses observations sur les résultats du Sprint et contribue à identifier les opportunités d'amélioration pour les Sprints à venir. En collaborant avec l'équipe dans cette réflexion, le Product Owner favorise une culture d'amélioration continue et de responsabilisation au sein de l'équipe Scrum.[19]

## 3.9 Délimitation du domaine d'étude

Dans le cadre de cette section, nous précisons les rôles et responsabilités attribués à chaque membre de l'équipe impliquée dans le projet.

### 3.9.1 Répartition des rôles

— Scrum Master :

**M. Foudil MIR** remplit le rôle de facilitateur entre le responsable du produit et l'équipe. Sa principale mission est d'assurer une communication fluide au sein de l'équipe, de lever les obstacles entravant l'atteinte des objectifs fixés pour chaque sprint, et de travailler à l'amélioration de la productivité et du savoir-faire de son équipe.

— Product Owner :

**M. Dalil HADJOUT**, en tant que Chef du produit, est chargé de définir les spécifications fonctionnelles et d'établir la liste des priorités concernant le développement. Il est celui qui valide les fonctionnalités développées.

— L'équipe :

Elle est chargée de transformer les besoins du client en produit ou service. Les membres de l'équipe sont responsables de la rédaction du code, de sa compréhension, de sa modification, de sa gestion, de sa sauvegarde, de ses versions et de sa transformation en code exécutable. Cette équipe est composée de **Mlle DRIAS Mira**.

### 3.9.2 Identification des acteurs

Dans cette section, nous procédons à l'identification des différents acteurs qui pourraient interagir avec le système. Tout d'abord, définissons ce qu'est un acteur : il s'agit d'une entité qui définit le rôle joué par un utilisateur ou par un système qui interagit avec le système modélisé.

Ensuite, nous citons les acteurs spécifiques de notre système :

- **Administrateur** : Il s'agit d'une personne disposant des droits d'accès à l'espace administrateur après son authentification.
- **Utilisateur** : Ce sont des individus déjà enregistrés dans la base de données qui se connectent pour accéder à ses fonctionnalités.

### 3.9.3 Diagramme de contexte

Le diagramme de contexte est une représentation visuelle mettant en évidence les acteurs de haut niveau du système [23].

Les intervenants qui interagissent avec le système au moyen de différents messages dans notre application sont les suivants :

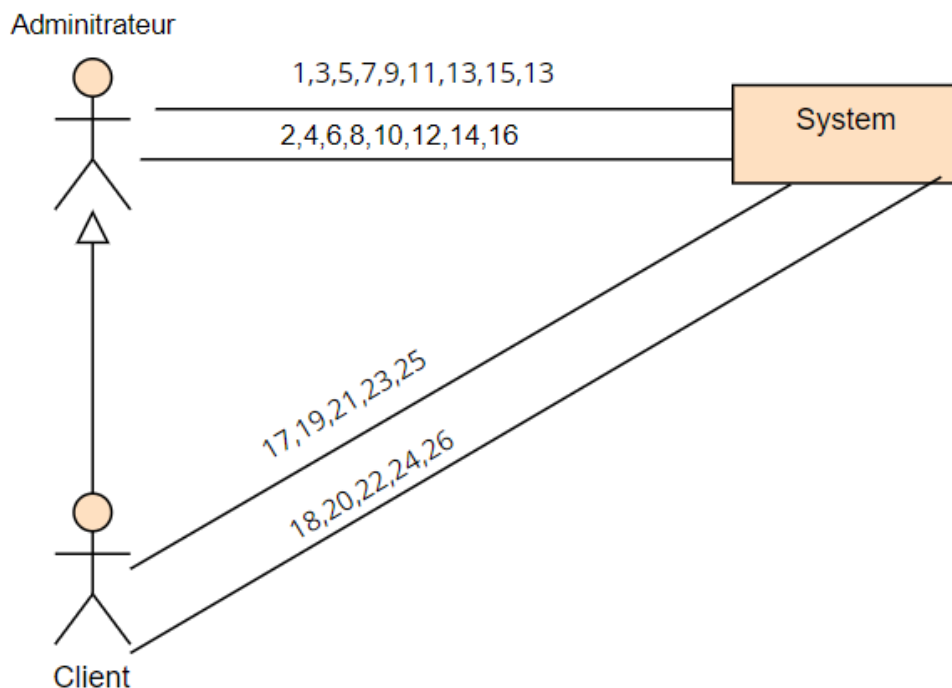


FIGURE 3.2 – Diagramme de contexte

La description des messages est donnée dans le tableau :

Acteur	Numéro	Message	Numéro	Message
Administrateur	1	Demande d'authentification	2	Afficher l'interface d'accueil
	3	Demande d'analyse statistique	4	Afficher de l'interface de visualisation
	5	Demande d'afficher la localisation des clients	6	Affichage de la carte avec la localisation des clients
	7	Demande de consultation de ventes	8	Affichage de l'interface d'un tableaux de consommation de tous les clients
	9	Effectuer un choix	10	Affichage de la page de consommation de client choisie
	11	Effectuer une recherche	12	Affichage du résultat de la recherche
	13	Demande de prédiction	14	Afficher un choix
	15	Effectuer un choix de prédiction	16	Affichage de résultat de prédiction
	13	Demande de comparaison	14	Affichage de l'interface avec la comparaison des résultat des 3 méthode en graphe
	15	Saisir le code de client qui veut prédire sa consommation	16	Affichage de l'interface avec le résultat
17	Demande de télécharger les données pdf ou csv	18	Document télécharger	
Utilisateur(Client)	19	Demande d'authentification	20	Affichage de tableau de bord
	19	Effectuer le choix de l'année qui veut voir sa consommation	21	Affichage de l'interface avec un tableaux et graphe de sa consommation de l'année choisie
	22	Demande de prédiction	23	Affichage de la page de prédiction avec un choix
	24	Effectuer un choix de prédiction	25	Affichage de résultat de prédiction
	26	Demande de comparaison	27	Affichage de l'interface avec la comparaison des résultat des 3 méthode en graphe

TABLE 3.1 – Messages échangés entre le système et les acteurs



## 3.10 Pilotage de notre projet avec SCRUM

### 3.10.1 Product backlog

Le Product backlog représente une liste hiérarchisée des exigences initiales du client concernant le produit à réaliser. Il est formulé sous forme d'un ensemble d'user stories. Ce document est dynamique et évolue continuellement tout au long du projet, en fonction des besoins du client. En d'autres termes, il agit comme un carnet de commandes pour le produit, sous la responsabilité du Product Owner [19]. Dans notre cas, les besoins du client sont exprimés comme suit :

ID du User Story	Nom du UserStory	Description du User Story	Priorité	Statut
US01	Authentification et gestion de la localisation	<ul style="list-style-type: none"><li>- En tant qu'administrateur, je souhaite avoir la capacité de visualiser la localisation géographique des clients dans le secteur économique sur une carte, ce qui me permettra d'obtenir une meilleure compréhension de leur répartition géographique et de leur emplacement précis. Pour ce faire, je dois fournir mon nom d'utilisateur et mon mot de passe pour me connecter à l'application et accéder à ses fonctionnalités.</li><li>- En tant que client, j'aimerais avoir la possibilité de m'inscrire en créant un compte sécurisé, de me connecter avec mes identifiants une fois inscrit, et de me déconnecter lorsque je n'ai plus besoin d'accéder à l'application.</li></ul>	Haute	Fait

ID du User Story	Nom du UserStory	Description du User Stor	Priorité	Statut
US02	Visualisation des données	<p>-En tant qu'administrateur, je désire avoir la possibilité de consulter les données de consommation électrique des clients du secteur économique sur une période spécifique, présentées à la fois sous forme graphique et dans un tableau. Cela me permettra d'analyser les tendances de consommation et d'en tirer des insights significatifs, avec la possibilité de télécharger les données au format CSV ou PDF.</p>	Haute	Fait
US03	Prédiction des ventes	<p>- En tant qu'administrateur, je désire la possibilité de stocker les données de consommation d'électricité des clients ainsi que les résultats de prédiction dans une base de données afin de les utiliser ultérieurement.</p> <p>- En tant qu'administrateur, je souhaite pouvoir appliquer plusieurs méthodes pour anticiper la consommation électrique future de chaque client en se basant sur les données historiques de consommation.</p> <p>- En tant qu'administrateur, j'ai l'objectif de visualiser les prévisions de consommation électrique de mes clients du secteur économique à travers des graphiques et des tableaux pour mieux appréhender ces données</p> <p>- En tant qu'administrateur , je désire pouvoir exporter ces données au format PDF ou CSV</p>	Haute	Fait

<b>ID du User Story</b>	<b>Nom du UserStory</b>	<b>Description du User Stor</b>	<b>Priorité</b>	<b>Statut</b>
US04	Comparaison les résultat de prédiction	En tant qu'administrateur, je souhaite avoir la possibilité de visualiser un graphique comparatif des résultats obtenus par les trois méthodes.	Faible	Fait
US05	Visualisation de la consommation personnelle d'un client. Prédiction des ventes	En tant que client ,je peux avoir ma consommation annuel et mensuel sous forme d'un tableau et sous forme d'un graphe pour mieux appréhender ces données En tant que client, je souhaite pouvoir appliquer plusieurs méthodes pour anticiper la consommation électrique future de chaque client en se basant sur les données historiques de consommation	Haut	Fait

TABLE 3.2 – Product Backlog

### 3.10.2 Planification des sprints

Les user stories précédemment définies dans le Product Backlog sont classées par ordre de priorité. Le travail sera organisé en sprints que nous avons définis.

Un Sprint correspond à une itération, une période allant de 2 à 4 semaines maximum, pendant laquelle une version achevée et utilisable du produit est développée. Un nouveau sprint débute immédiatement après la clôture du précédent. Chaque sprint est défini par un objectif spécifique et une liste de fonctionnalités à réaliser.

Sprint	Nom du sprint	Période
Sprint 1	Authentification et gestion de la localisations	1 semaines
Sprint 2	Visualisation des Consommations	une semaine
Sprint 3	Prédiction des Ventes avec un modèle machine learning avec possibilité d'impression des Données CSV ou PDF	2 semaines
Sprint 4	Comparaison des Consommations	une semaines
Sprint 5	Prédiction des Ventes avec un modèle deep learning avec possibilité d'impression des Données CSV ou PDF	2 semaines
Sprint 6	Visualiser et prédiction de consommation dans chaque compte d'un client	1 semaine

TABLE 3.3 – Planification des Sprints

### 3.11 Diagramme de cas d'utilisation

Le diagramme de cas d'utilisation est un diagramme UML (Unified Modeling Language) utilisé pour une représentation des besoins des utilisateurs par rapport au système [23]. Cas d'utilisation correspond à un certain nombre d'actions que le système devra exécuter en réponse à un besoin d'utilisateur La Figure 3.3 représente le diagramme de cas d'utilisation

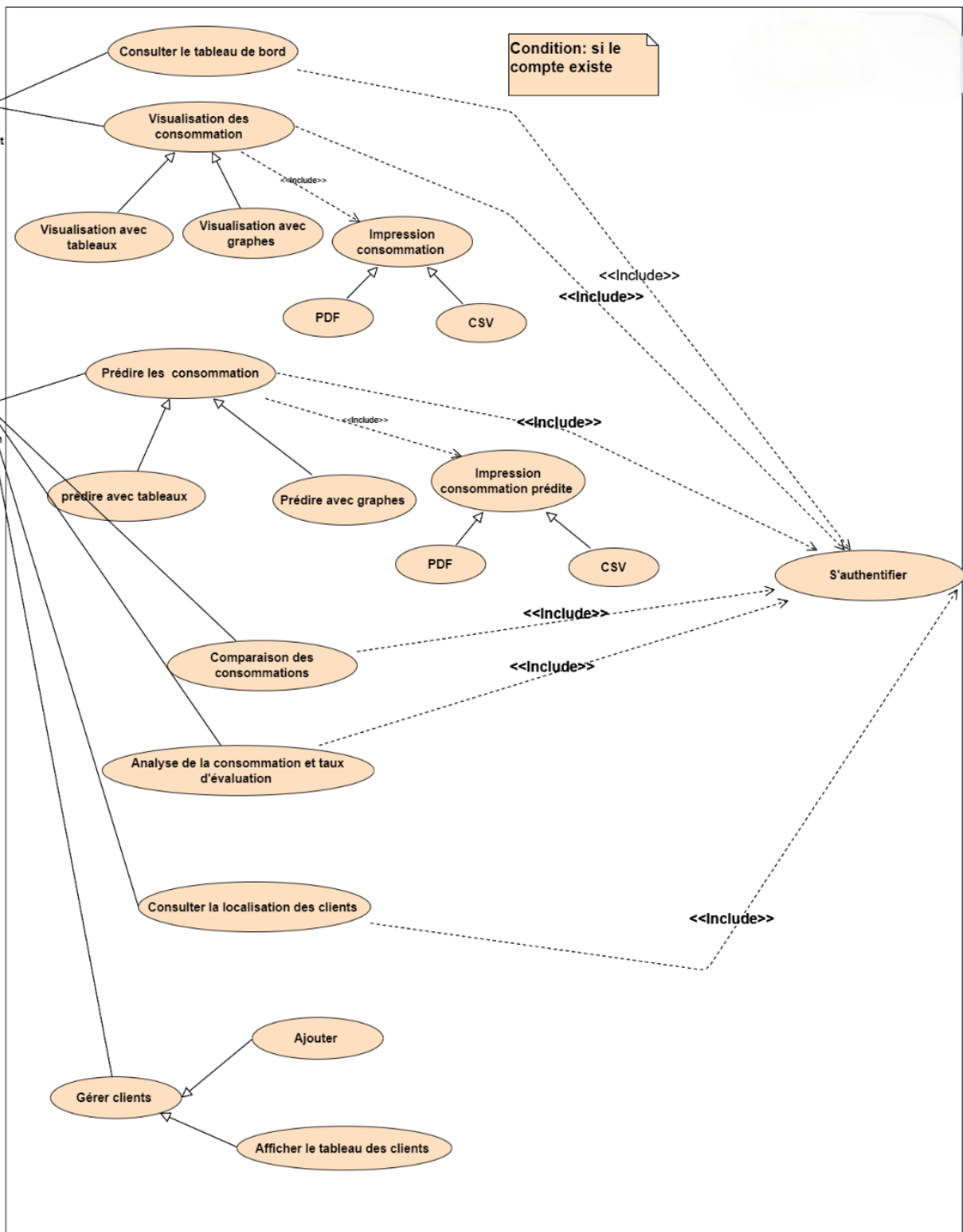


FIGURE 3.3 – Diagramme de cas d'utilisation

## 3.12 Description des cas d'utilisation

Une description textuelle des cas d'utilisation est présentée sous forme de tableau, détaillant les différentes étapes ou interactions entre les acteurs impliqués dans un scénario d'utilisation d'un système, d'une application ou d'un service [23]. Ces étapes sont organisées de manière logique pour illustrer comment un utilisateur peut exploiter le système afin de réaliser une tâche.

Chaque cas d'utilisation doit inclure les éléments suivants :

Nom du cas d'utilisation	
Acteurs	Les acteurs qui vont réaliser le cas d'utilisation
Description	Une description résumée du cas d'utilisation
Auteurs	Les intervenants de l'élaboration de la fiche
Date	La date de rédaction ou mise à jour de la fiche
Pré-conditions	Les conditions nécessaires pour déclencher le cas d'utilisation
Description des scénarios	
Scénario nominal	La description des interactions entre acteur et système, dans le meilleur des cas ou plus précisément le meilleur des scénarios
Scénario alternatif	La description détaillée de situation hypothétique où les événements ou les interactions entre acteurs et systèmes diffèrent de la trajectoire principale attendue, explorant ainsi différentes possibilités et leurs implications
Post-conditions	État du système après réalisation du cas d'utilisation et qui pourra témoigner du bon fonctionnement

TABLE 3.4 – Formalisme de description des cas d'utilisation

## 3.13 Diagrammes de séquences

Les diagrammes de séquences illustrent les interactions échangées entre l'utilisateur et le système.[23] Dans cette étude, nous avons identifié certains cas d'utilisation particulièrement pertinents à représenter par leurs diagrammes de séquences : l'authentification, la visualisation des consommations, la prédiction des ventes, et la comparaison des consommations.

### 3.13.1 Cas d'utilisation « S'authentifier »

Le tableau 3.5 représente la description textuelle du cas "S'authentifier" :

S'authentifier	
Acteurs	Administrateur
Description	Permet à l'utilisateur de s'authentifier en utilisant son nom d'utilisateur et son mot de passe.
Auteur	Drias mira
Date	30 Avril 2024
Pré-conditions	L'utilisateur doit avoir un compte d'utilisateur avec un nom d'utilisateur et un mot de passe valide.
Description des scénarios	
Scénario nominal	<p>1- L'utilisateur ouvre l'application et est redirigé vers l'écran d'authentification.</p> <p>2- L'utilisateur entre son nom d'utilisateur et son mot de passe.</p> <p>3- Le système vérifie que le nom d'utilisateur et le mot de passe correspondent à un compte utilisateur valide.</p> <p>4- Le système redirige l'utilisateur vers l'écran principal de l'application.</p>
Scénario alternatif	<p>L'enchaînement démarre au point 2 de la séquence nominale :</p> <p>1- L'utilisateur entre un nom d'utilisateur ou un mot de passe invalide.</p> <p>2- Le système affiche un message d'erreur correspondant et invite l'utilisateur à saisir de nouveau ses informations d'identification.</p> <p>3- L'utilisateur corrige les erreurs de saisie et valide à nouveau ses informations d'identification.</p> <p>4- Le système vérifie à nouveau les informations d'identification et redirige l'utilisateur vers l'écran principal de l'application si elles sont valides.</p>
Post-conditions	L'utilisateur est authentifié et peut accéder aux fonctionnalités de l'application.

TABLE 3.5 – description des cas d'utilisation "S'authentifier "

La Figure 3.4 représente le diagramme de séquence de cas d'utilisation "S'Authentifier" :

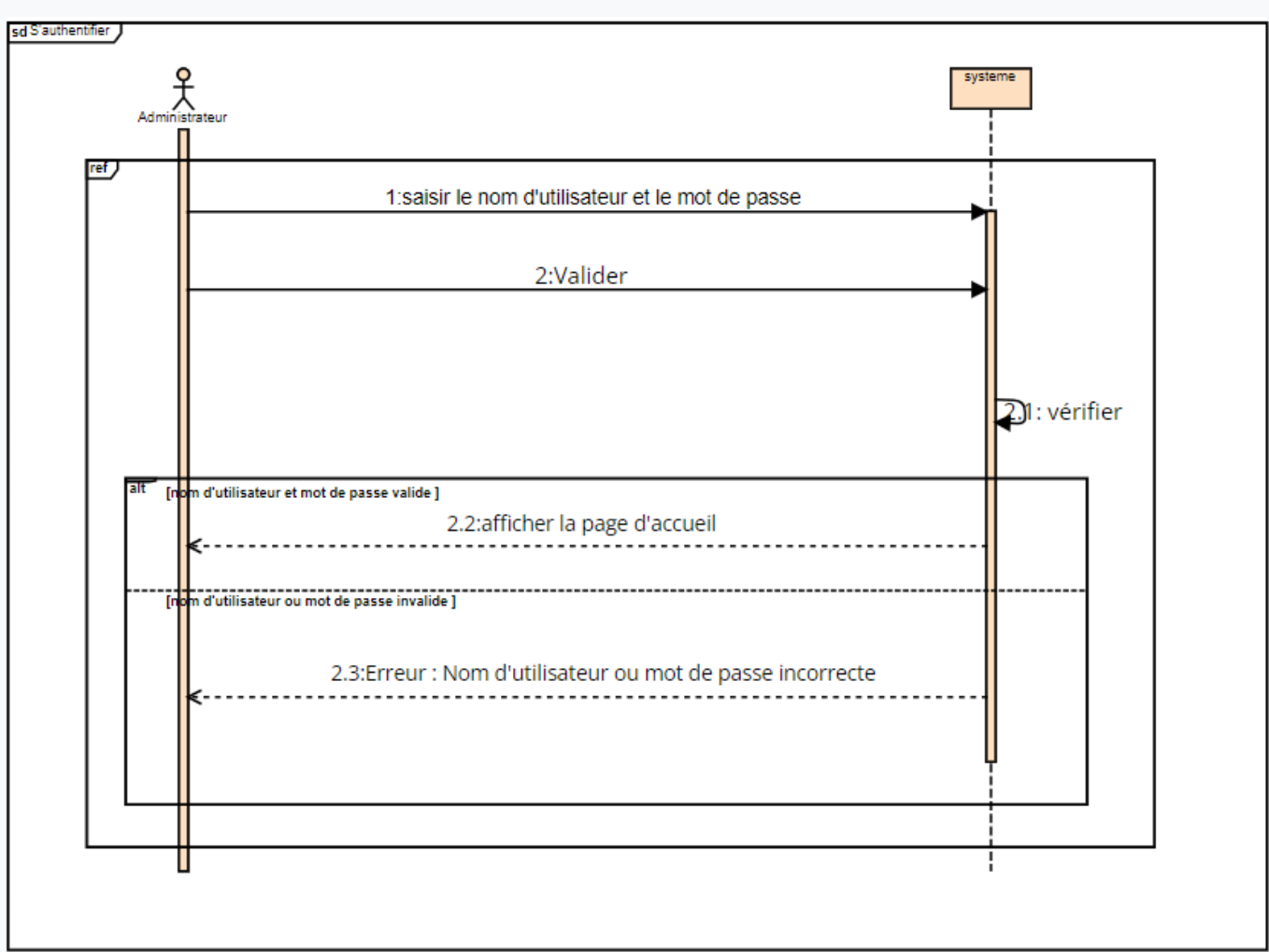


FIGURE 3.4 – Diagramme de cas d'utilisation "S'Authentifier"



### 3.13.2 Cas d'utilisation « Visualiser les consommations »

Le tableau 3.6 représente la description textuelle du cas "Visualiser les consommations" :

Visualiser les consommations	
Acteurs	Administrateur
Description	Permet à l'utilisateur de visualiser les consommations des clients sous deux formats : graphique et tableau. Le format graphique présente les données sous forme de graphiques, tandis que le format tableau les affiche sous forme de tableaux. L'utilisateur peut également imprimer les données au format PDF ou CSV
Auteur	Drias mira
Date	4 Mai 2024
Pré-conditions	L'utilisateur doit être authentifié.
Description des scénarios	
Scénario nominal	<ol style="list-style-type: none"> <li>1- L'utilisateur sélectionne l'option Visualisation dans le menu.</li> <li>2- Le système affiche un tableau complet des clients et leurs consommations, ainsi que des champs pour saisir le code du client dont il souhaite voir sa consommation..</li> <li>3- L'utilisateur effectue un choix de code client et l'année .</li> <li>4- L'utilisateur valide son choix</li> <li>5- Le système affiche les Résultats.</li> </ol>
Scénario alternatif	<p>L'enchaînement démarre au point 3 de la séquence nominale :</p> <ol style="list-style-type: none"> <li>1- Le système essaie d'accéder aux données de consommation correspondantes.</li> <li>2- Le système détecte une erreur d'accès aux données, soit en raison d'une panne de serveur ou d'une erreur de connexion de base de données.</li> <li>3- Le système affiche un message d'erreur informant l'utilisateur qu'il y a un problème technique et qu'il doit réessayer plus tard ou contacter le support technique.</li> </ol>
Post-conditions	Le résultat s'affiche

TABLE 3.6 – description des cas d'utilisation " Visualiser les consommations"

La Figure 3.5 représente le diagramme de séquence de cas d'utilisation "visualiser les consommations"

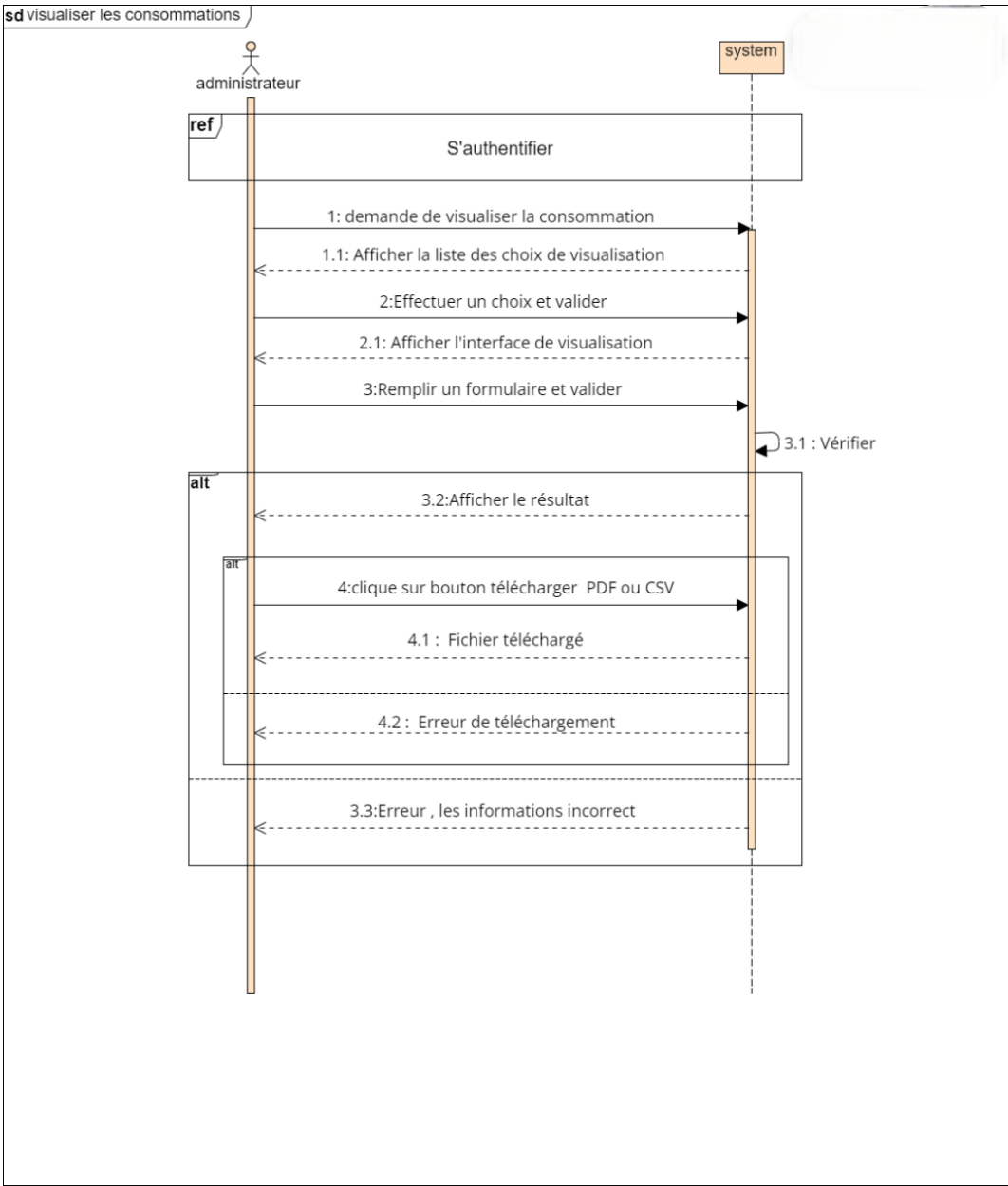


FIGURE 3.5 – Diagramme de cas d'utilisation "visualiser les consommations"

### 3.13.3 Cas d'utilisation « Prédire la consommation »

Le tableau 3.7 représente la description textuelle du cas "Prédire la consommation"

Prédiction la consommation	
Acteurs	Administrateur
Description	Permet à l'utilisateur de visualiser les prévisions de vente pour leur consommation . Le système affiche les prévisions de vente sous forme graphique et tableau. L'utilisateur peut également imprimer les données au format PDF ou CSV
Auteur	Drias mira
Date	15 Mai 2024
Pré-conditions	L'utilisateur doit être authentifié et avoir accès aux données de vente.
Description des scénarios	
Scénario nominal	<ol style="list-style-type: none"> <li>1- L'utilisateur sélectionne l'option "Prédiction" dans le menu.</li> <li>2- Le système affiche un champ.</li> <li>3- L'utilisateur choisit le code de client qui veut afficher sa prédiction .</li> <li>4- Le système affiche l'interface de prédiction .</li> <li>5- Le système affiche le résultat</li> <li>6- L'utilisateur peut imprimer les données affichées au format PDF ou CSV</li> </ol>
Scénario alternatif	<p>L'enchaînement démarre au point 3 de la séquence nominale :</p> <ol style="list-style-type: none"> <li>1- L'utilisateur saisit des valeurs erronées ou manquantes.</li> <li>2- Le système détecte l'erreur de saisie et affiche un message d'erreur correspondant.</li> <li>3- L'utilisateur corrige les erreurs de saisie et valide à nouveau le formulaire.</li> <li>4- Le système génère les prévisions de vente en utilisant les données saisies et les affiche à l'utilisateur.</li> </ol>
Post-conditions	Les prévisions de vente s'affichent.

TABLE 3.7 – description des cas d'utilisation " Prédire la consommation "

La Figure 3.6 représente le diagramme de séquence de cas d'utilisation "prédire la consommation" :

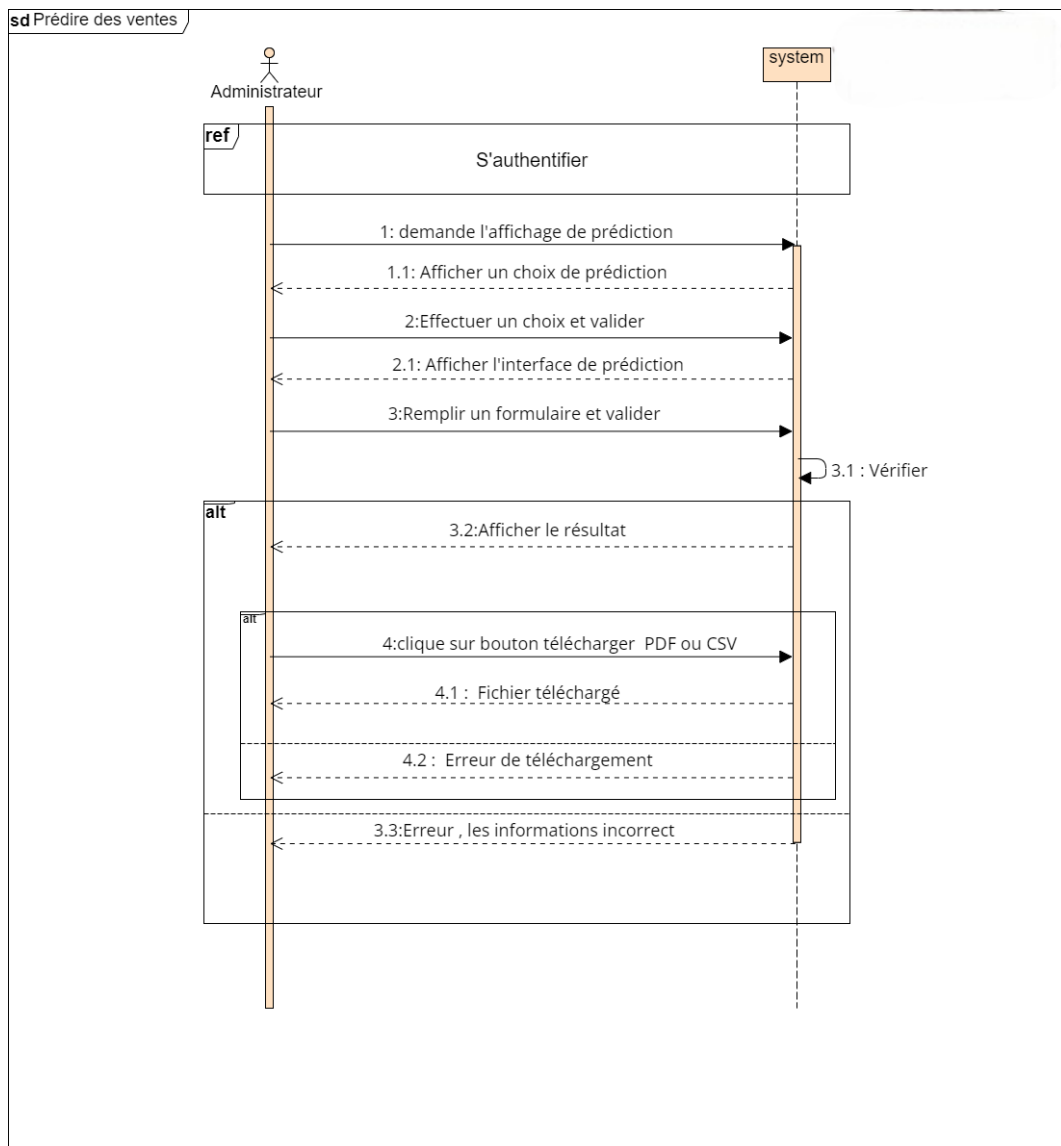


FIGURE 3.6 – Diagramme de cas d'utilisation "prédire la consommation"

### 3.13.4 Cas d'utilisation « Comparaison de la consommation »

Le tableau 3.8 représente la description textuelle du cas "Comparaison de la consommation"

Comparaison de la consommation	
Acteurs	Administrateur
Description	Permet à l'utilisateur de comparer la consommation prédite des 3 méthodes .
Auteur	Drias mira
Date	20 Mai 2024
Pré-conditions	L'utilisateur doit être authentifié.
Description des scénarios	
Scénario nominal	<ol style="list-style-type: none"> <li>1- L'utilisateur sélectionne l'option "Comparaison" dans le menu.</li> <li>2- Le système affiche un champ pour saisir le code client pour lequel il souhaite voir la comparaison des résultats avec les trois méthodes.</li> <li>3- L'utilisateur valide sa saisie en cliquant sur le bouton "Afficher la comparaison".</li> <li>4- Le système affiche un tableau et un graphique présentant les résultats de prédiction obtenus par les trois méthodes</li> </ol>
Scénario alternatif	<p>L'enchaînement démarre au point 4 de la séquence nominale :</p> <ol style="list-style-type: none"> <li>1- Le système détecte une erreur d'accès aux données, soit en raison d'une panne de serveur ou d'une erreur de connexion de base de données.</li> <li>2- Le système affiche un message d'erreur informant l'utilisateur qu'il y a un problème technique et qu'il doit réessayer plus tard ou contacter le support technique.</li> </ol>
Post-conditions	Le résultat des 3 méthode s'affiche dans un tableau et un graphe de comparaison

TABLE 3.8 – description des cas d'utilisation " Comparaison de la consommation "

La Figure 3.7 représente le diagramme de séquence de cas d'utilisation "Comparaison de la consommation"

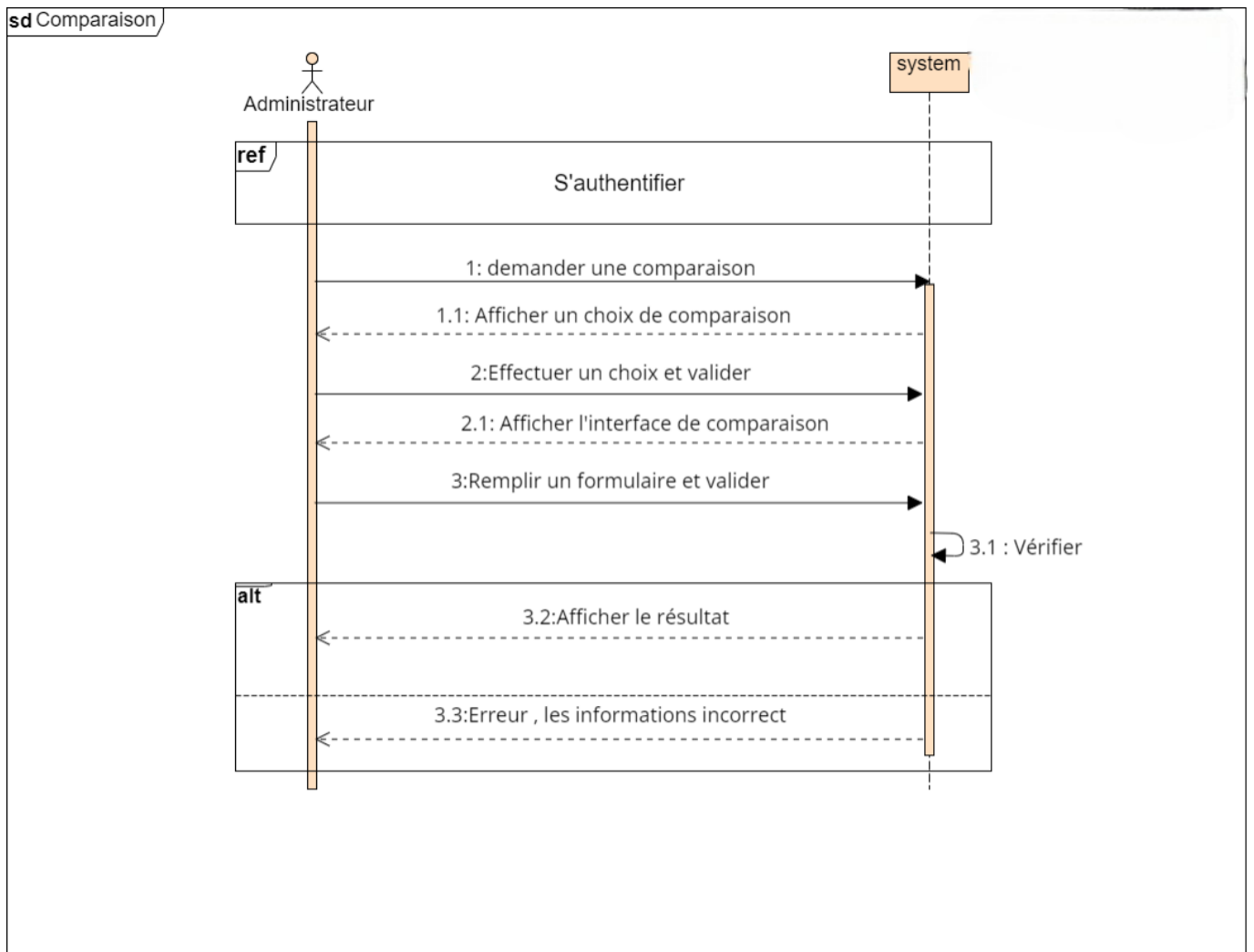


FIGURE 3.7 – Diagramme de cas d'utilisation "Comparaison de la consommation"

### 3.13.5 Cas utilisation “ prédire la consommation d’un client”

Le tableau 3.9 représente la description textuelle du cas ”prédire la consommation d’un client ”

Prédiction la consommation	
Acteurs	Client
Description	Permet à l'utilisateur de visualiser les prévisions de vente pour leur consommation . Le système affiche les prévisions de vente sous forme graphique et tableau
Auteur	Drias mira
Date	15 Mai 2024
Pré-conditions	L'utilisateur doit être authentifié et avoir accès aux données de vente.
Description des scénarios	
Scénario nominal	<ol style="list-style-type: none"> <li>1- Le client sélectionne l'option ”Prédiction ” dans le menu.</li> <li>2- Le système affiche un choix de prédiction.</li> <li>3- L'utilisateur effectue un choix .</li> <li>4- Le système affiche le résultat</li> </ol>
Scénario alternatif	<p>L'enchaînement démarre au point 3 de la séquence nominale :</p> <ol style="list-style-type: none"> <li>1- Le système essaie d'accéder aux données de consommation correspondantes.</li> <li>2- Le système détecte une erreur d'accès aux données, soit en raison d'une panne de serveur ou d'une erreur de connexion de base de données.</li> <li>3- Le système affiche un message d'erreur informant l'utilisateur qu'il y a un problème technique et qu'il doit réessayer plus tard ou contacter le support technique</li> </ol>
Post-conditions	Les prévisions de vente s'affichent.

TABLE 3.9 – description des cas d'utilisation ”prédire la consommation d’un client ”

La Figure 3.8 représente le diagramme de séquence de cas d'utilisation "prédire la consommation d'un client"

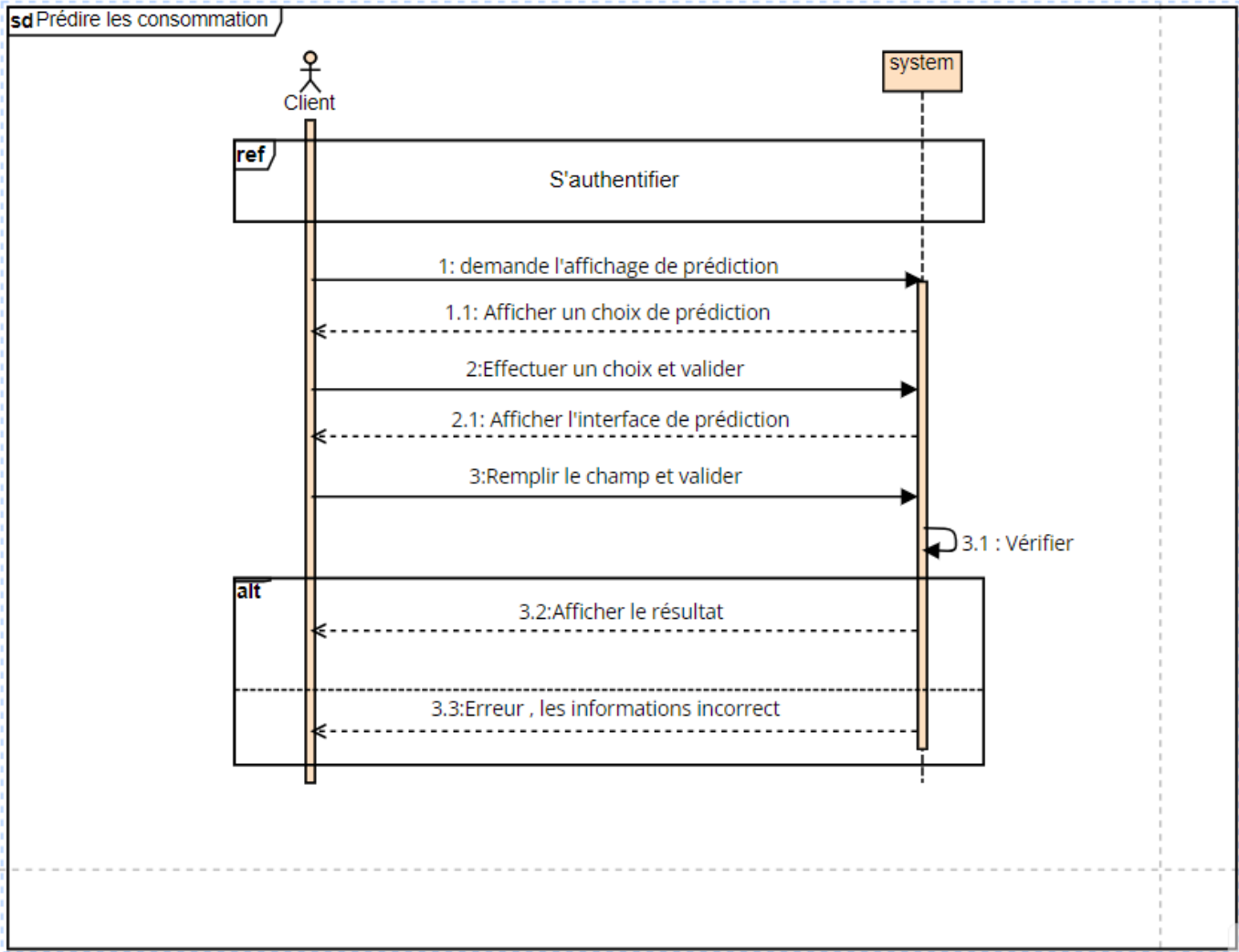


FIGURE 3.8 – Diagramme de cas d'utilisation "prédire la consommation d'un client"



## 3.14 Diagramme de séquence détaillé

Les diagrammes de séquence détaillés permettent de visualiser clairement l'ordre d'exécution des actions, les conditions et les boucles, les appels de méthode et les retours, ainsi que les interactions entre les différents objets du système [23]. Ils sont utiles pour comprendre en détail le fonctionnement d'un scénario ou d'un processus, et peuvent être utilisés pour la conception, le débogage et la documentation des systèmes logiciels .

### 3.14.1 Diagramme de séquence détaillé du cas d'utilisation "S'authentifier" :

L'acteur "Utilisateur" démarre le processus d'authentification en saisissant ses identifiants dans l'interface utilisateur. Le composant "Système d'authentification" reçoit le message et vérifie les informations d'identification fournies en interrogeant une base de données. Ensuite, le composant "Système d'authentification" envoie un message de retour à l'interface utilisateur, indiquant si l'authentification a réussi ou échoué. L'interface utilisateur reçoit ce message et affiche le résultat à l'utilisateur. Si l'authentification échoue, l'interface utilisateur peut afficher un message d'erreur et permettre à l'utilisateur de réessayer. Si l'authentification réussit, l'interface utilisateur redirige l'utilisateur vers la page d'accueil. Enfin, le processus d'authentification se termine.

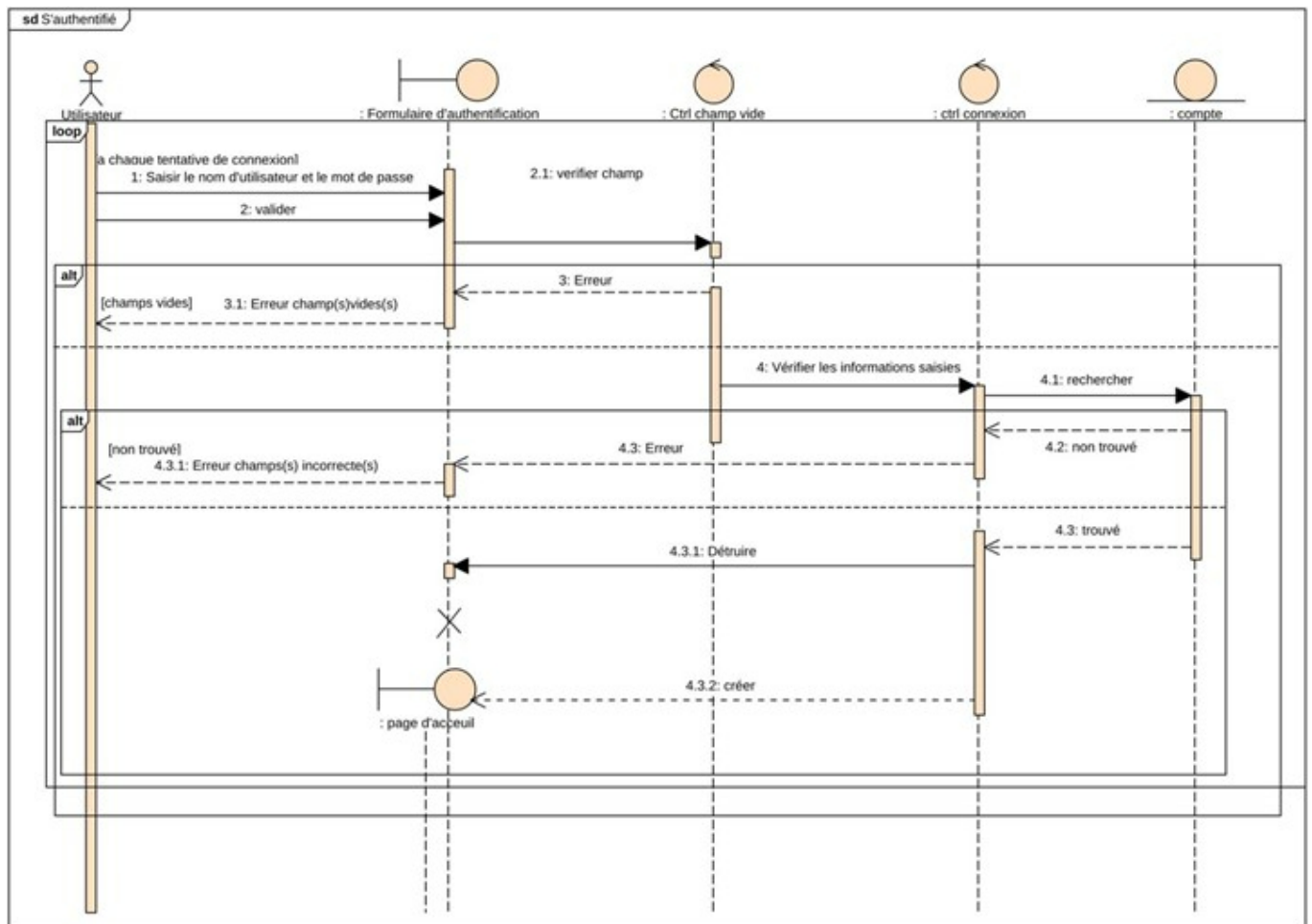


FIGURE 3.9 – Diagramme de séquence détaillé du cas d'utilisation «S'authentifier»

### 3.14.2 Diagramme de séquence détaillé du cas d'utilisation ”Visualiser les consommations”

Le diagramme de séquence détaillé du cas d'utilisation Visualiser les consommations illustre la séquence d’actions et d’échanges de messages entre les acteurs et les composants du système lorsqu’un utilisateur demande à visualiser les informations de consommation. Ce diagramme met en évidence les étapes clés, telles que la demande de visualisation, la récupération des informations de consommation, l’affichage des données à l’utilisateur et la possibilité d’accéder aux détails spécifiques d’une consommation.

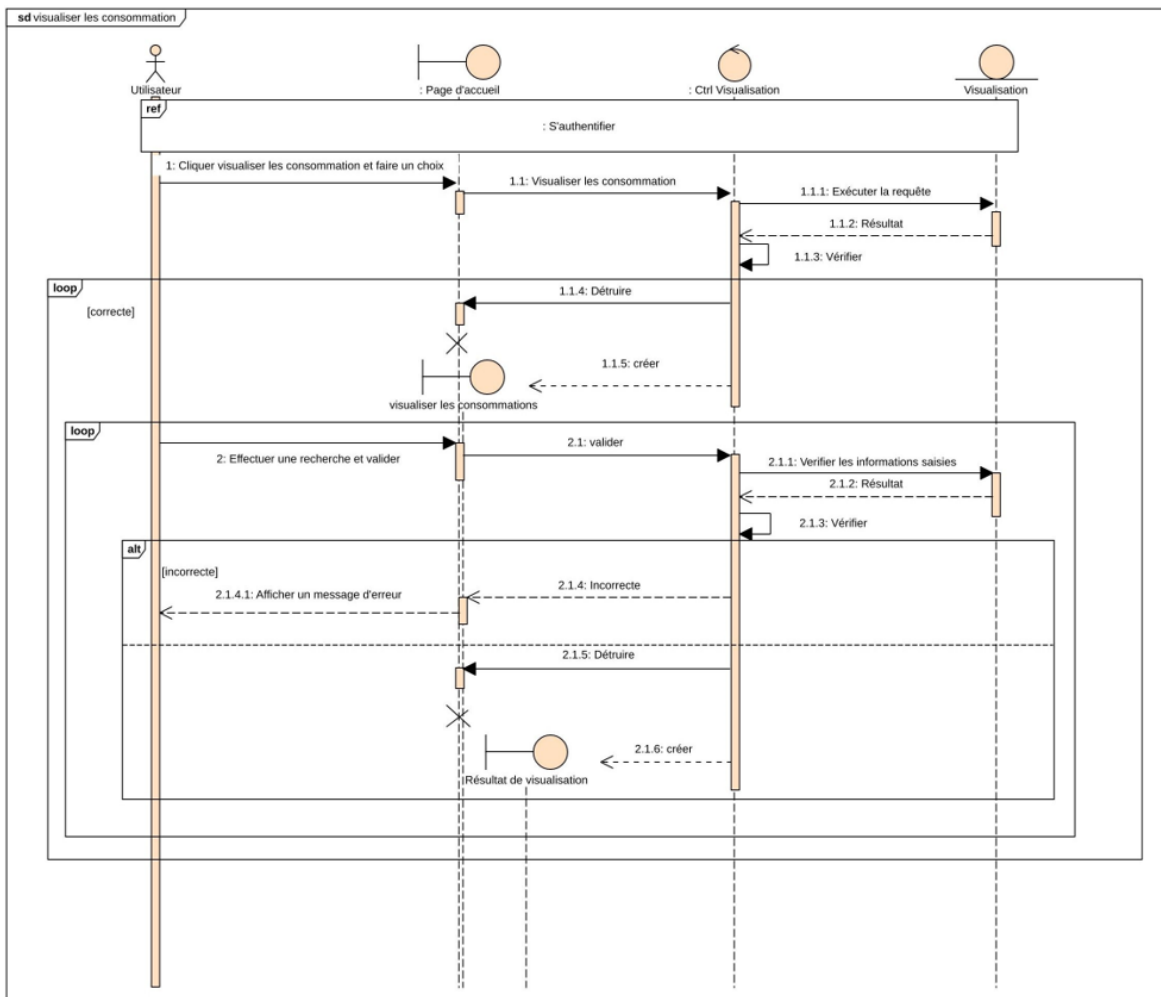


FIGURE 3.10 – Diagramme de séquence détaillé du cas d’utilisation «Visualiser les consommations»

### 3.14.3 Diagramme de séquence détaillé du cas d’utilisation ”Prédire les consommations ”

Le diagramme de séquence détaillé du cas d’utilisation ” Prédire les consommations ” illustre la séquence d’actions et d’échanges de messages entre les acteurs et les composants du système lorsqu’un utilisateur demande une prédiction des consommations . Ce diagramme met en évidence les étapes clés, telles que la saisie des données de vente, la transmission des données au composant de prédiction, le traitement des données, la génération de la prédiction et la présentation des résultats à l’utilisateur. Il offre une représentation visuelle précise de la séquence chronologique des interactions, permettant de comprendre comment le système utilise les données disponibles pour effectuer une prédiction précise des consommations .

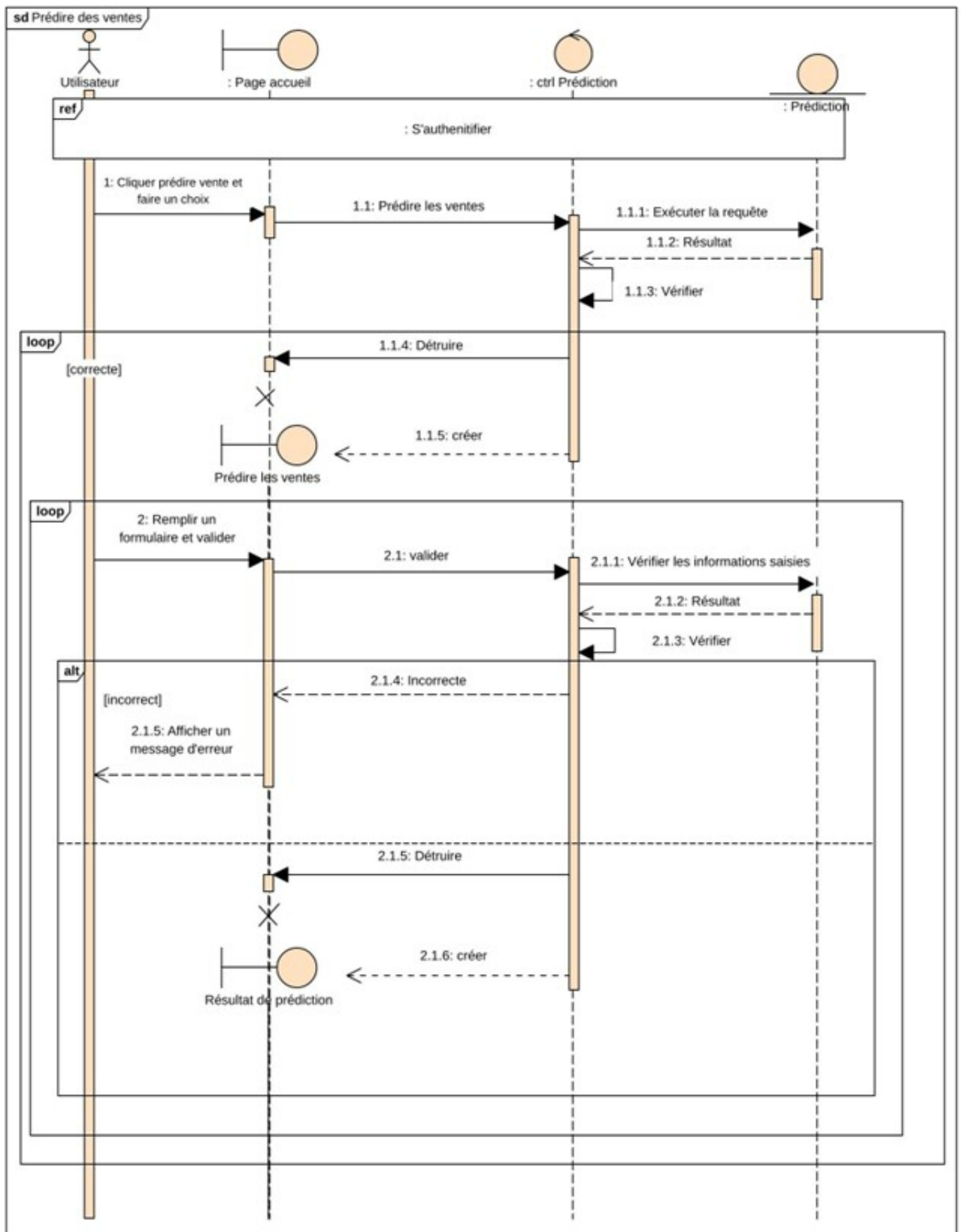


FIGURE 3.11 – Diagramme de séquence détaillé du cas d'utilisation «Prédire les consommations»

#### **3.14.4 Diagramme de séquence détaillé du cas d'utilisation ” Comparaison des Consommations prédite ”**

Le diagramme de séquence détaillé du cas d'utilisation ” Comparaison des Consommations prédite ” représente la séquence d’actions et d’échanges de messages entre les acteurs et les composants du système lorsqu’un utilisateur souhaite comparer les données de consommations . Ce diagramme met en évidence les étapes clés, telles que la sélection des critères de comparaison, la récupération des données de consommations correspondantes, le traitement des données, la génération des résultats de comparaison et la présentation des résultats à l’utilisateur. Il offre une représentation visuelle précise de la séquence chronologique des interactions, permettant de comprendre comment le système répond aux requêtes de comparaison de l’utilisateur en fournissant des informations pertinentes et significatives sur les ventes, facilitant ainsi l’analyse comparative des données de consommation .

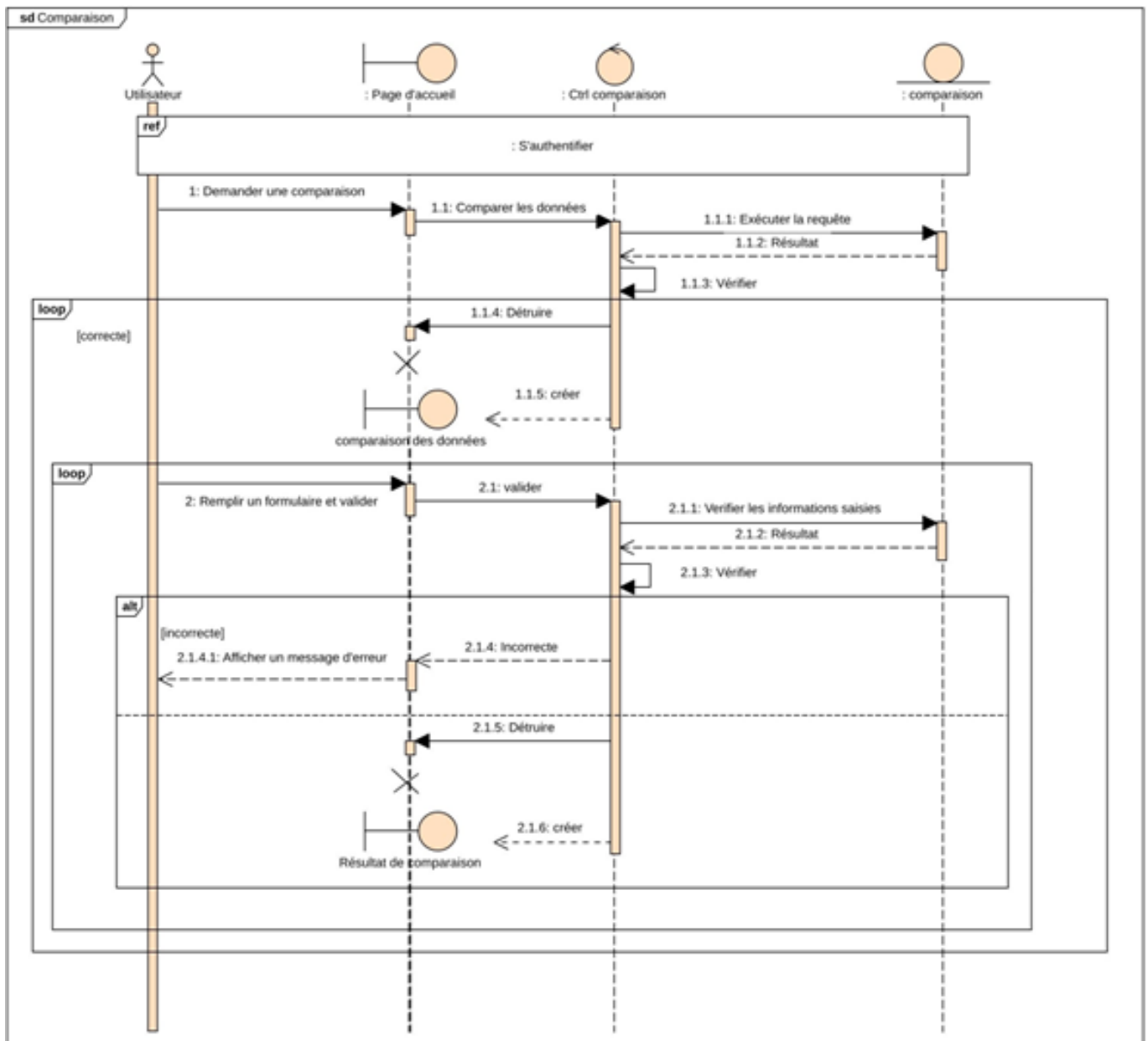


FIGURE 3.12 – Diagramme de séquence détaillé du cas d'utilisation «Comparaison »

### 3.15 Diagramme de navigation

Le diagramme de navigation est un outil utilisé dans la conception d'interfaces utilisateur pour représenter la structure et le flux de navigation entre les différentes pages d'une application. Ces diagrammes sont utiles pour comprendre, planifier, améliorer et documenter la structure de navigation d'une application, ce qui contribue à offrir une expérience utilisateur cohérente.

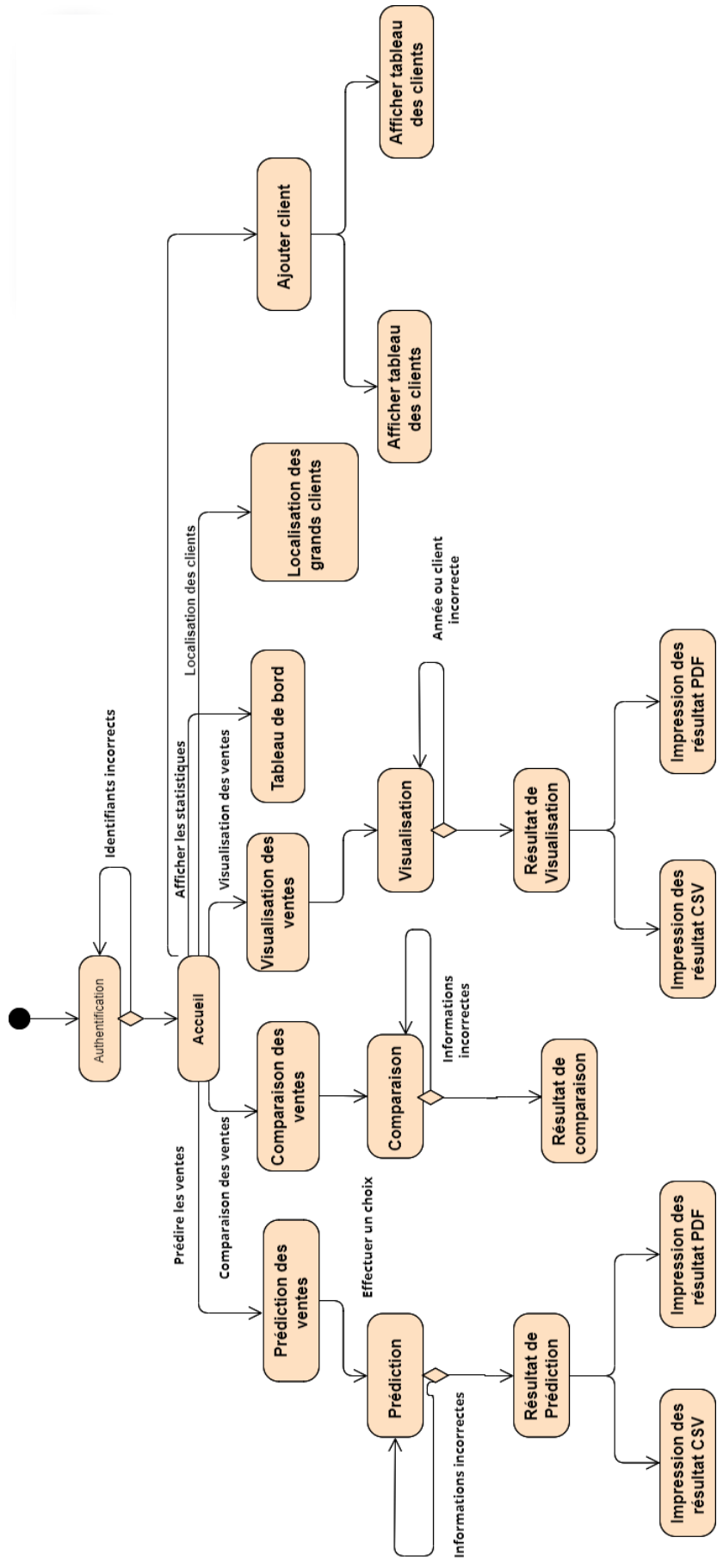


FIGURE 3.13 – Diagramme de Navigation

### 3.16 Diagrammes de Classe

Le diagramme de classe est un outil utilisé en génie logiciel pour représenter les classes, les objets, les attributs et les relations entre eux dans un système logiciel. Il permet de visualiser l'organisation et les interactions des objets dans un système, ainsi que les attributs et les méthodes qui leur sont associés.[23] De plus, il sert de base pour la conception et la mise en œuvre du code source, en aidant à la modélisation des classes, des relations et de la logique de programmation.

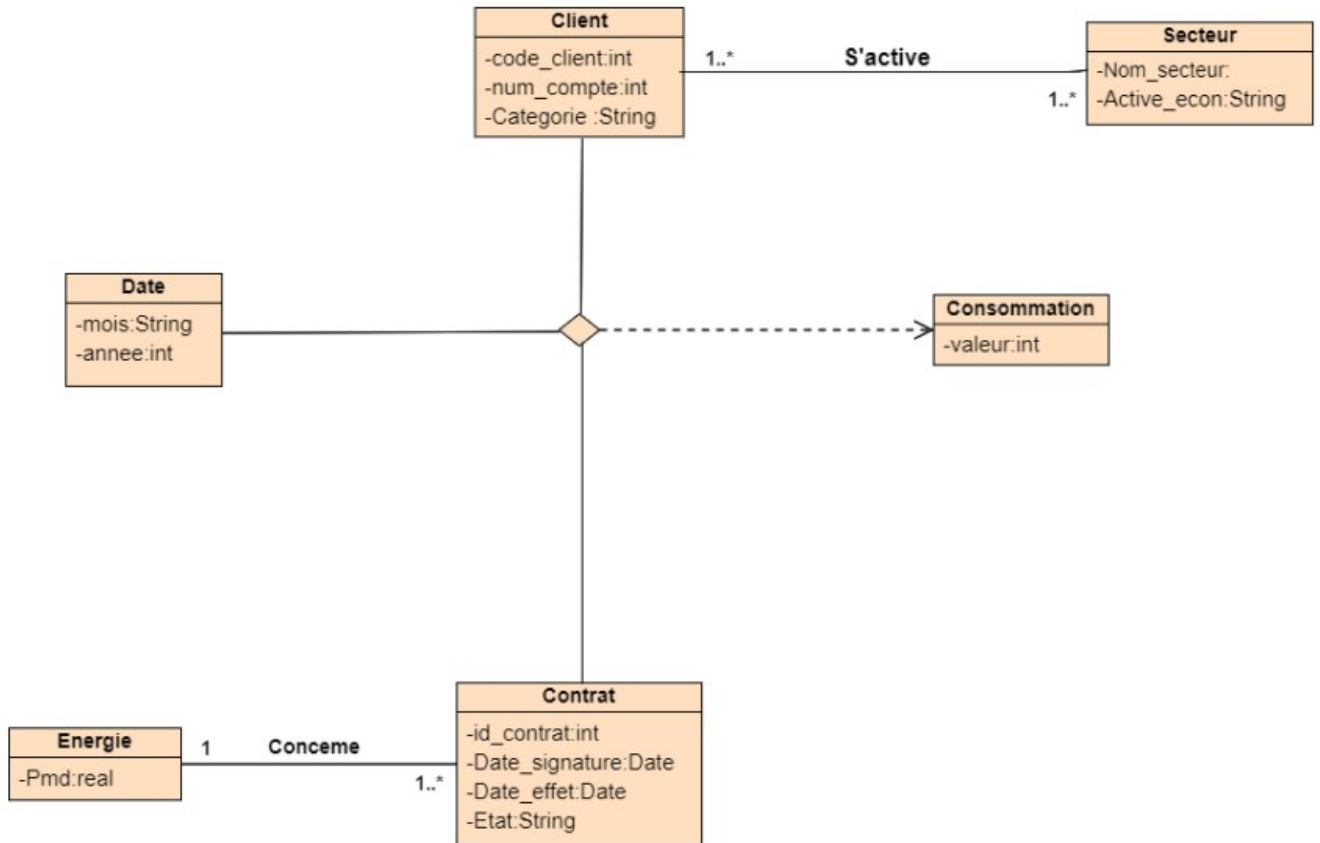


FIGURE 3.14 – Diagramme de Classe



Pour des raisons de confidentialité, nous avons reçu uniquement les données et les informations nécessaires pour notre stage. Ainsi, nous nous sommes uniquement appuyés sur les informations fournies, lesquelles sont représentées dans le diagramme présenté dans la figure suivante :

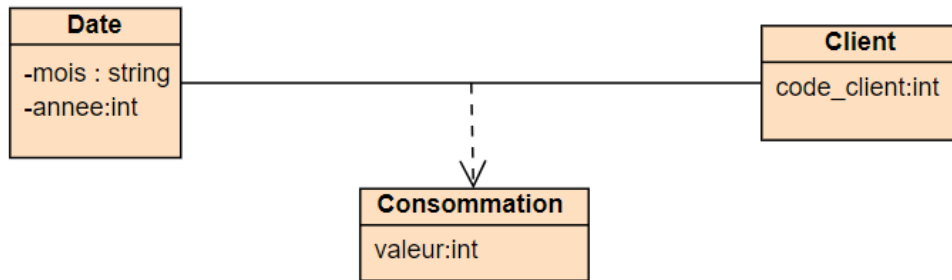


FIGURE 3.15 – Diagramme de Classe

### 3.16.1 Passage Relationnel

À partir du diagramme de classes présenté ci-dessus, le modèle relationnel correspondant est défini comme suit :

- Date(mois,annee)

- Client(code\_client)

code\_client : int (Clé primaire, Référencé par la table **Consommation**)

- Consommation(code\_client, mois, annee, valeur)

## 3.17 Conclusion

Ce chapitre a été crucial pour identifier les principales fonctionnalités de mon système et pour appliquer la méthode Scrum à mon projet. Il a également offert une présentation détaillée des différentes fonctionnalités de mon système à travers des diagrammes de séquence détaillés, ainsi que la conception de diagrammes de navigation et de classes. En somme, ce chapitre a préparé le terrain pour l'étape de réalisation de mon système, qui sera abordée dans le prochain chapitre.

# Chapitre 4

## Réalisation

### 4.1 Introduction

Dans ce dernier chapitre, je présenterai l'ensemble du travail réalisé. Tout d'abord, je décrirai l'environnement matériel et logiciel utilisé pour le développement de l'application, en mettant l'accent sur les technologies et les langages de programmation employés. Ensuite, je détaillerai les différentes étapes suivies pour modéliser une prévision à l'aide de techniques d'apprentissage automatique et d'apprentissage profond. Enfin, je conclurai ce chapitre par un aperçu global du projet, accompagné de captures d'écran illustrant le fonctionnement de l'application. Ces captures d'écran offriront une visualisation concrète de l'interface et des diverses fonctionnalités proposées par mon application.

### 4.2 Environnement de développement de l'application

#### 4.2.1 Architectures

Streamlit offre un moyen simple et intuitif de créer des applications Web interactives à l'aide de Python. Les applications peuvent être construites autour du framework Streamlit selon les principes suivants :

- **Architecture basée sur les pages** : Streamlit suit une approche basée sur les pages, où chaque page représente une vue ou une fonctionnalité distincte de l'application. Le code peut être organisé en créant des fichiers séparés pour chaque page ou en utilisant des fonctions dédiées dans le fichier principal [20].
- **Gestion des dépendances** : Streamlit permet d'importer des bibliothèques Python et de les utiliser dans l'application. Les dépendances peuvent être organisées selon les besoins spécifiques de chaque page ou fonction. Il est généralement recommandé de gérer les dépendances dans des fichiers de configuration (comme 'requirements.txt') pour faciliter la maintenance et le déploiement des applications[20].

L'utilisation de Streamlit comme framework simplifie le développement d'applications Web interactives car il prend en charge de nombreux aspects de l'interface utilisateur et de l'interaction des données. On peut profiter de la simplicité de Streamlit tout en organisant l'application de manière modulaire en fonction des fonctionnalités spécifiques souhaitées.[20]

### 4.2.2 Matériel

- **Type de la machine** : pc Acer
- **Processeur** : 11th Gen Intel(R) Core(TM) i5-1135G7 @ 2.40GHz 2.42 GHz
- **RAM** :8 Go.
- **Système d'exploitation** : Windows 11 (64 bits).

### 4.2.3 Outils et logiciels utilisés

Dans cette section, je vais présenter une gamme variée d'outils et de logiciels que nous avons utilisés pour accomplir nos tâches :

(a) **Visual Studio Code (VS Code)** :

Vs code est un éditeur de code source léger, open-source et multiplateforme développé par Micro-soft.

Il est conçu pour être rapide, facile à utiliser et extensible, offrant des fonctionnalités avancées pour les développeurs tels que la coloration syntaxique, la complétion de code, le débogage, l'intégration de Git et d'autres outils de gestion de version, ainsi que des extensions pour prendre en charge différents langages de programmation et Framework.

VS Code est disponible sur Windows, MacOS et Linux et est devenu l'un des éditeurs de code les plus populaires et les plus utilisés dans la communauté des développeurs [25].

(b) **Annaconda** :

**Type** : Distribution de logiciels pour l'analyse de données.

**Date de création** : La première version d'Anaconda est sortie en 2012.

**Développeur** : Anaconda, Inc. Environnement :Multiplateforme . Langage de **développement** : Python.

**Langues** : Principalement anglais, mais la communauté propose également des traductions dans d'autres langues.

**Licence** : Anaconda est basé sur des logiciels open-source, et la plupart des packages inclus sont également open-source[1].

(c) **Visual paradigm :**

Visual Paradigm est un logiciel de modélisation visuelle et de conception de logiciels [24].

**Type :** Logiciel de modélisation et de conception de logiciels.

**Date de création :** Visual Paradigm a été fondé en 1994.

**Développeur :** Visual Paradigm International Ltd.

**Environnement :** Multiplateforme .

**Langage de développement :** Java.

**Langues :** Disponible dans de nombreuses langues, dont l’anglais, le français, l’allemand, le chinois, etc.

#### 4.2.4 Langages utilisés

- **Python :** Python est le langage de programmation open source le plus couramment utilisé par les informaticiens. Parmi ses nombreuses fonctionnalités, Python permet aux développeurs de se concentrer sur ce qu’ils font, plutôt que sur la manière de le faire. Il libère les développeurs des contraintes formelles des langages plus anciens, rendant ainsi le développement de code plus rapide que dans d’autres langages [15].
- **CSS :** Les astuces CSS, également connues sous le nom de ”CSS tricks” ou ”CSS workarounds”, désignent des techniques non standard ou des contournements utilisés pour résoudre des problèmes spécifiques liés à la mise en page et à l’apparence des éléments sur une page web en utilisant CSS (Cascading Style Sheets) [3].

#### 4.2.5 Frameworks

- **Streamlit :** Streamlit est une bibliothèque Python open source créée en 2018, permettant une grande flexibilité pour répondre aux besoins de chacun grâce à sa nature réutilisable. L’open source favorise la production collaborative, améliorant ainsi la qualité des logiciels. Streamlit permet de créer des applications en écrivant simplement du code Python, offrant une méthode pratique et accessible pour ceux qui connaissent Python et souhaitent créer des applications Web. C’est un outil rapide et simple pour intégrer la visualisation de données dans les applications[20, 15] .

L’intégration de modèles d’apprentissage automatique dans l’interface est également facile. Grâce à l’application, n’importe qui peut prédire et tester ses modèles avec des données en collaboration avec des clients ou des collaborateurs. Un des principaux atouts de Streamlit est que son interface de programmation d’application (API) nécessite très peu de commandes, simplifiant ainsi son utilisation.

## 4.2.6 Bibliothèques Python utilisées

- **Pandas** : Pandas est une bibliothèque Python qui facilite la manipulation des données à analyser. Elle permet de manipuler des tableaux de données avec des étiquettes de variables (colonnes) et d'individus (lignes), appelés DataFrames [14].
- **NumPy** : NumPy est une bibliothèque Python très populaire, principalement utilisée pour effectuer des calculs mathématiques et scientifiques. Elle offre de nombreuses fonctionnalités et outils utiles pour les projets de data science [13].
- **Matplotlib** : Matplotlib est une bibliothèque Python open source permettant de créer des visualisations de données [8].
- **Keras** : Keras est une API de réseau de neurones écrite en Python. Il s'agit d'une bibliothèque open source exécutée par-dessus des frameworks tels que Theano et TensorFlow. Conçue pour être modulaire, rapide et simple d'utilisation, elle offre une façon intuitive de créer des modèles de deep learning. Aujourd'hui, Keras est l'une des API de réseaux de neurones les plus utilisées pour le développement et le testing de réseaux de neurones, permettant de créer facilement des « layers » pour les réseaux neuronaux ou de mettre en place des architectures complexes [7].
- **TensorFlow** : TensorFlow est une bibliothèque open source compatible avec Python pour le calcul numérique, rendant l'apprentissage automatique et le développement de réseaux neuronaux plus rapides et plus faciles [22].
- **Scikit-Learn** : Aussi appelée sklearn, cette bibliothèque est la plus puissante et la plus robuste pour le machine learning en Python. Elle fournit une sélection d'outils efficaces pour l'apprentissage automatique et la modélisation statistique, notamment la classification, la régression et le clustering via une interface cohérente en Python [18].
- **Folium** : Folium est une bibliothèque open source complémentaire de Python qui permet d'étendre ses fonctionnalités. C'est un puissant outil de visualisation permettant la conception de cartes interactives [4].
- **Transformers (Hugging Face)** : La bibliothèque Transformers de Hugging Face est une bibliothèque open source en Python permettant de travailler avec des modèles basés sur l'architecture des transformers pour des séries temporelles. Elle est utilisée pour des tâches variées telles que la prévision des valeurs futures, la détection d'anomalies, et la classification de séquences temporelles. Transformers facilite l'utilisation des modèles pré-entraînés et offre des outils puissants pour ajuster ces modèles sur des

jeux de données temporelles spécifiques [16].

### 4.2.7 Modèle utilisé

- **BERT (Bidirectional Encoder Representations from Transformers)** : BERT est un modèle développé par Google qui utilise l'architecture des Transformers, adapté pour les séries temporelles. Il est conçu pour comprendre le contexte des valeurs dans les séquences temporelles en considérant les dépendances des deux côtés de chaque point de données. BERT est utilisé pour une variété de tâches temporelles, notamment la prévision de valeurs futures, la détection d'anomalies et la classification de séquences. Les implémentations de BERT sont souvent intégrées avec la bibliothèque Transformers de Hugging Face pour faciliter leur utilisation et leur ajustement sur des tâches spécifiques de séries temporelles [16].
- **ADEM** : ADEM (Automatic Dialogue Evaluation Model) est un modèle d'évaluation automatique de dialogues utilisé dans le traitement du langage naturel. Il utilise des techniques de deep learning pour comparer les réponses générées par des systèmes de dialogue avec des réponses humaines de référence, fournissant ainsi une évaluation quantitative de la qualité des dialogues. Cette approche peut être adaptée à d'autres applications telles que la prévision des séries temporelles, où elle permet de mesurer objectivement la précision des prédictions des modèles en les comparant avec les valeurs réelles observées [16].

## 4.3 Modélisation des méthodes de prédiction utilisé

Notre objectif principal avec cette application est de développer un système avancé pour prédire les ventes, c'est-à-dire la consommation électrique des clients. Pour ce faire, nous explorons trois méthodes clés : les transformers, discutés dans le premier chapitre, ainsi que les modèles LSTM et la régression linéaire, également abordés dans ce même chapitre. Nous utilisons ces trois approches pour anticiper les tendances des séries temporelles. Dans les prochains paragraphes, nous examinerons en détail ces trois méthodes que nous avons appliquées dans le cadre de cette étude.

### 4.3.1 Méthode LSTM

La méthode LSTM (Long Short-Term Memory) est particulièrement bien adaptée pour modéliser les séries temporelles et prédire des valeurs futures, comme

la consommation électrique. [29]Voici comment appliquer la méthode LSTM pour prédire la consommation future de données temporelles :

### Étapes de la modélisation LSTM

- (a) **Préparation des données :**
  - **Collecte des données :** Obtenez les données historiques de consommation électrique, qui peuvent être sous forme de séries temporelles avec des valeurs de consommation enregistrées à des intervalles réguliers (par exemple, quotidiennement, hebdomadairement, mensuellement).
  - **Nettoyage des données :** Traitez les valeurs manquantes, les anomalies et les données incohérentes.
  - **Normalisation :** Échellez les données pour que les valeurs soient comprises entre 0 et 1 ou -1 et 1, ce qui aide le modèle à converger plus rapidement.
- (b) **Diviser les données en séquences :** Créez des séquences de données historiques pour entraîner le modèle LSTM. Une séquence pourrait inclure, par exemple, les données des 30 derniers jours pour prédire la consommation du jour suivant.
- (c) **Définir l'architecture du modèle LSTM :** Créez un modèle LSTM en spécifiant le nombre de couches LSTM, le nombre de neurones par couche, et les fonctions d'activation.
- (d) **Entraînement du modèle :** Utilisez les données d'entraînement pour ajuster les paramètres du modèle LSTM
- (e) **Interprétation des résultats :** Visualisez les prédictions du modèle par rapport aux valeurs réelles pour évaluer la performance. Utilisez des métriques de performance comme la racine carrée de l'erreur quadratique moyenne (RMSE) pour quantifier la précision des prédictions.

### 4.3.2 Méthode régression linéaire :

La régression linéaire est une méthode statistique simple et efficace pour modéliser la relation entre une variable dépendante et une ou plusieurs variables indépendantes. Elle est souvent utilisée comme point de départ pour les tâches de prédiction en raison de sa simplicité et de son interprétabilité. Voici comment appliquer la régression linéaire pour prédire la consommation future de données temporelles :

### Étapes de la modélisation de la régression linéaire

- (a) **Préparation des données :**
  - **Collecte des données :** Obtenez les données historiques de consommation électrique. Ces données doivent inclure des valeurs de consommation enregistrées à des intervalles réguliers.
  - **Nettoyage des données :** Traitez les valeurs manquantes, les anomalies et les incohérences.

- **Feature Engineering** : Créez des caractéristiques supplémentaires à partir des données de consommation historique, telles que les décalages temporels (lag features).
- (b) **Diviser les données en ensembles d'entraînement et de test** : Séparez les données en ensembles d'entraînement et de test pour évaluer les performances du modèle.
- (c) **Définir et entraîner le modèle de régression linéaire** : Utilisez la bibliothèque scikit-learn pour créer et entraîner le modèle de régression linéaire.
- (d) **Évaluation et prédiction** : Évaluez le modèle sur l'ensemble de test pour vérifier sa capacité de généralisation. Utilisez le modèle entraîné pour effectuer des prédictions sur de nouvelles données.
- (e) **Interprétation des résultats** : Visualisez les prédictions du modèle par rapport aux valeurs réelles pour évaluer la performance. Utilisez des métriques de performance comme la racine carrée de l'erreur quadratique moyenne (RMSE) pour quantifier la précision des prédictions.



### 4.3.3 Méthode transformers

Les transformers sont une classe de modèles d'apprentissage profond qui ont révolutionné de nombreux domaines de l'apprentissage automatique, y compris le traitement du langage naturel, la vision par ordinateur et les séries temporelles.[35] Leur architecture principale, comme le Transformer original et ses variantes telles que BERT, GPT, et T5, est basée sur des mécanismes d'attention qui permettent aux modèles de capturer des dépendances à long terme dans les données séquentielles.

#### Étapes de la modélisation avec des Transformers

- (a) **Préparation des données :**
  - **Formatage des données :** Les données doivent être formatées selon les spécifications du modèle Transformer utilisé. Cela peut inclure la tokenization des séquences et la création de masques d'attention.
  - **Traitement des séquences :** Les séquences doivent être préparées sous forme de tenseurs pour être utilisées en entrée du modèle.
- (b) **Chargement du modèle pré-entraîné :**
  - Utilisez des bibliothèques telles que Hugging Face Transformers pour charger un modèle pré-entraîné adapté à votre tâche spécifique, par exemple, BERT, GPT, ou T5.
- (c) **Fine-tuning ou Feature Extraction :**
  - Selon le cas d'utilisation, vous pouvez choisir de fine-tuner le modèle sur vos propres données ou d'utiliser le modèle pré-entraîné comme extracteur de caractéristiques pour une tâche spécifique.
- (d) **Entraînement et évaluation :**
  - Entraînez le modèle sur vos données d'entraînement et évaluez sa performance sur un ensemble de validation ou de test.
  - Utilisez des métriques appropriées pour évaluer la qualité des prédictions, telles que la précision, le rappel, ou les métriques spécifiques à la tâche.
- (e) **Interprétation des résultats :**
  - Analysez les prédictions du modèle pour comprendre comment il prend des décisions.
  - Utilisez des techniques d'interprétabilité telles que l'attention pour identifier les parties importantes des séquences d'entrée qui influencent les prédictions du modèle.

## 4.4 Présentation de l'application

### 4.4.1 Application Web

L'architecture de cette application web repose sur le framework Streamlit, un outil puissant qui fusionne les aspects client et serveur en une seule entité. Avec

Streamlit, on peut développer des applications web interactives en Python sans recourir à un serveur externe.

Dans ce modèle, le navigateur du client envoie une requête à l'application Streamlit, qui exécute le code Python correspondant. Streamlit traite ensuite la requête et génère une réponse, présentée sous la forme d'une interface utilisateur interactive dans le navigateur du client.

Pour la partie client de l'interface graphique, j'utilise les fonctionnalités intégrées de Streamlit, qui proposent une gamme d'éléments interactifs tels que des curseurs, des menus déroulants et des boutons. Ces éléments permettent aux utilisateurs de manipuler les données et d'interagir aisément avec l'application.

En ce qui concerne le développement du côté serveur, Streamlit simplifie considérablement le processus en adoptant une approche de développement tout-en-un. Ainsi, je n'ai pas besoin de recourir à un framework web externe tel que Flask. Le code Python de l'application Streamlit sert à la fois de serveur et de modèle, intégrant directement la logique métier et les opérations de traitement des données.

En résumé, l'architecture de notre application web développée avec Streamlit offre une structure simplifiée où le navigateur du client communique directement avec l'application Streamlit. Cette dernière gère les requêtes et renvoie les réponses correspondantes. L'interface utilisateur est créée à l'aide des fonctionnalités intégrées de Streamlit, tandis que la logique métier et le traitement des données sont encapsulés dans le code Python de l'application.

## 4.4.2 Interfaces Graphiques

Nous allons maintenant présenter différentes interfaces de notre application et expliquer leur utilité respective :

### 4.4.2.1 Interface d'Authentification

L'interface d'authentification, illustrée dans la figure 4.1, permet aux utilisateurs de saisir leur nom d'utilisateur et leur mot de passe pour accéder aux fonctionnalités du système.



FIGURE 4.1 – Interface d'Authentification

#### 4.4.2.2 Interface Inscription

L'interface d'inscription, représentée dans la figure 4.2, est l'interface qui s'affiche lorsque les utilisateurs souhaitent créer un nouveau compte.



FIGURE 4.2 – Interface Inscription

#### 4.4.2.3 Interface du Tableau de Bord

L'interface de tableau de bord, représentée dans la figure 4.3, est la première interface qui apparaît après que l'utilisateur (administrateur) s'est connecté avec succès.

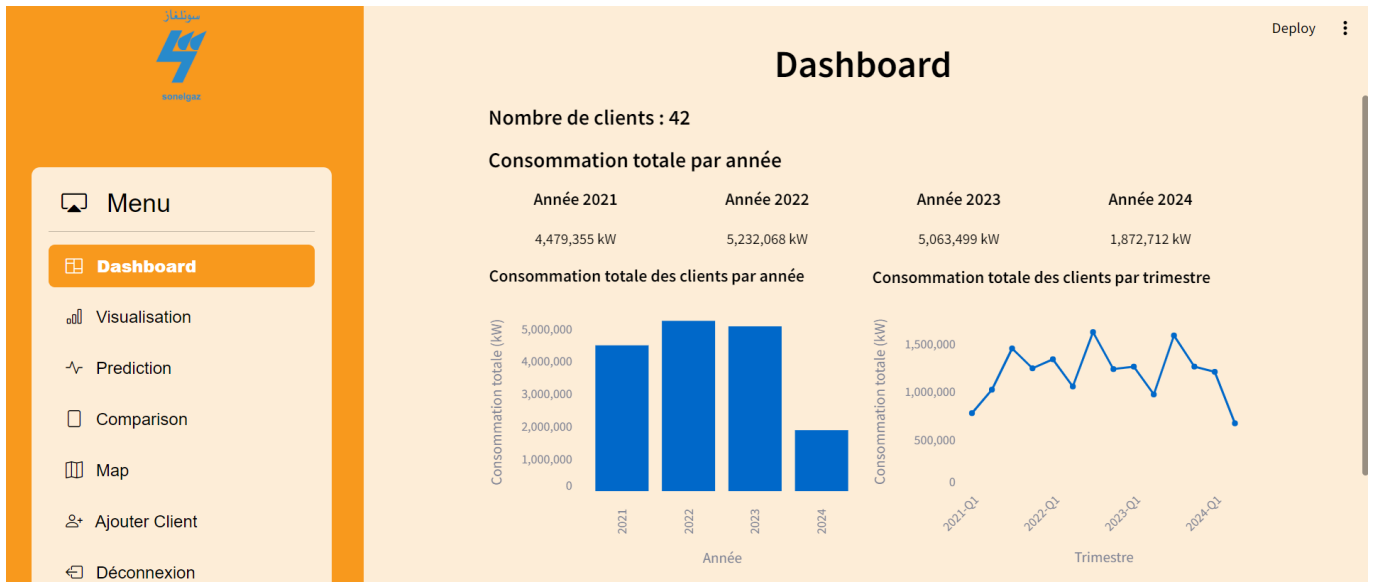


FIGURE 4.3 – Interface Tableau de Bord

#### 4.4.2.4 Interface de Visualisation Générale des Consommations

L'interface de visualisation générale des consommations, présentée dans les figures 4.4 et 4.5, offre aux utilisateurs une vue d'ensemble complète et intuitive de leurs données de vente.



FIGURE 4.4 – Interface de Visualisation Générale des Consommations Tableau



FIGURE 4.5 – Interface de Visualisation Générale des Consommations Graphe

#### 4.4.2.5 Interface Visualisation des Consommations par Client

L'interface de visualisation des consommations par client, illustrée dans les figures 4.6 et 4.7, offre aux utilisateurs une expérience interactive pour explorer les données de vente à travers des graphiques et des tableaux.

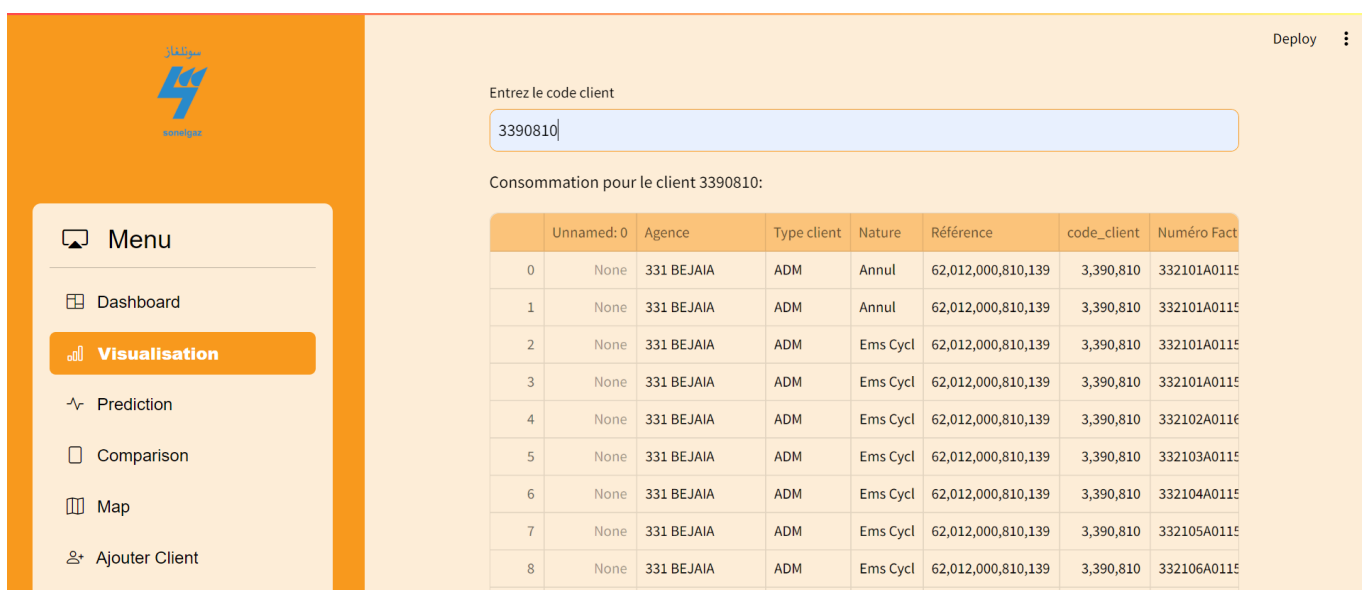


FIGURE 4.6 – Interface Visualisation des Consommations par Client - Tableau

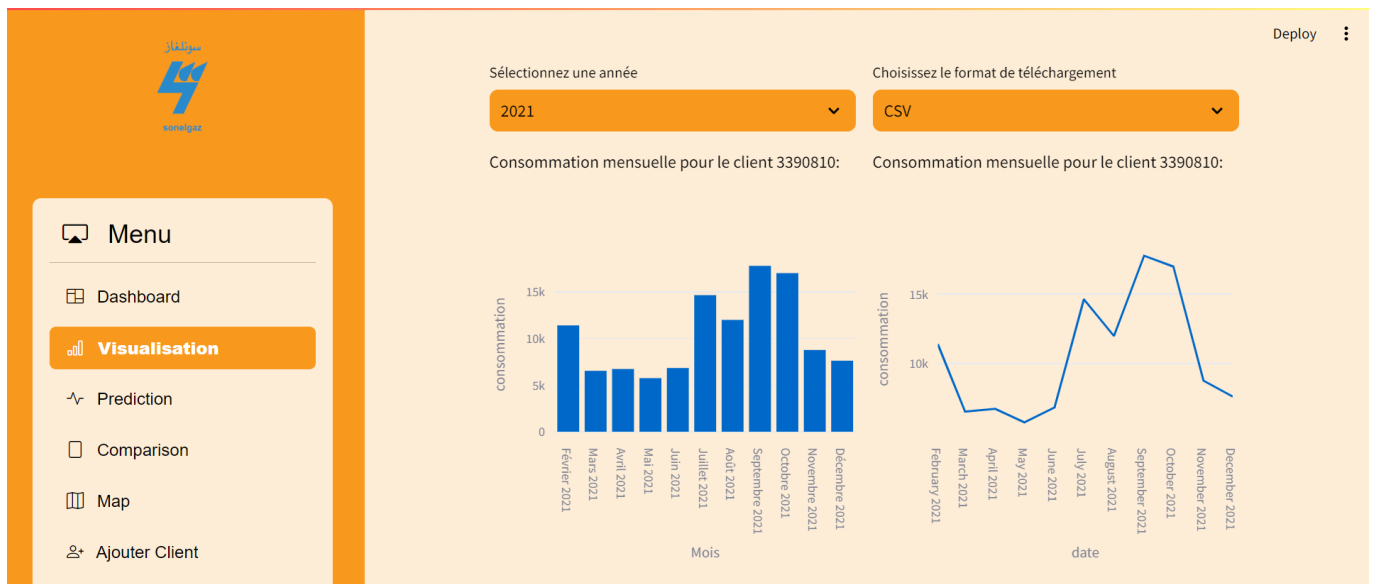


FIGURE 4.7 – Interface Visualisation des Consommations par Client - Graphe

#### 4.4.2.6 Interface Localisation des Clients

L'interface de visualisation de la localisation des grands clients, illustrée dans la figure 4.8, présente sur une carte géographique de la wilaya de Bejaia les emplacements de quelques clients importants.

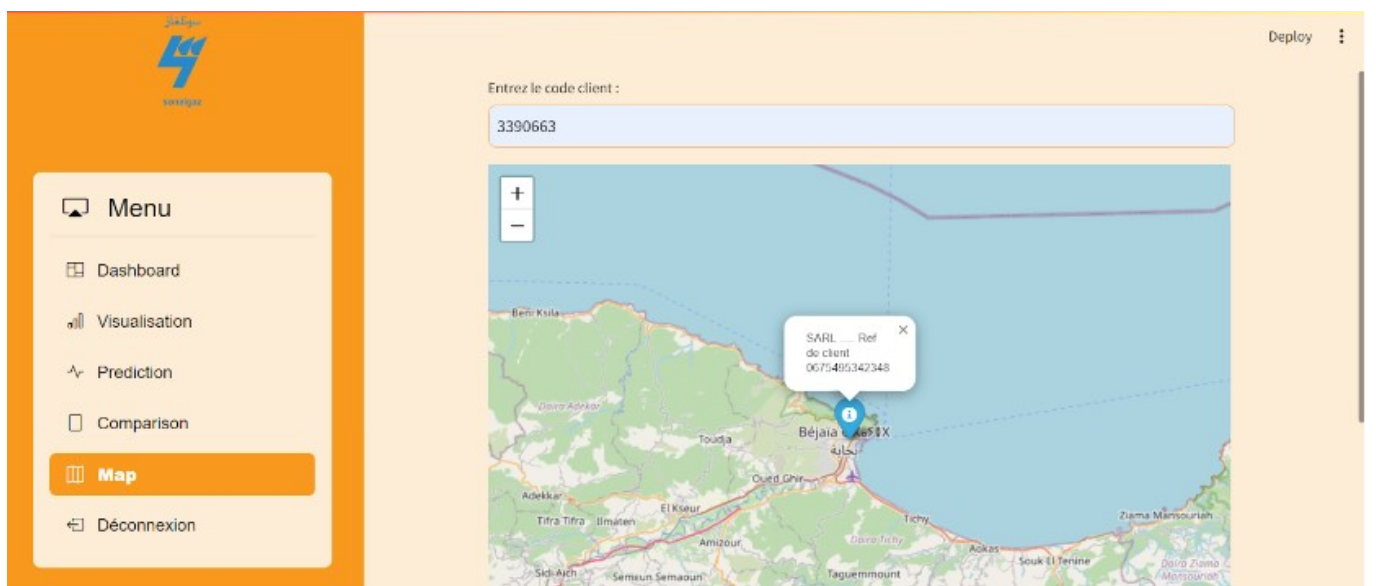


FIGURE 4.8 – Interface de Visualisation de la Localisation des Grands Clients

#### 4.4.2.7 Interface Prédiction des Consommations

L'interface de prédiction des consommations propose trois méthodes différentes pour anticiper les ventes à venir : la régression linéaire, le LSTM et les Transformers. Chacune de ces approches utilise les données historiques pour estimer les ventes futures. En offrant ces trois méthodes, l'interface permet aux utilisateurs

de choisir celle qui correspond le mieux à leurs besoins et à la nature de leurs données.

Grâce à cette interface, présentée dans les figures 4.9, 4.10, 4.11, et 4.12, l'administrateur a la possibilité d'afficher les prédictions de la consommation des clients. Pour cela, ils doivent simplement saisir le code de client concerné. Par exemple, si l'utilisateur saisit le code client 3390095 et choisit une méthode de prévision (par ex. régression linéaire), les prédictions correspondantes s'afficheront dans les figures suivantes. Il peut également imprimer ces données en format PDF ou CSV.

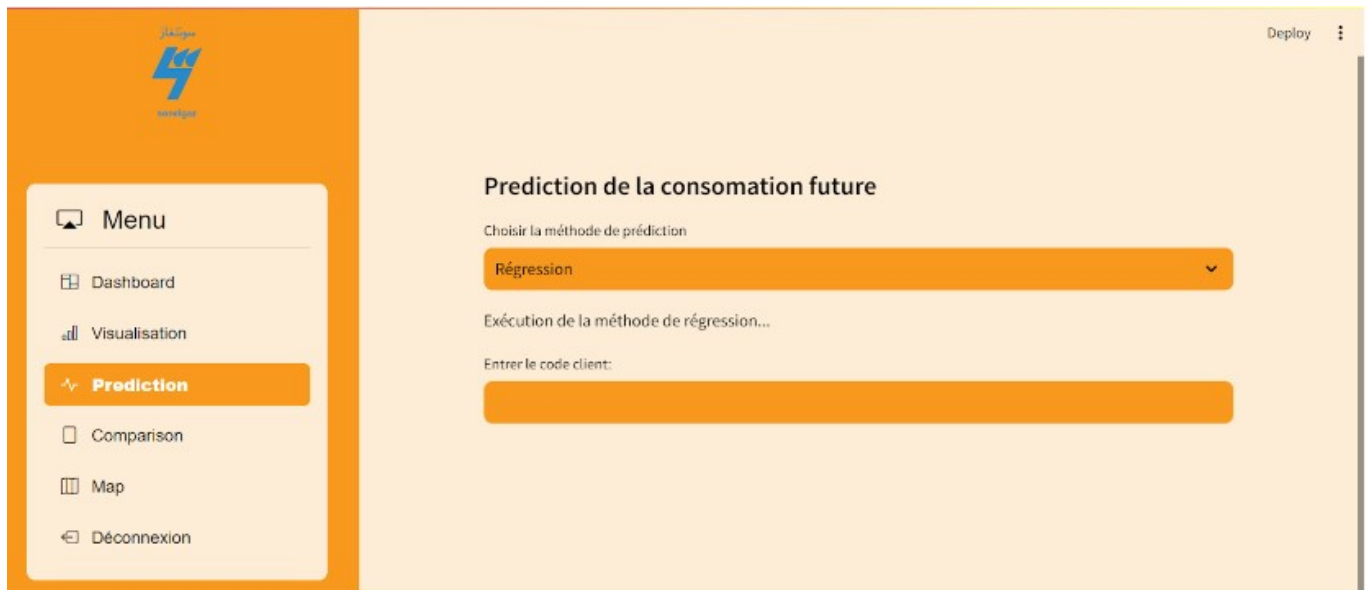


FIGURE 4.9 – Interface Prédiction des Consommations

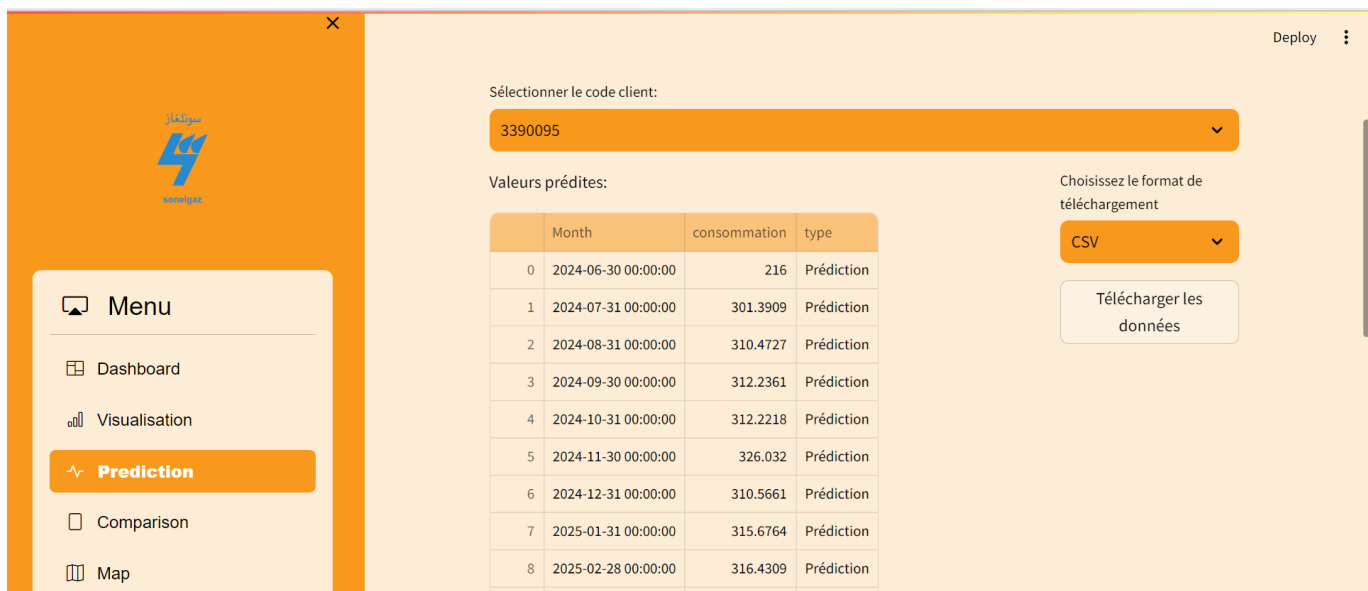


FIGURE 4.10 – Interface Prédiction avec Régression - Tableau



FIGURE 4.11 – Capture d'un Exemple de la Consommation Prédite en Format PDF



FIGURE 4.12 – Interface Prédiction avec Régression Linéaire - Graphe





FIGURE 4.13 – Interface Prédiction avec Transformers - Tableau



FIGURE 4.14 – Interface Prédiction avec Transformers - Graphe



FIGURE 4.15 – Interface Prédiction avec LSTM - Tableau

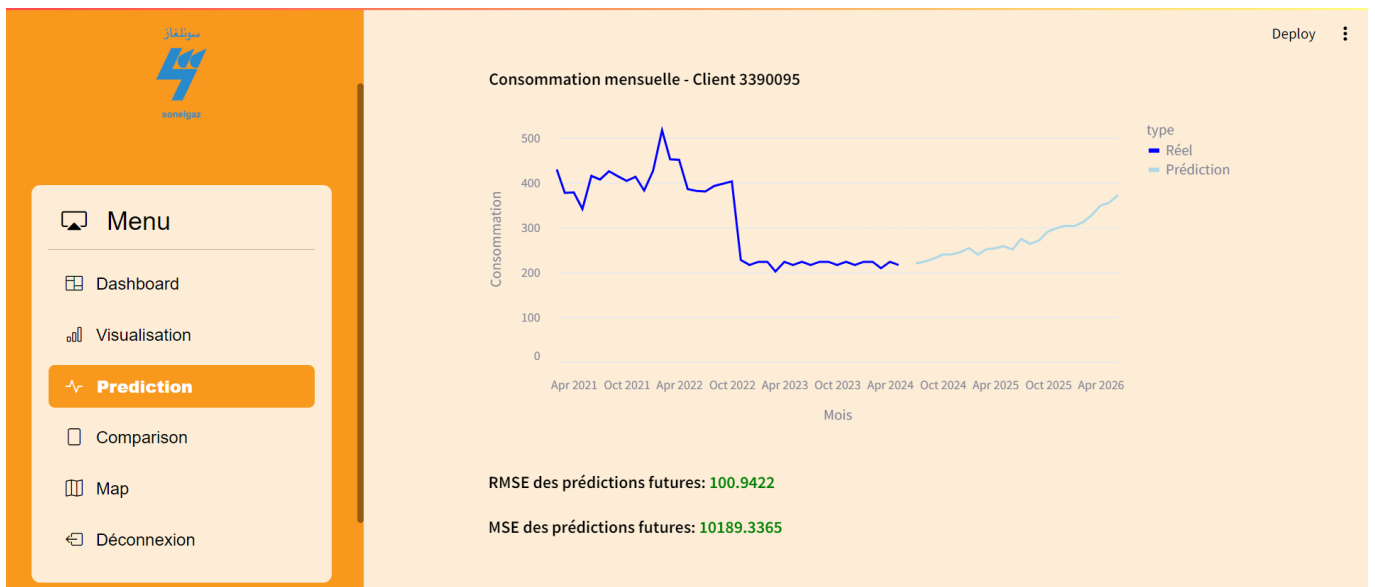


FIGURE 4.16 – Interface Prédiction avec LSTM - Graphe

#### 4.4.2.8 Interfaces de Comparaison

L'interface de comparaison, illustrée dans les figures 4.17, 4.18 et 4.19, offre à l'administrateur la possibilité de comparer les valeurs prédites d'un client avec les 3 méthodes.



FIGURE 4.17 – Interface de Comparaison de Prédiction par Client

	Unnamed: 0	Month	consommation	type	code_client	nom_client
24	0	2024-06-30 00:00:00	1,020.3952	Regression linéaire	3,390,801	None
25	1	2024-07-31 00:00:00	946.8073	Regression linéaire	3,390,801	None
26	2	2024-08-31 00:00:00	897.9083	Regression linéaire	3,390,801	None
27	3	2024-09-30 00:00:00	944.5059	Regression linéaire	3,390,801	None
28	4	2024-10-31 00:00:00	1,007.0251	Regression linéaire	3,390,801	None
29	5	2024-11-30 00:00:00	888.8026	Regression linéaire	3,390,801	None
30	6	2024-12-31 00:00:00	953.2653	Regression linéaire	3,390,801	None

FIGURE 4.18 – Interface de Comparaison de Prédiction par Client - Tableau

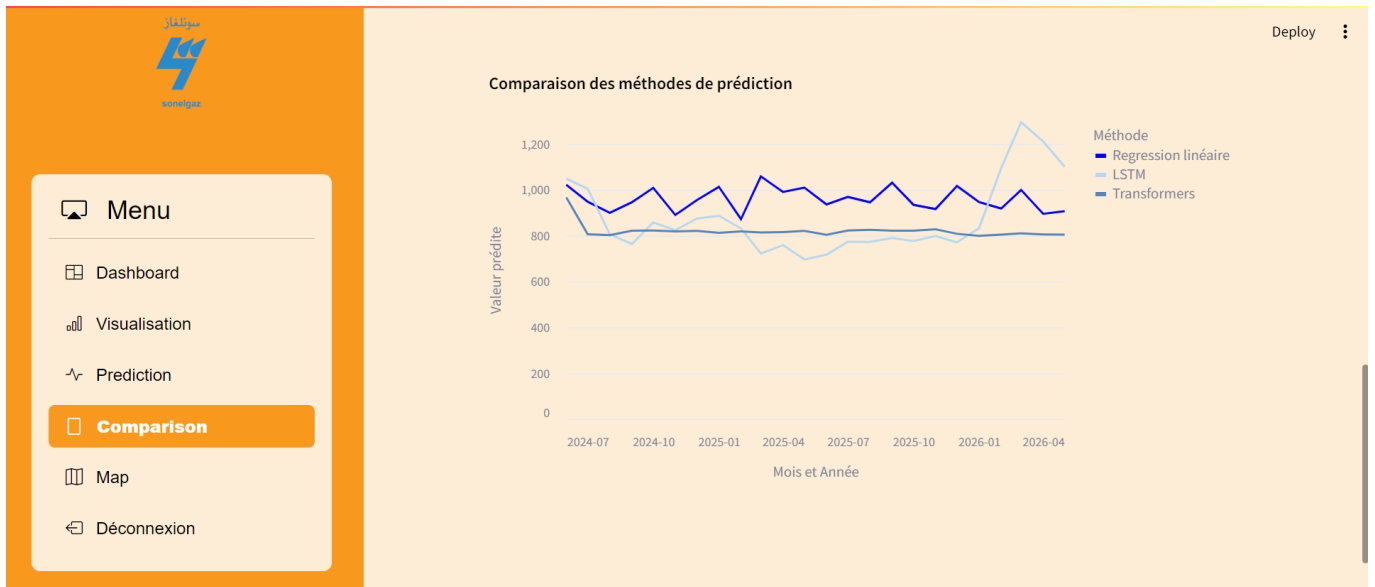


FIGURE 4.19 – Interface de Comparaison de Prédiction par Client - Graphe

#### 4.4.2.9 Interface d'Ajout et de Suppression d'un Client

Les interfaces d'ajout et de suppression d'un client sont illustrées dans la figure 4.20 . Elles permettent respectivement d'ajouter de nouveaux clients et de supprimer des clients existants.

**Ajouter un nouveau client**

Nom d'utilisateur  
admin

Mot de passe

Type d'utilisateur  
admin

Ajouter

FIGURE 4.20 – Interface d'Ajout d'un Client

**Liste des utilisateurs**

	Nom d'utilisateur	Mot de passe	Type d'utilisateur
0	admin	admin	admin
1	client1	client1	client
2	client11	client11	admin

**Supprimer un utilisateur**

Nom d'utilisateur à supprimer

Supprimer

FIGURE 4.21 – Interface de Suppression d'un Client

#### 4.4.2.10 Interface de Tableau de Bord de Client

Lorsqu'un client accède à l'interface, il est accueilli par son tableau de bord, comme illustré dans les figures 4.22, 4.23 et 4.24. Il peut choisir l'année souhaitée (par exemple, 2021) et un tableau de sa consommation ainsi qu'un graphique apparaîtront pour faciliter la visualisation de ses données.



FIGURE 4.22 – Interface de Tableau de Bord de Client - Tableau

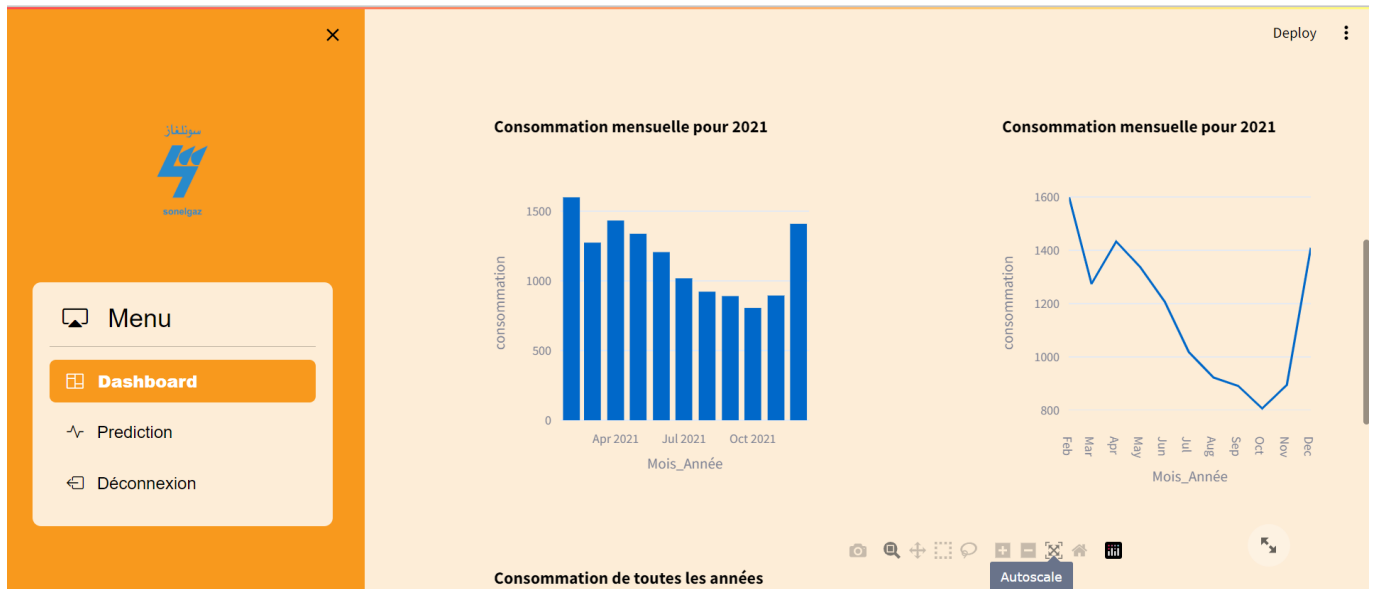


FIGURE 4.23 – Interface de Tableau de Bord de Client - Graphe

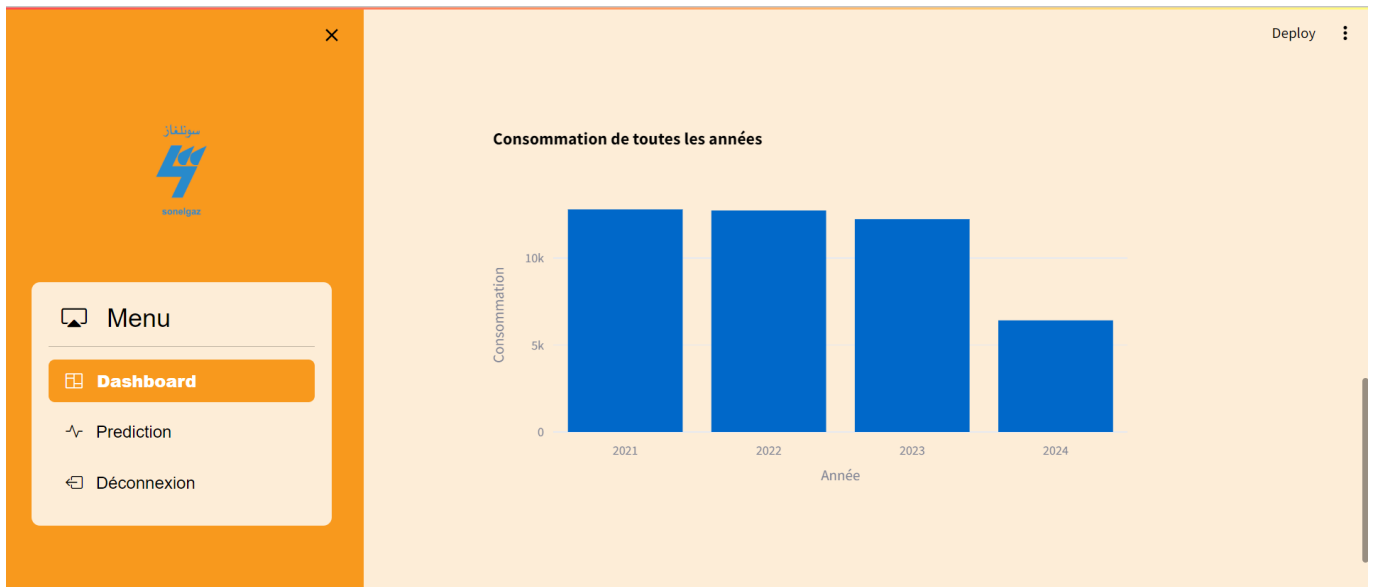


FIGURE 4.24 – Interface de Tableau de Bord de Client - Consommation durant les 4 ans

#### 4.4.2.11 Interface de Prédiction pour le Client

L'interface de prédiction pour le client est illustrée dans les figures 4.25, 4.26, 4.27 et 4.28. Elle permet aux clients de visualiser leurs consommations futures basées sur différentes méthodes de prédiction : régression linéaire, Transformers, et LSTM.

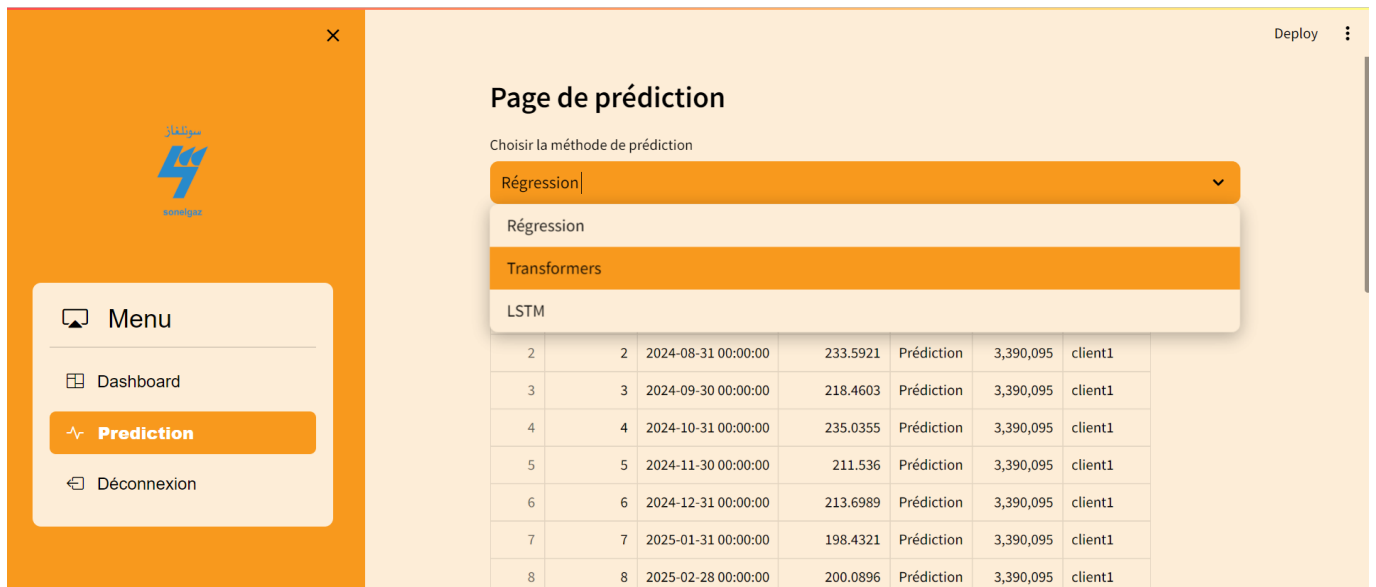


FIGURE 4.25 – Interface de Prédiction pour le Client - Tableau



FIGURE 4.26 – Interface de Prédiction avec Régression Linéaire - Graphe





FIGURE 4.27 – Interface de Prédiction avec Transformers - Graphe

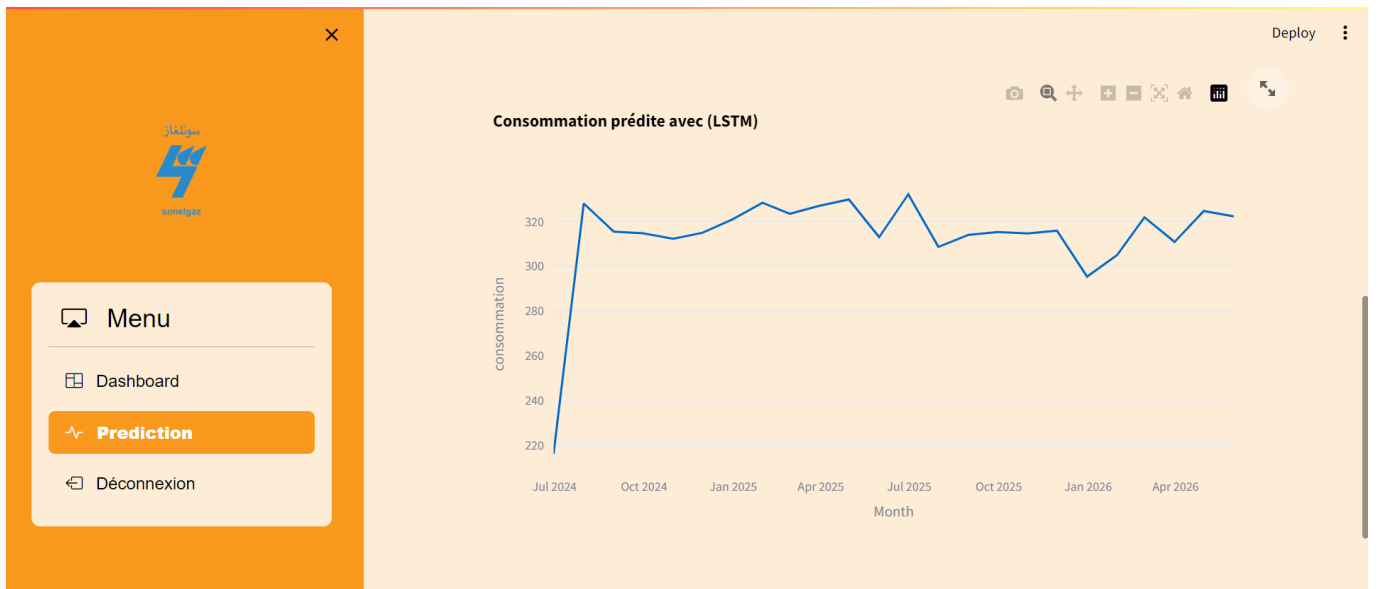


FIGURE 4.28 – Interface de Prédiction avec LSTM - Graphe

## 4.5 Comparaison entre les Méthodes de Régression Linéaire, LSTM et Transformers

En analysant les réalisations et les prédictions obtenues avec les méthodes de régression linéaire, LSTM et Transformers, nous avons observé des différences significatives dans leurs performances.

Pour commencer, en ce qui concerne les prédictions de la régression linéaire, nous avons constaté qu'elles ne sont pas aussi précises et fiables par rapport à celles de LSTM et de Transformers. Les prédictions de la régression linéaire ont montré une tendance à sous-estimer ou surestimer les valeurs réelles de la série temporelle. Cela peut être dû aux limitations de la régression linéaire dans la modélisation de structures plus complexes et dans la capture des dépendances à long terme.

En revanche, les prédictions de LSTM ont présenté des résultats plus satisfaisants et nettement supérieurs à ceux de la régression linéaire. LSTM, grâce à sa capacité à capturer les motifs temporels complexes et les dépendances à long terme, a pu mieux modéliser les variations et les tendances de la série temporelle. Cela a conduit à des prédictions plus précises et cohérentes par rapport à la régression linéaire.

Quant aux Transformers, leurs prédictions ont été nettement meilleures et plus professionnelles que celles obtenues par LSTM et la régression linéaire. Les Transformers utilisent un mécanisme d'attention qui leur permet de traiter des séquences de données en parallèle et de capturer les dépendances à différentes échelles temporelles de manière plus efficace. Contrairement aux LSTM, qui traitent les séquences de données de manière séquentielle et peuvent rencontrer des difficultés avec les dépendances à long terme, les Transformers peuvent pondérer l'importance des différentes parties de la séquence indépendamment de leur position, ce qui leur permet de mieux modéliser les caractéristiques complexes de la série temporelle.

Grâce à cette capacité à traiter les données de manière parallèle et à capturer les dépendances temporelles à différentes échelles, les Transformers ont démontré une capacité supérieure à apprendre et à modéliser les caractéristiques subtiles de la série temporelle, ce qui leur a permis de générer des prédictions plus fiables et adaptées aux données.

La figure 4.29 montre le taux d'erreur calculé avec les trois méthodes et la meilleure méthode est celle des Transformers.

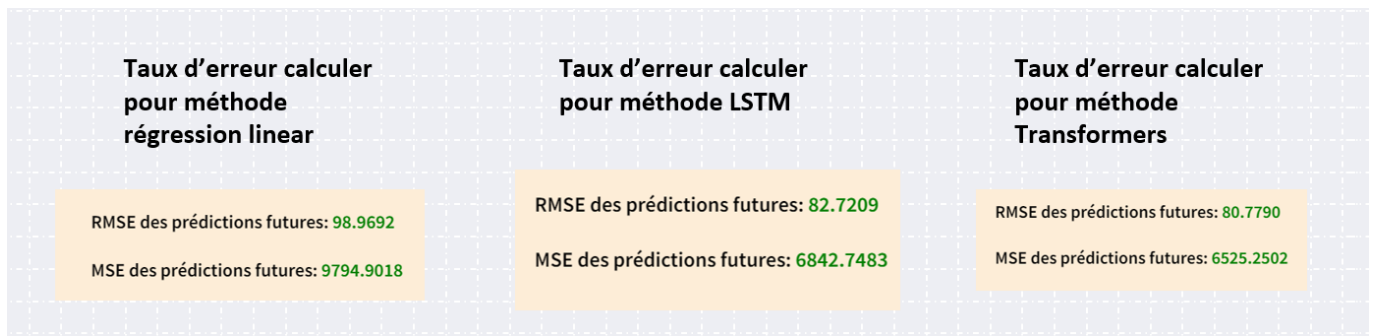


FIGURE 4.29 – Comparaison des taux d'erreur entre la Régression Linéaire, LSTM et Transformers

## 4.6 Conclusion

Dans ce chapitre, nous avons abordé l'aspect pratique de notre projet. Nous avons commencé par décrire les environnements matériels et logiciels que nous avons utilisés.

Ensuite, nous avons mis l'accent sur les différentes interfaces de notre application et expliqué comment elles fonctionnent dans la pratique.

Enfin, nous nous sommes plongés dans les détails de la construction et de la modélisation des prédictions de ventes. Nous avons utilisé les puissants modèles de prévision de séries temporelles, tels que la régression linéaire, LSTM et les Transformers.

# Conclusion et perspectives

Pour conclure, ce projet visait à concevoir et déployer une application web pour la prévision et l'analyse des ventes d'électricité pour les clients de Sonelgaz. Nous avons développé une application web d'analyse et de prévision des ventes, axée sur le thème : "Optimisation de la Planification Industrielle par la Prédiction à l'Aide de Techniques Avancées" de la gestion des consommations électriques, répondant ainsi aux besoins du Responsable Commercial de Sonelgaz à Béjaïa pour améliorer la prise de décision dans le service grands comptes.

Notre approche a ciblé la résolution de plusieurs défis rencontrés par la direction commerciale de Sonelgaz à Béjaïa : perte d'énergie, insatisfaction des clients, manque d'outils automatisés pour la prévision des consommations, et l'évitement des méthodes manuelles et traditionnelles coûteuses et chronophages.

L'adoption de la méthode Scrum nous a permis de gérer efficacement le projet en divisant les objectifs en itérations (sprints) et en favorisant une collaboration étroite avec le client. Le product backlog a joué un rôle central dans la définition des besoins fonctionnels, offrant une vision claire des tâches à accomplir et assurant la satisfaction des attentes du client.

Le langage UML a été utilisé avec succès pour modéliser les différentes fonctionnalités de notre application de prédiction. Cela nous a permis de mieux comprendre et de communiquer efficacement les concepts et les processus au sein de l'équipe de développement.

En intégrant trois modèles de prévision, à savoir la régression linéaire, LSTM et Transformers, nous avons renforcé les capacités de prédiction de notre application. Ces modèles, soigneusement sélectionnés en fonction des spécificités du problème et des données disponibles, ont apporté une plus grande précision et ont permis d'obtenir des résultats significatifs dans la prédiction des événements futurs. Les résultats obtenus montrent que le modèle Transformers a offert les prédictions les plus précises, suivi par le modèle LSTM, tandis que la régression linéaire a montré des performances moindres.

Les Transformers se sont révélés être les plus performants en termes de calcul du taux d'erreur, fournissant ainsi les prévisions les plus fiables et les plus précises parmi les trois modèles étudiés.

Ce projet a été une occasion d'apprentissage précieuse, nous permettant de nous familiariser avec les technologies avancées telles que l'intelligence artificielle, l'apprentissage automatique et le deep learning. Nous avons pu exploiter ces approches innovantes pour améliorer nos compétences en programmation et approfondir notre compréhension des concepts clés de l'IA.

En plus des aspects techniques, ce projet nous a confrontés aux réalités du monde professionnel et aux défis inhérents à la gestion d'un projet d'envergure. Nous avons développé des compétences essentielles en matière de planification, de collaboration et de résolution de problèmes, qui nous seront bénéfiques dans notre future carrière. Ce projet représente une étape cruciale dans notre parcours

académique, nous permettant d'acquérir des compétences précieuses et de nous préparer à relever les défis futurs dans le domaine de la programmation et de l'intelligence artificielle.

À l'avenir, nous prévoyons de développer de nouvelles fonctionnalités, telles que la détection d'anomalies, et d'étendre notre système pour analyser et prédire les ventes pour d'autres catégories de clients, tout en continuant à perfectionner nos connaissances et compétences dans ce domaine en constante évolution.

# Bibliographie

- [1] Anaconda. <https://www.anaconda.com/>, Consulté le 01/04/2024.
- [2] Bert. [https://huggingface.co/docs/transformers/model\\_doc/bert](https://huggingface.co/docs/transformers/model_doc/bert), Consulté le 02/05/2024.
- [3] Css. <https://developer.mozilla.org/fr/docs/Web/CSS>, Consulté le 08/03/2024.
- [4] Folium. <https://www.foliumofficiel.com/>, Consulté le 10/04/2024.
- [5] Groupe sonelgaz. sonelgaz. <https://www.sonelgaz.dz/fr,1969>, Consulté le 08/04/2024.
- [6] Introduction aux séries temporelles, tendance et composante saisonnière. [https://www.imo.universite-paris-saclay.fr/~yannig.goude/Materials/time\\_series/cours2\\_tendance\\_composante\\_saisonniere.pdf](https://www.imo.universite-paris-saclay.fr/~yannig.goude/Materials/time_series/cours2_tendance_composante_saisonniere.pdf), Consulté le 26/02/2024.
- [7] Keras. <https://keras.io/>, Consulté le 20/04/2024.
- [8] Matplotlib. <https://matplotlib.org/>, Consulté le 22/04/2024.
- [9] Methode agile. <https://slack.com/intl/fr-fr/blog/collaboration/methode-agile#:~:text=La%20m%C3%A9thode%20agile%20est%20une,de%20l'avancement%20du%20projet.>, Consulté le 2/05/2024.
- [10] Modèle de décomposition. <https://slideplayer.fr/slide/5546777/>, Consulté le 10/03/2024.
- [11] Méthodes-prévisions. <https://www.faq-logistique.com/Methodes-Previsions.html>, Consulté le 02/03/2024.
- [12] neural-networks. <https://www.ibm.com/fr-fr/topics/neural-networks>, Consulté le 02/04/2024.
- [13] Numpy. <https://numpy.org/>, Consulté le 06/04/2024.
- [14] Pandas. <https://pandas.pydata.org/>, Consulté le 16/04/2024.
- [15] Python. <https://www.python.org/>, Consulté le 14/03/2024.
- [16] Pytorch. <https://pytorch.org/>, Consulté le 19/04/2024.
- [17] Random forests. <https://www.javatpoint.com/machine-learning-random-forest-algorithm>, Consulté le 19/04/2024.
- [18] Scikit-learn. <https://scikit-learn.org/>, Consulté le 08/04/2024.
- [19] Scrum. <https://asana.com/fr/resources/what-is-scrum>, Consulté le 1/04/2024.
- [20] Streamlit. <https://streamlit.io/>, Consulté le 20/04/2024.

- [21] Série temporelle. <https://essicolo.github.io/ecologie-mathematique-R/chapitre-temps.html>, Consulté le 22/03/2024.
- [22] Transflow. <https://www.tensorflow.org/?hl=fr>, Consulté le 18/04/2024.
- [23] Uml. <https://www.ibm.com/docs/fr/rsm/7.5.0?topic=models-uml-diagrams>, Consulté le 30/04/2024.
- [24] Visual paradigm. <https://www.visual-paradigm.com/>, Consulté le 22/05/2024.
- [25] Vs code. <https://code.visualstudio.com/>, Consulté le 08/03/2024.
- [26] Christopher M Bishop. *Pattern Recognition and Machine Learning*. Springer, 2006.
- [27] Gwilym M et Reinsel Gregory C et Ljung Greta M Box, George EP et Jenkins. *Time Series Analysis : Forecasting and Control*. John Wiley & Sons, 2015.
- [28] Richard A Brockwell, Peter J et Davis. *Introduction to Time Series and Forecasting*. Springer, 2016.
- [29] Jason Brownlee. *Deep Learning for Time Series Forecasting*. Machine Learning Mastery, 2018.
- [30] Jason Brownlee. *Deep Learning for Time Series Forecasting : Predict the Future with MLPs, CNNs and LSTMs in Python*. Machine Learning Mastery, 2018.
- [31] Tong Johnson, Rie et Zhang. Accelerating stochastic gradient descent using predictive variance reduction. In *Advances in Neural Information Processing Systems*, pages 315–323, 2013.
- [32] Mykel J Kochenderfer and Tim A Wheeler. *Algorithms for Optimization*. MIT Press, 2019.
- [33] Kjell Kuhn, Max et Johnson. *Applied Predictive Modeling*. Springer, 2013.
- [34] Yann LeCun and Geoffrey Bengio, Yoshua et Hinton. Deep learning. *Nature*, 521(7553) :436–444, 2015.
- [35] Bryan Lim and Stefan Zohren. Temporal fusion transformers for interpretable multi-horizon time series forecasting. *International Journal of Forecasting*, 37(4) :1748–1764, 2021.
- [36] Noam et Parmar Niki et Uszkoreit Jakob et Jones Llion et Gomez Aidan N et Kaiser Łukasz et Polosukhin Illia Vaswani, Ashish et Shazeer. Attention is all you need. *arXiv preprint arXiv :1706.03762*, 2017.
- [37] StatQuest with Josh Starmer. Deep learning : Transformers - attention is all you need. YouTube, 2022. <https://youtu.be/v5ijNXvIC5A?si=CuRWQ6TCNvpo8PBg>, Consulté le 19/03/2024.



# Résumé

Avec l'avènement de l'intelligence artificielle, ces dernières années ont vu des progrès significatifs, notamment dans les domaines de l'apprentissage automatique (ML) et de l'apprentissage profond (DL). Ces avancées ont suscité un intérêt considérable chez les entreprises, qui voient dans ces nouvelles technologies une opportunité d'améliorer leurs opérations et de relever les défis auxquels elles sont confrontées, comme répondre aux demandes croissantes en énergie de leurs clients. Dans ce contexte, ce mémoire propose de concevoir et mettre en œuvre une application web visant à analyser et prévoir les consommations électriques au sein de la direction commerciale de Sonelgaz de BEJAIA à l'aide de techniques d'intelligence artificielle.

Cette application offre des capacités de visualisation et de prévision des consommations électriques basées sur trois modèles éprouvés dans le domaine des séries temporelles : la régression linéaire, les réseaux de neurones à mémoire à long terme (LSTM) et les Transformers. Pour exécuter ce projet, on a adopté la méthodologie Scrum, une approche agile de développement et de gestion de projet. Les tests d'application ont été menés à partir des données d'environ 1699 clients, couvrant la consommation mensuelle d'électricité de BEJAIA pendant 14 ans.

**Mots clés :** Série temporelle, Méthode Agile, Intelligence Artificielle, Deep Learning, Analyse et Prédiction des consommations électriques.

# Abstract

With the advent of artificial intelligence, recent years have seen significant progress, particularly in the fields of machine learning (ML) and deep learning (DL). These advancements have sparked considerable interest among businesses, viewing these new technologies as an opportunity to enhance their operations and address the challenges they face, such as meeting the increasing energy demands of their customers. In this context, this thesis proposes designing and implementing a web application aimed at analyzing and forecasting electricity consumption within the commercial department of Sonelgaz in BEJAIA using artificial intelligence techniques.

This application offers visualization and forecasting capabilities for electricity consumption based on three proven models in the field of time series analysis : linear regression, long short-term memory (LSTM) neural networks, and Transformers. To execute this project, we have adopted the Scrum methodology, an agile approach to project development and management. Application testing was conducted using data from approximately 1699 clients, covering BEJAIA's monthly electricity consumption over 14 years..

**Keywords :** Time Series, Agile Methodology, Artificial Intelligence, Deep Learning, Analysis and Prediction of Electricity Consumption.