



FILIÈRE : BUSINESS INTELLIGENCE & ANALYTICS

RAPPORT DE STAGE

---

Développement d'un Chatbot Intelligent  
pour un service du Catering

---

*Étudiants :*

KADDOURI OUSSAMA

*Encadrante :*

Mme. Naciri SAMIA

*Soutenu devant le Jury :*

Pr. Daadaoui

Pr. Abbad

# Remerciements

Ces quelques lignes m'offrent l'opportunité de remercier chaleureusement toutes les personnes qui ont contribué à la réalisation de ce projet. Je tiens avant tout à exprimer ma profonde gratitude à Dieu, Le Tout-Puissant, pour m'avoir donné la force, la patience et la persévérance nécessaires à la réalisation de ce travail. Je souhaite exprimer mes sincères remerciements à **Madame Naciri Samia**, mon encadrante, pour son accompagnement précieux tout au long de ce projet de fin d'année. Son expertise, ses conseils et son soutien constant ont été d'une importance immense pour la réussite de ce travail. Grâce à sa disponibilité et à ses orientations éclairées, j'ai pu surmonter les défis techniques et avancer avec rigueur et motivation. Je tiens également à remercier chaleureusement **M. Alami** notre chef de filière, pour son engagement envers l'enseignement, son expertise approfondie et son expérience précieuse, qui ont grandement contribué à notre réussite. Enfin, je remercie ma famille, mes amis et mes collègues pour leur soutien moral et leur confiance inébranlable. C'est grâce à vous que j'ai trouvé la force d'aller jusqu'au bout.

# Résumé

La dépendance des entreprises à l'intelligence artificielle augmente en parallèle avec le besoin d'un accès fiable et personnalisé aux informations sur les services de restauration. Ce projet, réalisé dans le cadre d'un stage chez DA Technologies pour le client Bel Mokhtar Catering, vise à concevoir un assistant catering basé sur LLM intégrant une approche RAG pour fournir des réponses précises, sourcées et adaptées au langage courant. L'application permettra aux utilisateurs de poser des questions sur les menus, services et tarifs en langage naturel et de recevoir des conseils personnalisés, appuyés par des données validées issues des sources fiables du client (produits et services). Cette solution vise à améliorer l'accès à l'information sur les services de catering tout en minimisant les hallucinations grâce à un système de réponse ancré dans les documents vérifiés de Bel Mokhtar.

**Mots-clés :** Intelligence Artificielle, Assistant Catering, RAG, LLM, Interface Web.

# Abstract

As people's dependence on artificial intelligence increases, the need for reliable and personalized access to catering services continues to grow. This project, developed during my internship at DA Technologies for client Bel Mokhtar Catering, aims to design a catering assistant based on Large Language Models (LLM) integrating a RAG (Retrieval-Augmented Generation) approach to provide accurate, sourced, and accessible responses. The application will enable users to ask catering questions in natural language and receive personalized advice, supported by validated data from reliable sources such as the client's product and service databases. This solution seeks to improve access to catering information while minimizing hallucinations through a response system grounded in verified documents.

**Keywords :** Artificial Intelligence, Catering Assistant, RAG, LLM, Product Databases, Web Interface.

# Liste des Abréviations

ABRÉVIATION	SIGNIFICATION
API	Application Programming Interface
BERT	Bidirectional Encoder Representations from Transformers
CSV	Comma-Separated Values
CSS	Cascading Style Sheets
GEMINI	Google's Generative Language Model
IA	Intelligence Artificielle
JSON	JavaScript Object Notation
JSONL	JSON Lines
LLM	Large Language Model
NLP	Natural Language Processing (Traitement du langage naturel)
RAG	Retrieval-Augmented Generation
UI	User Interface (Interface Utilisateur)
URL	Uniform Resource Locator
WP	WordPress

TABLE 1 – Liste des Abréviations

# Table des figures

1.1	D&A technologies . . . . .	10
1.2	Divisions de l'entreprise . . . . .	12
2.1	Architecture des LLM . . . . .	19
2.2	Processus de Tokenization . . . . .	20
2.3	Embedding . . . . .	20
2.4	L'Évolution des Techniques de ML et NLP . . . . .	22
2.5	Architecture d'un Transfomer . . . . .	24
2.6	Architecture de la RAG . . . . .	26
3.1	Interface du site web . . . . .	30
3.2	Architecture de l'Application . . . . .	33
3.3	Base de données produits . . . . .	34
3.4	Base de données services . . . . .	34
3.5	Enter Caption . . . . .	35
4.1	Pandas . . . . .	39
4.2	ChromaDB . . . . .	40
4.3	Google AI Studio . . . . .	40
4.4	WordPress . . . . .	41
4.5	Interface du chatbot intégrée au site web . . . . .	43
4.6	Interface de conversation du chatbot . . . . .	44

# Liste des tableaux

1	Liste des Abréviations . . . . .	4
1.1	Fiche descriptive de l'entreprise D&A Technologies . . . . .	11
3.1	Comparaison des caractéristiques des LLMs . . . . .	36

# Table des matières

<b>Remerciements</b>	<b>1</b>
<b>Résumé</b>	<b>2</b>
<b>Abstract</b>	<b>3</b>
<b>Liste des Abréviations</b>	<b>4</b>
<b>Liste des Figures</b>	<b>5</b>
<b>Liste des Tableaux</b>	<b>6</b>
<b>1 Contexte générale du projet</b>	<b>10</b>
1.1 Introduction . . . . .	10
1.2 Organisme d'accueil : Présentation de D&A Technologies . . . . .	10
1.2.1 D&A Technologies . . . . .	10
1.3 Divisions de l'entreprise . . . . .	11
1.3.1 Divisions de l'entreprise . . . . .	11
1.4 Les services de l'entreprise . . . . .	13
1.4.1 Conseil et Stratégie CRM . . . . .	13
1.4.2 Intégration d'un CRM Cloud dans un SI . . . . .	14
1.4.3 Déploiement Salesforce « quick-start » . . . . .	14
1.4.4 Développement d'applications métier connectées . . . . .	14
1.4.5 Formation Salesforce . . . . .	14
1.5 Contexte général du projet . . . . .	14
1.5.1 Présentation du client « le groupe HS traiteur ou Belmokhtar traiteur » .	14
1.5.2 Présentation du Projet . . . . .	15
<b>2 Litterature Review</b>	<b>18</b>
2.1 Large Language Model (LLM) . . . . .	18

2.1.1	Définition et Concepts de Base . . . . .	18
2.1.2	Architecture des LLM . . . . .	19
2.2	Transformers . . . . .	22
2.2.1	Historique et Évolution . . . . .	22
2.2.2	Architecture des Transformers . . . . .	24
2.3	Retrieval Augmented Generation (RAG) . . . . .	25
2.3.1	Définition . . . . .	25
2.3.2	Architecture de la RAG . . . . .	26
<b>3</b>	<b>Conception de la Solution</b>	<b>29</b>
3.1	Étude de l'existant . . . . .	29
3.1.1	Analyse des solutions actuelles du client . . . . .	29
3.1.2	Benchmark des solutions similaires sur le marché internationale . . . . .	30
3.1.3	Analyse des technologies existantes . . . . .	31
3.1.4	Synthèse de l'étude de l'existant . . . . .	31
3.2	Proposition de la Solution et Architecture . . . . .	32
3.3	Sources de Données . . . . .	33
3.4	LLM Choisi . . . . .	35
<b>4</b>	<b>Réalisation</b>	<b>38</b>
4.1	Technologies Utilisées . . . . .	38
4.1.1	Collecte des Données . . . . .	38
4.1.2	Embedding et Indexation . . . . .	39
4.1.3	Génération de Réponse . . . . .	40
4.1.4	Développement de l'Interface Utilisateur . . . . .	40
4.2	Présentation de la Solution . . . . .	41
4.2.1	Collecte de Données, Vectorisation et Pipeline du RAG . . . . .	41
4.2.2	Présentation de l'Application . . . . .	43
<b>Conclusion Générale</b>		<b>47</b>
<b>Références Bibliographiques</b>		<b>49</b>

# Introduction Générale

Dans un contexte économique en perpétuelle évolution, la capacité des entreprises à offrir une expérience client optimale est essentielle pour assurer leur succès. Les avancées technologiques ont introduit de nouveaux outils et méthodes pour relever ce défi, parmi lesquels les grands modèles de langage (LLM) et les systèmes de génération augmentée par récupération (RAG) jouent un rôle central. Les LLM sont des modèles d'intelligence artificielle conçus pour comprendre et générer du langage naturel de manière sophistiquée, offrant des interactions conversationnelles fluides. Parallèlement, les systèmes RAG combinent ces modèles avec des bases de connaissances spécifiques, permettant de fournir des réponses précises et sourcées en temps réel.

L'intégration de ces technologies dans les services d'assistance client présente de nombreux avantages. Les LLM permettent une compréhension approfondie des requêtes clients et une génération de réponses pertinentes, offrant ainsi une assistance personnalisée et disponible 24h/24. Les systèmes RAG, quant à eux, garantissent la fiabilité des informations en les ancrant dans des données vérifiées, réduisant ainsi les risques d'erreurs et de désinformation. En combinant ces technologies avec des interfaces web intuitives et des systèmes de gestion de contenu comme WordPress, les entreprises peuvent transformer radicalement leur service client, en particulier dans des secteurs exigeants comme la restauration et le catering.

Dans ce rapport de projet de fin d'études, nous analyserons en profondeur l'utilisation des LLM, des systèmes RAG et des interfaces web modernes dans le domaine du catering. Nous examinerons les différentes techniques et méthodologies employées pour développer, intégrer et déployer notre solution de chatbot pour Bel Mokhtar Catering, ainsi que les défis et les opportunités associés à sa mise en œuvre. Enfin, nous discuterons des implications de ces technologies sur la compétitivité et la croissance des entreprises de restauration dans un environnement commercial en constante mutation.

# Chapitre 1

## Contexte générale du projet

### 1.1 Introduction

Ce chapitre traite le contexte général de mon stage. Tout d'abord, une présentation de l'Organisme d'accueil . Ensuite, le cadre général du projet sera exposé.

### 1.2 Organisme d'accueil : Présentation de D&A Technologies

#### 1.2.1 D&A Technologies



FIGURE 1.1 – D&A technologies

D&A Technologies est une société privée créée en 2012. Elle accompagne les entreprises à tirer le meilleur parti de la puissance de la plateforme Salesforce. Sa mission consiste à concevoir, intégrer et déployer une solution flexible pour accompagner les clients, avec une solution adaptée permettant d'améliorer à chaque étape l'efficacité de la Relation Client. L'objectif principal est de contribuer à l'optimisation de la performance des différents métiers et d'œuvrer aux côtés d'entreprises qui ont choisi de se réinventer et de s'ouvrir à de nouvelles opportunités business. Les Consultants couvrent tout les phases projet, depuis les phases de conseil et d'analyse des

processus métier jusqu'à l'implémentation et à la formation, ainsi que l'évolution des solutions Salesforce existantes.

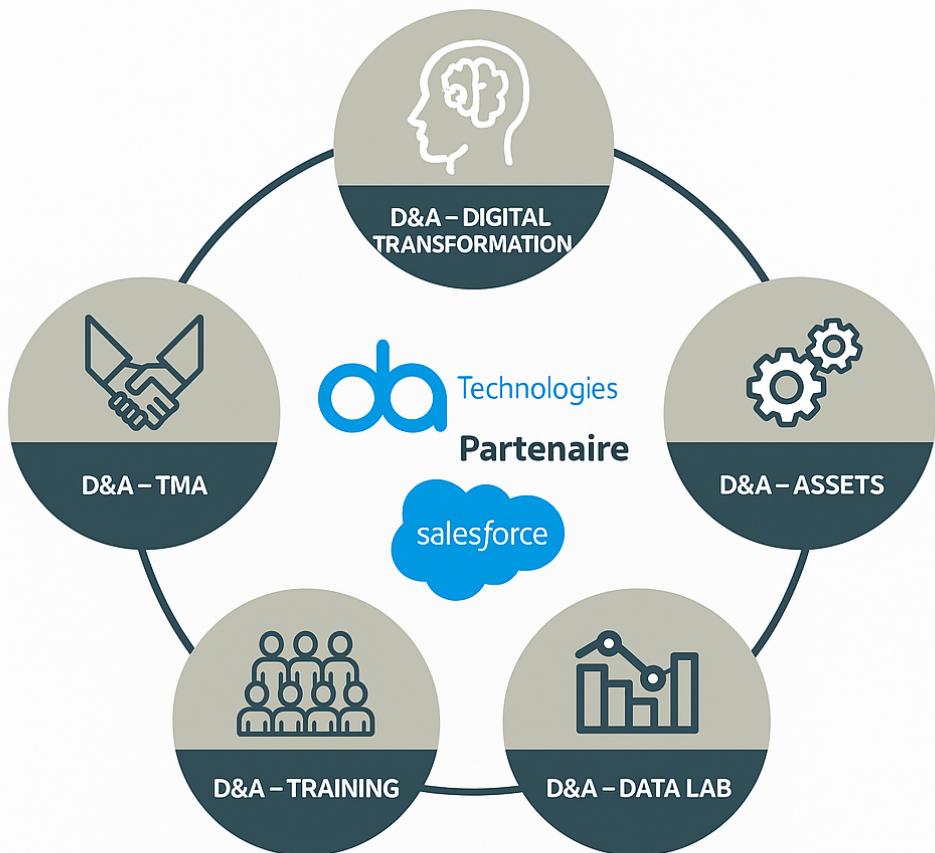
<b>Création</b>	2012
<b>Fondateurs</b>	Driss Lahrichi
<b>Forme juridique</b>	S.A.R.L
<b>Siège social</b>	Casablanca, Maroc
<b>Direction</b>	Othmane Cherif Alami (PDG)
<b>Activité</b>	Conseil, implémentation et formation Salesforce
<b>Effectif</b>	11-50 personnes
<b>Capital</b>	2 000 000 DHS
<b>Adresse</b>	33, Rue Daoud Dahiri, Appt. N° 1, 1er Étage, Maârif - Casablanca

TABLE 1.1 – Fiche descriptive de l'entreprise D&A Technologies

## 1.3 Divisions de l'entreprise

### 1.3.1 Divisions de l'entreprise

D&A Technologies est structurée en plusieurs divisions, chacune ayant un rôle spécifique dans l'accompagnement des clients. Voici les principales divisions :



**Figure 2 : Divisions de l'entreprise**

FIGURE 1.2 – Divisions de l'entreprise

### Data et Business Intelligence

Elle accompagne les entreprises dans la conception et la mise en œuvre de leurs stratégies de Transformation Digitale. Elle détient une expertise sectorielle dans les domaines de la Transition Énergétique, de l'Assurance et de l'Éducation, et développe de nouvelles expertises métiers, telles que dans la Santé et les Télécoms.

### CRM

Grâce à son expertise dans le domaine du CRM, D&A Technologies accompagne ses clients pour la réorganisation et la réorientation « Client » dans leur système d'information. En effet, le CRM recouvre l'ensemble des fonctions de l'entreprise visant à conquérir, connaître, cibler et fidéliser sa clientèle. Le CRM regroupe la gestion des opérations de marketing, l'automatisation des forces de vente, le service client, etc. Forte de son expertise dans la valorisation du potentiel « Client », elle accompagne ses clients sur toutes les phases de leur projet CRM, grâce à un

conseil méthodologique, pratique et orienté résultats.

## TMA

Elle accompagne ses clients à travers un dispositif de maintenance applicative agile et éprouvé, ainsi que la mise à disposition de ressources expérimentées. Elle est capable de répondre aux besoins de montée en charge de ses clients en un temps rapide, grâce à un modèle de staffing performant et fiable.

## TRAINING

Pôle de formation capable de former des ressources, jeunes ou expérimentées, à monter en compétences sur Salesforce à travers des cycles courts alliant programmes mixtes (techniques et fonctionnels).

## ASSETS

Grâce aux expertises sectorielles et fonctionnelles développées à travers l'accompagnement de ses nombreux clients, D&A Technologies est en mesure de mettre à la disposition de ses clients et partenaires des offres verticales prêtes à l'emploi (notamment dans les secteurs de l'Assurance, de l'Éducation, des Énergies, etc.), et ce, en un temps court.

## DATALAB

Pôle dédié à l'ingénierie de la Data, alliant conception de stratégies de gestion des données et mise en œuvre de solutions d'analyse avancées, pour garantir une utilisation optimale des informations, favorisant ainsi la prise de décision éclairée et la croissance durable de nos clients.

## 1.4 Les services de l'entreprise

D&A Technologies offre un large éventail de services à ses clients, que ces derniers soient des particuliers ou des entreprises. Parmi ses services :

### 1.4.1 Conseil et Stratégie CRM

Accompagne les clients dans l'analyse de leurs gestions de la relation client et aide à redéfinir la stratégie CRM et e-marketing. Audit CRM, benchmark des solutions du marché, adaptées au besoin, assistance dans sa consultation.

### 1.4.2 Intégration d'un CRM Cloud dans un SI

Elle propose d'interfacer le CRM Salesforce avec les sites web, ERP, pages réseaux sociaux ou toute autre brique du SI de l'entreprise. Analyse, développement et documentation des interfaces. Intégration temps réel grâce aux nombreuses APIs de Salesforce.

### 1.4.3 Déploiement Salesforce « quick-start »

Elle propose un déploiement Salesforce « Quick start » (une à trois semaines) en immersion dans les équipes de l'entreprise. Déploiement Salesforce 'Clé-En-Main' Accompagne les entreprises de la conception à la réalisation avec différents niveaux de paramétrage.

### 1.4.4 Développement d'applications métier connectées

Le développement d'une large palette d'outils pour créer de nouvelles applications nativement mobiles avec la plateforme Force.com.

### 1.4.5 Formation Salesforce

D&A Technologies propose des formations théoriques et pratiques pour les utilisateurs techniques et fonctionnels de Salesforce, allant jusqu'à la préparation des certifications (ADM et DEV).

## 1.5 Contexte général du projet

Nous décrivons dans ce qui suit notre projet en présentant le client et en exposant la problématique, les objectifs du projet et la conduite de sa réalisation.

### 1.5.1 Présentation du client « le groupe HS traiteur ou Belmokhtar traiteur »

#### Identité et Positionnement

Le Groupe HS & Belmokhtar Traiteur est un acteur majeur de la restauration événementielle au Maroc, spécialisé dans l'organisation de mariages, buffets et réceptions professionnelles. Basé à Salé, le groupe dessert principalement les régions de Rabat, Salé et Témara, avec une réputation établie depuis 2006 dans le domaine de la gastronomie et de la gestion d'événements.

## Services Principaux

- **Traiteur Mariage** : Organisation complète de mariages, fiançailles et cérémonies de henna
- **Buffets d'entreprise** : Services pour séminaires, assemblées générales et lancements de produits
- **Buffets de soutenance** : Solutions sur mesure pour les soutenances universitaires
- **Réception à domicile** : Prestations personnalisées pour événements privés
- **Services événementiels** : Anniversaires, réceptions professionnelles et grands séminaires

## Valeurs et Engagement

- **Innovation** : Exploration continue de nouvelles tendances gastronomiques
- **Qualité** : Engagement pour des prestations alliant professionnalisme et excellence
- **Personnalisation** : Solutions adaptées aux besoins spécifiques de chaque client
- **Service client** : Disponibilité et réactivité pour garantir la réussite de chaque événement

## Positionnement sur le Marché

Le Groupe HS & Belmokhtar Traiteur se positionne comme un *maître traiteur* combinant tradition et modernité. Avec une présence établie dans plusieurs villes marocaines (Rabat, Salé, Casablanca, Marrakech), l'entreprise cible aussi bien les particuliers pour événements familiaux que les entreprises pour leurs manifestations professionnelles.

### 1.5.2 Présentation du Projet

#### Cadre général du projet

Nous avons déjà mentionné que le Groupe HS & Belmokhtar Traiteur propose des services de restauration pour divers événements (mariages, buffets d'entreprise, réceptions). Pour ce faire, les équipes commerciales et le service client doivent répondre à de nombreuses demandes d'information concernant les menus, tarifs et disponibilités. Cependant, gérer manuellement ces requêtes est une tâche fastidieuse, exigeant un effort et un temps considérable.

Des travaux antérieurs, comme le développement du site web et des formulaires de contact, ont été réalisés dans ce contexte. Ces outils permettent aux clients d'envoyer leurs demandes,

mais nécessitent une intervention humaine pour chaque réponse. Sur la base de ces interactions, les équipes décident des suites à donner (devis, rendez-vous, etc.).

Ainsi, l'objectif général de ce projet est d'optimiser la gestion des demandes clients en fournissant des réponses instantanées et personnalisées aux questions fréquentes sur les services de restauration.

## Problématiques et Défis

Étant donné que le site web actuel et les canaux traditionnels s'améliorent avec le temps, certains problèmes ont été rencontrés :

- Les clients peuvent avoir des besoins urgents ou des questions simples en dehors des heures d'ouverture, rendant difficile l'obtention immédiate d'informations.
- De plus, certaines demandes répétitives mobilisent inutilement le personnel qui pourrait se concentrer sur des tâches à plus forte valeur ajoutée.

Ces défis ont conduit à la recherche et au développement du projet d'assistant virtuel, dont l'objectif est de détecter et de répondre automatiquement aux questions des clients à partir de leurs requêtes en langage naturel, et de cartographier leurs besoins en lien avec les services de restauration.

## Objectif

Notre objectif principal est de développer un workflow automatisé pour répondre aux besoins critiques de notre entreprise. Ce projet sera nommé « Assistant Intelligent pour Belmokhtar Traiteur ».

Nous prévoyons également de concevoir cet assistant pour être adaptable à plusieurs cas d'utilisation spécifiques en matière de type d'événement (mariage, entreprise, soutenance). En rendant le système spécifique à ces besoins, nous pourrons offrir une solution efficace et rentable pour la gestion des interactions clients.

## Planification et conduite du projet

Le projet s'est déroulé sur une période de deux mois (juillet et août), selon une méthodologie Agile en deux phases principales :

**Phase 1 : Spécifications (Semaines 1-2)** Cette phase initiale a permis de définir clairement les contours du projet :

- **Semaine 1** : Analyse approfondie des besoins client et benchmark des solutions existantes
- **Semaine 2** : Rédaction des spécifications fonctionnelles (réponses aux questions fréquentes, gestion des menus) et non fonctionnelles (performance, sécurité, disponibilité 24/7)

**Phase 2 : Conception et réalisation (Semaines 3-8)** Cette phase opérationnelle s'est articulée autour de plusieurs étapes clés :

- **Semaines 3-4** : Conception de l'architecture système et de l'interface utilisateur
- **Semaines 5-6** : Développement du backend (API FastAPI) et intégration du système RAG
- **Semaine 7-8** : Développement du frontend et tests unitaires, tests d'acceptation avec le client, formation et documentation finale

Cette planification a permis un développement progressif avec des livraisons intermédiaires, assurant une validation régulière par le client et une adaptation continue aux besoins spécifiques de Belmokhtar Traiteur.

## Conclusion du chapitre

Ce chapitre a établi le cadre de notre projet en présentant D&A Technologies et le client Belmokhtar Traiteur, mettant en lumière les enjeux d'optimisation de la relation client dans le secteur de la restauration événementielle. Face aux défis identifiés, notre assistant intelligent se positionne comme une solution innovante pour répondre aux besoins de disponibilité et de personnalisation. La planification structurée sur deux mois assure une mise en œuvre méthodique, préparant ainsi le terrain pour l'étude technique détaillée qui suivra.

# Chapitre 2

## Litterature Review

### Introduction

Après avoir parcouru les fonctionnalités du cahier des charges, ce chapitre a pour objectif de fournir une base solide des concepts théoriques que nous allons utiliser tout au long de notre projet. Cette étape est primordiale car la compréhension des **Large Language Models (LLM)** et de la **Retrieval Augmented Generation (RAG)** est indispensable à la compréhension de notre solution d'assistant santé.

### 2.1 Large Language Model (LLM)

#### 2.1.1 Définition et Concepts de Base

Un **LLM** est un type d'intelligence artificielle qui applique des techniques de réseaux neuronaux avec un grand nombre de paramètres pour traiter et comprendre les langages humains en utilisant des techniques d'apprentissage auto-supervisées. Des tâches telles que la génération de texte, la traduction automatique, la génération d'images à partir de textes, les chatbots ou encore l'IA conversationnelle sont des applications des **LLM**.

De nombreuses techniques ont été essayées pour effectuer des tâches liées au langage naturel, mais les **LLM** reposent exclusivement sur les méthodologies du Deep Learning. Les modèles **LLM** sont très efficaces pour capturer les relations complexes entre les entités dans un texte donné et peuvent générer du texte en utilisant la syntaxe de la langue particulière dans laquelle nous souhaitons opérer[1].

## 2.1.2 Architecture des LLM

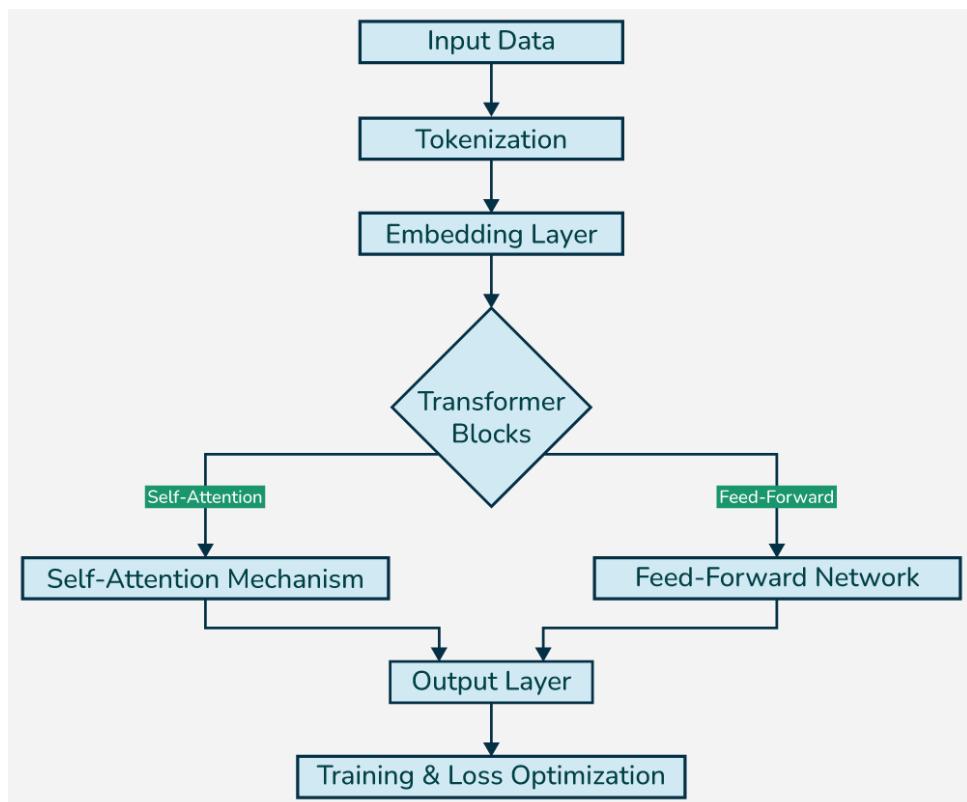


FIGURE 2.1 – Architecture des LLM

Les **LLM** sont des systèmes conçus pour traiter et générer un texte de type humain. Leur architecture implique plusieurs couches et composants, chacun contribuant à la capacité du modèle à comprendre et à produire du langage. La figure précédente présente ces différents composants du **LLM**, parmi lesquels on trouve :

### **Input Layer : Tokenization**

C'est le processus de conversion d'une séquence de texte en parties plus petites, connues sous le nom de tokens. Ces jetons peuvent être aussi petits que des caractères ou aussi longs que des mots. La principale raison de l'importance de ce processus est qu'il aide les machines à comprendre le langage humain en le décomposant en petits morceaux, plus faciles à analyser. La figure suivante présente une brève explication de ce processus :

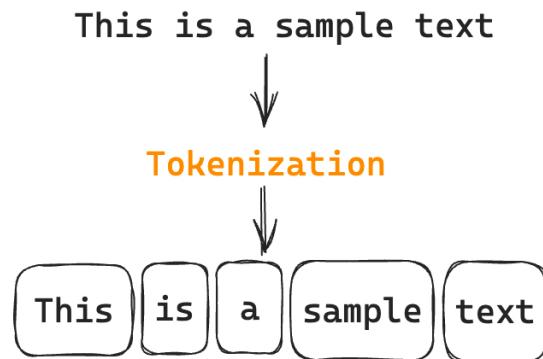


FIGURE 2.2 – Processus de Tokenization

## Embedding Layer

Cette étape est constituée de deux parties essentielles :

- **Word Embeddings** étant une méthode d'encodage qui vise à représenter les mots ou les phrases d'un texte par des vecteurs de nombres réels, décrit dans un modèle vectoriel (ou Vector Space Model). D'une manière plus simple, chaque mot du vocabulaire V étudié sera représenté par un vecteur de taille m. Le principe du Word Embedding est de projeter chacun de ces mots dans un espace vectoriel d'une taille fixe N (N étant différent de m). C'est-à-dire quelle que soit la taille du vocabulaire, on devra être capable de projeter un mot dans son espace.
- **Positional Embeddings** : Étant donné que les transformateurs ne comprennent pas intrinsèquement l'ordre des Tokens, des embeddings de position sont ajoutés aux embeddings de mots pour fournir des informations sur les positions des Tokens à l'intérieur d'une phrase.

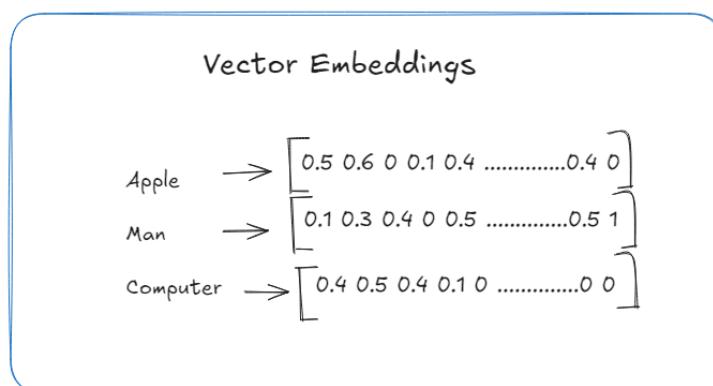


FIGURE 2.3 – Embedding

## Transformer Blocks

L'architecture des LLM implique généralement l'empilement de plusieurs couches de transformers (ou blocs) les unes sur les autres. Chaque bloc se compose d'un mécanisme d'auto-attention multi-têtes et d'un réseau neuronal feed-forward. Cet empilement permet au modèle d'apprendre des représentations hiérarchiques complexes des données.

### — Mécanisme d'auto-attention :

- 1. **Calcul des requêtes, des clés et des valeurs** : Pour chaque jeton en entrée, trois vecteurs sont dérivés à l'aide de matrices de pondération apprises : Requêtes (Q), Clés (K) et Valeurs (V). Ces vecteurs sont projetés dans un espace de dimension inférieure pour un traitement efficace.
- 2. **Calcul des scores d'attention** : Les scores d'attention sont calculés en effectuant le produit scalaire du vecteur de requête de chaque jeton avec tous les vecteurs de clés, ce qui donne des scores représentant la pertinence ou la similarité entre les jetons. Une fonction softmax normalise ensuite ces scores pour garantir que leur somme est égale à 1, représentant les probabilités.
- 3. **Génération de la sortie** : Le vecteur de sortie de chaque jeton est calculé comme la somme pondérée de tous les vecteurs de valeur, les pondérations étant données par les scores d'attention. Cette étape combine les informations des différentes parties de la séquence d'entrée selon les pertinences calculées.
- **Multi-Head Attention** : Plusieurs têtes d'attention sont utilisées pour capturer différents aspects des relations entre les Tokens. Chaque tête fonctionne dans un sous-espace séparé, et les résultats sont concaténés puis retranscrits dans l'espace d'origine.
- **Feed-Forward Neural Network** : Après le mécanisme d'attention, la sortie est passée à travers un réseau neuronal feedforward (une série de couches denses avec des fonctions d'activation), appliqué indépendamment à chaque position.
- **Layer Normalization and Residual Connections** : Chaque sous-couche est suivie d'une normalisation de couche et d'une connexion résiduelle, ce qui aide à stabiliser l'entraînement et permet des réseaux plus profonds.

## Couche de Sortie (Output Layer)

Cette couche est utilisée pour le **décodage**. Elle se présente comme :

- **Objectif de la modélisation linguistique** : Dans des modèles autorégressifs comme

le TPG, le modèle est entraîné pour prédire le Token suivant dans une séquence étant donné les Tokens précédents. Dans des modèles de langage masqués comme l'ORET, le modèle prédit les Tokens manquants dans une séquence.

- **Couche de Softmax :** La couche finale est généralement une fonction softmax qui convertit la sortie du modèle en une distribution de probabilité sur le vocabulaire, lui permettant de sélectionner le Token le plus probable suivant ou de remplir un Token masqué.

## 2.2 Transformers

### 2.2.1 Historique et Évolution

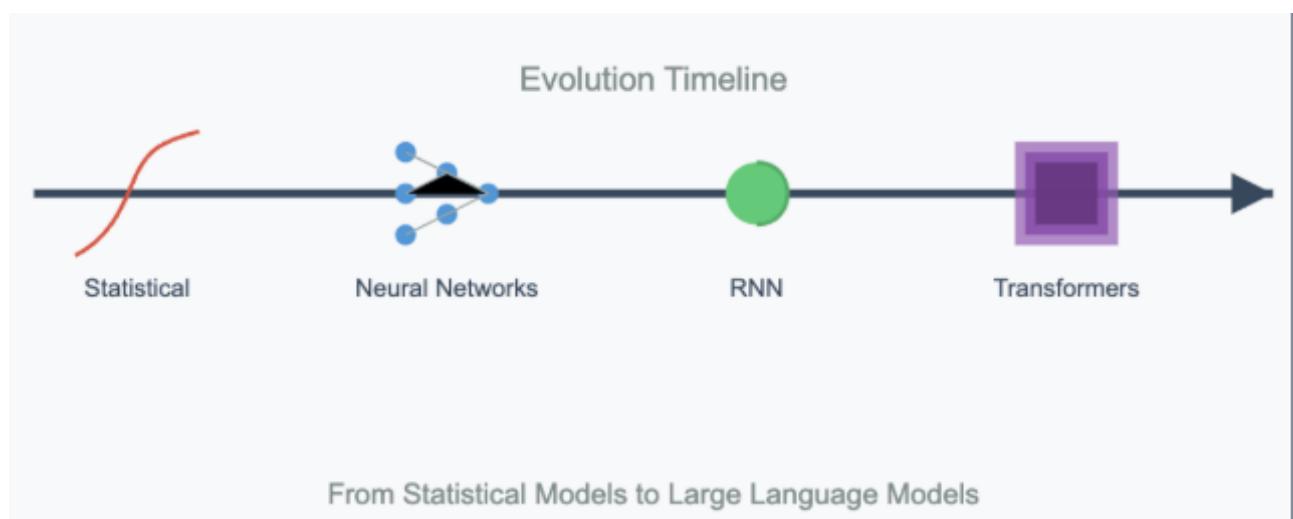


FIGURE 2.4 – L’Évolution des Techniques de ML et NLP

### Courte Introduction aux Réseaux LSTM

Les réseaux **Long Short-Term Memory (LSTM)** ont constitué une avancée significative dans le domaine de l’intelligence artificielle, en particulier pour la gestion des séquences de données. Issus d’une forme avancée des Réseaux Neuronaux Récurrents (RNN), les LSTMs ont été conçus pour surmonter les défis liés à l’apprentissage des dépendances à long terme, une limitation inhérente aux RNN traditionnels. L’innovation clé des LSTMs réside dans leur mécanisme à base de portes (input gate, forget gate et output gate), qui régule le flux d’informations. Cette conception permet aux LSTMs de retenir des informations sur de plus longues périodes, les rendant très efficaces pour des tâches complexes dans le traitement du langage naturel (NLP), la reconnaissance vocale et l’analyse de séries temporelles.

## Limitations des Réseaux LSTM

Bien que les LSTMs représentent une avancée significative par rapport aux RNN traditionnels, ils ont beaucoup de limitations. Un problème majeur est leur complexité computationnelle et leur inefficacité, principalement à cause de la nature séquentielle de leur traitement. Les LSTMs peinent à traiter des séquences très longues, car le temps et les ressources de calcul nécessaires augmentent linéairement avec la longueur de la séquence. Cela les rend moins pratiques pour certaines applications du monde réel impliquant des ensembles de données massifs ou nécessitant un traitement rapide.

## L'Avènement des Transformers

Cependant, le rythme incessant des recherches en IA a conduit au développement d'une architecture encore plus puissante : le **Transformer**. Introduit dans l'article fondateur « **Attention Is All You Need** » en 2017, les Transformers ont marqué un changement radical par rapport à la structure récurrente des LSTMs. Au lieu de traiter les données de manière séquentielle, les Transformers utilisent un mécanisme appelé **self-attention** pour traiter des séquences entières de données en parallèle. Ce changement architectural aborde les limitations des LSTMs dans la gestion des séquences très longues et permet des calculs beaucoup plus rapides, ce qui est particulièrement avantageux étant donné la taille croissante des ensembles de données dans les tâches de traitement du langage naturel (NLP).

## Mécanisme de Self-Attention dans les Transformers

Le mécanisme de **self-attention** des Transformers permet à chaque position dans la séquence d'entrée d'accéder directement à toutes les autres positions, contrairement aux LSTMs qui traitent les données étape par étape. Cela permet au modèle de capturer des relations complexes et distantes dans les données de manière plus efficace. De plus, les Transformers éliminent totalement le besoin de récurrence, conduisant à des améliorations significatives en termes d'efficacité d'entraînement et de scalabilité. Cette architecture a été le socle de nombreux modèles de langage de pointe, y compris la série GPT d'OpenAI et BERT de Google, révolutionnant des tâches comme la traduction, la génération de texte et la réponse à des questions.

## Impact et Futur des Transformers

Les Transformers ont non seulement surpassé les LSTMs dans de nombreux benchmarks de NLP, mais ont également ouvert de nouvelles perspectives dans la recherche en IA. Leur capacité

à gérer des ensembles de données à grande échelle et à capturer des motifs complexes dans les données a repoussé les limites de ce qui est possible en matière de compréhension machine du langage. En outre, leur architecture polyvalente a été adaptée à diverses applications au-delà du traitement du langage, comme dans la vision par ordinateur et le traitement audio.

### 2.2.2 Architecture des Transformers

Le modèle est principalement composé de deux blocs :

- **Encodeur (à gauche)** : l'encodeur reçoit une entrée et construit une représentation de celle-ci (ses caractéristiques). Cela signifie que le modèle est optimisé pour acquérir une compréhension venant de ces entrées.
- **Décodeur (à droite)** : le décodeur utilise la représentation de l'encodeur (les caractéristiques) en plus des autres entrées pour générer une séquence cible. Cela signifie que le modèle est optimisé pour générer des sorties.

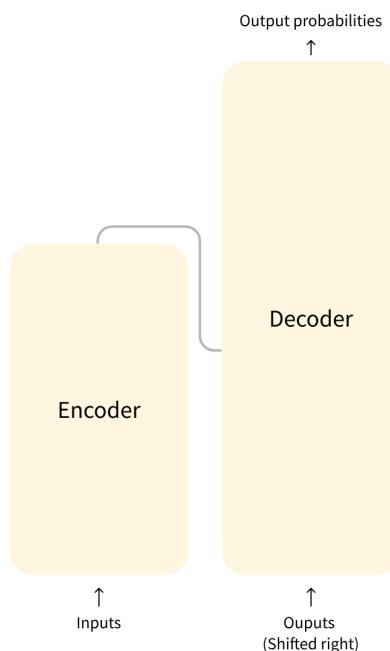


FIGURE 2.5 – Architecture d'un Transfomer

L'architecture du Transformer est construite comme suit :

- **Un bloc "Embedding & Encoding"** est connecté à l'entrée de la pile d'encodeur et décodeur. Cette couche transforme les mots de la séquence d'entrée en vecteur pour qu'ils soient compris par l'encodeur et le décodeur. Ces derniers pourront alors effectuer des opérations calculatoires sur ces vecteurs pour y déceler les mots de la séquence d'entrée ayant une signification voisine. A cela s'ajoute un encodage positionnel qui permet au

Transformer de gérer l'ordre des mots dans une phrase et ainsi comprendre le contexte global de la séquence d'entrée.

- **Une pile d'encodeurs** : Chaque encodeur prenant en entrée la sortie de l'encodeur précédent sauf le premier qui prend en entrée les mots vectorisés et encodés par encodage positionnel.
- **Une pile de décodeurs** : Chaque décodeur prenant en entrée la sortie du décodeur précédent et la sortie du dernier encodeur sauf pour le premier décodeur qui ne prend en entrée que la sortie du dernier encodeur.

En résumé, un Transformer est composé de blocs d'encodeurs et de décodeurs dont chacun possède sa propre matrice de poids. A cela s'ajoute un bloc transformant la séquence d'entrée dans un format qui peut être ingéré par les blocs encodeurs et décodeurs.

## 2.3 Retrieval Augmented Generation (RAG)

### 2.3.1 Définition

L'augmentation de génération par récupération (**Retrieval-Augmented Generation - RAG**) est une approche innovante dans le domaine du traitement du langage naturel (NLP). Elle combine les atouts des modèles basés sur la récupération d'informations et ceux basés sur la génération afin d'améliorer la qualité du texte produit.

Dans les modèles de langage traditionnels (LLMs), les réponses sont générées uniquement à partir des données sur lesquelles le modèle a été entraîné. Cela peut limiter l'accès à des informations récentes ou à des détails spécifiques nécessaires pour certaines tâches. RAG surmonte cette limite en intégrant un mécanisme de récupération qui permet au modèle d'accéder, en temps réel, à des bases de données ou à des documents externes[2].

### 2.3.2 Architecture de la RAG

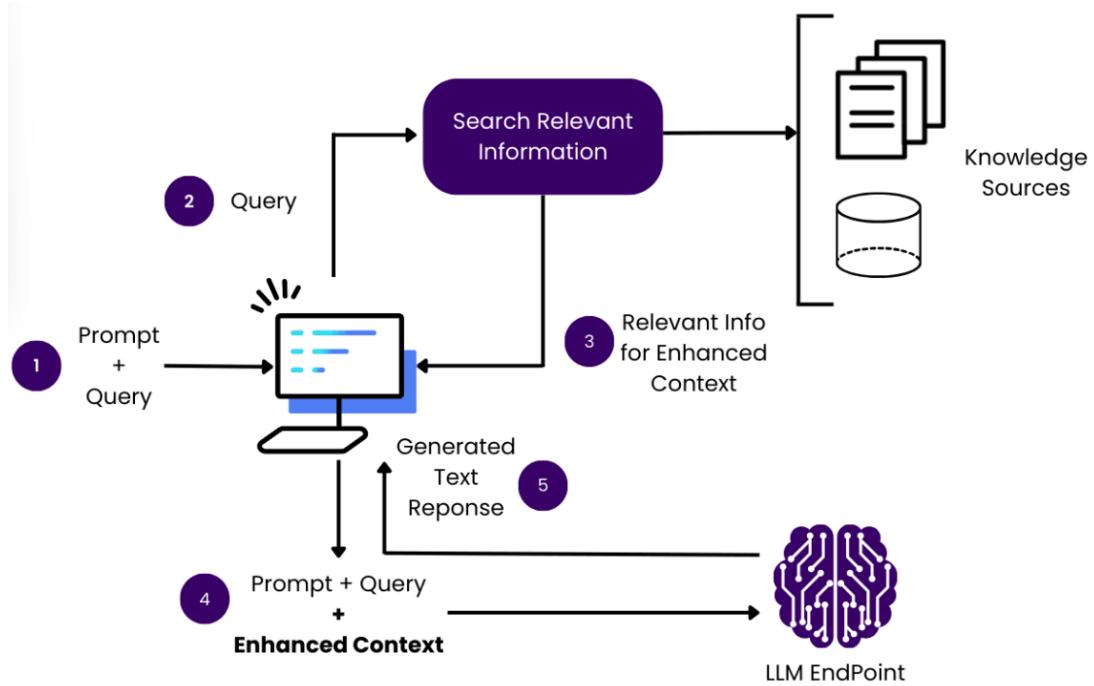


FIGURE 2.6 – Architecture de la RAG

L'architecture de la RAG, présentée sur la figure ci-dessus, est composée de plusieurs étapes qui définissent son fonctionnement.[3]

1. **Réception d'une requête :** Retrieval-Augmented Generation (RAG) commence lorsqu'une requête ou un prompt est soumis par un utilisateur. Il peut s'agir d'une question spécifique, comme une demande d'actualités récentes sur un sujet, ou d'une requête plus large pour la génération de contenu créatif.
2. **Recherche d'informations pertinentes :** Une fois la requête reçue et interprétée, le composant de récupération (retriever) du système RAG se met à rechercher des informations pertinentes dans diverses bases de données et sources externes. Cela peut inclure des articles de presse récents, des publications scientifiques ou des référentiels de données préalablement sélectionnés.

L'objectif majeur de cette étape est de trouver du contenu qui correspond ou est lié à la requête. Cette recherche est alimentée par des algorithmes capables de comprendre le sens sémantique de la question utilisateur, garantissant ainsi que la récupération soit la plus pertinente possible.

3. **Récupération des informations pertinentes :** Une fois la recherche effectuée, le système extrait les documents ou les extraits d'informations les plus pertinents identifiés.

La qualité des données récupérées influence directement la précision et la pertinence du résultat final. C'est pourquoi le processus de récupération est conçu pour sélectionner des sources crédibles et fiables qui correspondent le mieux à la requête de l'utilisateur.

4. **Augmentation du prompt avec du contexte supplémentaire :** Les informations récupérées sont ensuite utilisées pour enrichir la requête initiale. Ce prompt augmenté intègre désormais du contexte supplémentaire qui améliore la compréhension de la demande de l'utilisateur.

Par exemple, si la requête porte sur les avancées scientifiques récentes en robotique, le prompt enrichi peut inclure des extraits des dernières publications scientifiques ou des articles d'experts sur le sujet. Ce processus d'augmentation permet au modèle génératif de produire des réponses plus précises et informées, intégrant des faits récents qui n'étaient pas présents dans les données d'entraînement du modèle de langage.

5. **Récupération de la réponse du LLM :** À cette étape, le prompt enrichi est soumis à un grand modèle de langage (LLM). Ce dernier, entraîné sur d'énormes volumes de textes, exploite ses capacités génératives pour synthétiser les informations du prompt augmenté en une réponse cohérente et pertinente. Selon la nature de la requête initiale, une réponse est générée et transmise à l'utilisateur. L'objectif est de produire une réponse plus précise, plus informative et mieux contextualisée que celle qui aurait été générée par un simple modèle génératif, sans augmentation du prompt.

L'objectif principal du RAG est de fournir une réponse fiable et précise sur des questions touchant des informations spécifiques et récentes, hors de la portée d'un LLM. C'est la méthode que nous avons adopté pour la création de notre chatbot.

## Conclusion

Ce chapitre a posé les concepts essentiels sur lesquels se base notre projet, en explorant les LLM et l'approche RAG. Nous avons détaillé les processus fondamentaux tels que la tokenisation, les embeddings (word et positional), ainsi que l'architecture des transformers, incluant les mécanismes d'auto-attention et les réseaux feed-forward. Ces éléments sont cruciaux pour comprendre comment les LLM traitent et génèrent du texte. Enfin, nous avons introduit la RAG, une méthode innovante qui combine la récupération d'informations externes avec la génération de texte pour améliorer la précision et la pertinence des réponses. Ces concepts constituent la base théorique nécessaire pour la suite de notre projet, notamment dans la conception de la

solution avant de passer au développement d'un chatbot performant.

# Chapitre 3

## Conception de la Solution

### Introduction

Dans un monde où l'accès à une information fiable est crucial, l'intégration de l'intelligence artificielle dans les services de restauration représente une évolution significative. Ces outils, souvent basés sur l'intelligence artificielle, visent à améliorer l'expérience client, fournir des informations précises sur les menus et services, et accompagner les utilisateurs dans leurs démarches d'organisation d'événements. Cependant, malgré les progrès réalisés dans ce domaine, plusieurs défis demeurent, notamment en termes de précision des réponses, de personnalisation des interactions et d'intégration avec les données spécifiques des entreprises de restauration. Dans ce chapitre, nous explorerons les outils existants dans le domaine des assistants virtuels pour la restauration. Nous analyserons leurs fonctionnalités, leurs forces et leurs limites afin de comprendre les possibilités d'amélioration. À travers cette étude, nous mettrons en évidence les lacunes et les opportunités du marché. À la fin de ce chapitre, nous proposerons notre valeur ajoutée, qui repose principalement sur une approche innovante combinant les **LLM** (Large Language Models) et les données spécifiques de Belmokhtar Traiteur, ainsi qu'une interface simple et intuitive dédiée aux clients et aux équipes de l'entreprise.

### 3.1 Étude de l'existant

#### 3.1.1 Analyse des solutions actuelles du client

Le Groupe HS & Belmokhtar Traiteur dispose actuellement de plusieurs canaux de communication pour interagir avec ses clients :

## Site web institutionnel

Le site web actuel ([www.hs-traiteur.com](http://www.hs-traiteur.com)) présente les différents services et menus proposés, mais présente plusieurs limitations :

- **Contenu statique** : Les informations sur les menus et tarifs nécessitent une mise à jour manuelle
- **Absence d'interactivité** : Aucun système de conversation ou de recommandation personnalisée
- **Formulaire de contact** : Simple formulaire de saisie sans traitement automatisé des demandes

## Canaux traditionnels

Les clients peuvent contacter l'entreprise via :

- **Téléphone** : Disponible uniquement pendant les heures d'ouverture
- **Email** : Délai de réponse pouvant atteindre 24-48h
- **Réseaux sociaux** : Présence sur Facebook, Instagram et WhatsApp sans système de messagerie automatisé

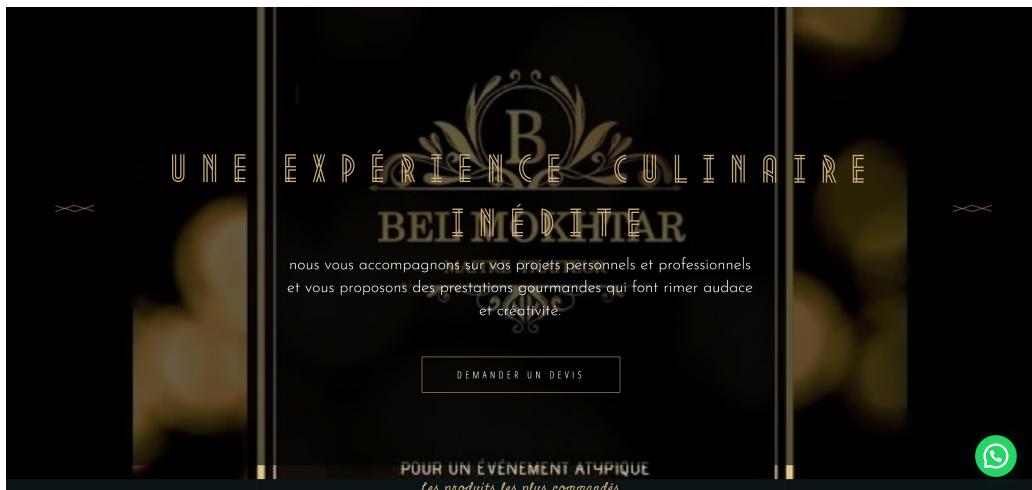


FIGURE 3.1 – Interface du site web

### 3.1.2 Benchmark des solutions similaires sur le marché internationale

L'analyse des solutions concurrentes révèle plusieurs approches d'assistants virtuels dans le secteur de la restauration :

## Chatbots basiques

- **Fonctionnalités** : Réponses préprogrammées à des questions fréquentes (horaires, adresse)
- **Limites** : Incapacité à gérer des requêtes complexes ou personnalisées
- **Exemples** : Chatbots de réservation pour restaurants (LaFourchette, Deliveroo)

## Assistants avancés avec IA

- **Fonctionnalités** : Compréhension du langage naturel, recommandations personnalisées
- **Technologies** : Intégration de NLP et parfois de systèmes RAG
- **Exemples** : Chatbots de service client pour grandes chaînes (Starbucks, McDonald's)

### 3.1.3 Analyse des technologies existantes

Plusieurs approches technologiques sont disponibles pour implémenter notre solution :

#### Solutions de chatbot traditionnelles

- **Chatbots basés sur règles** : Dialogflow, Microsoft Bot Framework
- **Avantages** : Faciles à implémenter pour des cas simples
- **Inconvénients** : Peu flexibles, nécessitent une maintenance manuelle importante

### 3.1.4 Synthèse de l'étude de l'existant

L'analyse des solutions actuelles de Belmokhtar Traiteur révèle un manque criant d'automatisation dans la gestion des interactions clients. Les canaux existants sont chronophages et ne répondent pas aux attentes modernes de disponibilité et de personnalisation. Le benchmark des solutions similaires montre que l'approche RAG combinée à des LLM représente la solution la plus adaptée pour notre contexte, permettant à la fois :

- De fournir des réponses précises basées sur les données réelles du client
- D'offrir une expérience conversationnelle naturelle et personnalisée
- De s'intégrer facilement à l'écosystème WordPress existant

Cette étude confirme donc la pertinence de notre approche technique et justifie le développement d'un assistant intelligent basé sur l'architecture RAG pour répondre aux besoins spécifiques de Belmokhtar Traiteur.

## 3.2 Proposition de la Solution et Architecture

Après avoir parcouru les solutions les plus utilisées dans le domaine de l'assistance virtuelle pour la restauration, nous proposons une application de nouvelle génération conçue pour révolutionner l'accès aux services de catering. Elle utilise une approche centrée principalement sur l'utilisateur, offrant une expérience à la fois fiable, personnalisée et intuitive.

### Technologies utilisées

Notre application utilisera :

- **Large Language Model (LLM)** : Un modèle de langage avancé capable de comprendre et générer du langage naturel de manière fluide.
- **Retrieval-Augmented Generation (RAG)** : Une technologie qui combine la puissance du LLM avec l'accès aux données spécifiques de Belmokhtar Traiteur, garantissant des réponses à la fois précises et contextuellement pertinentes.

### Ce qui nous rend unique

Pour se distinguer des solutions existantes, on propose plusieurs fonctionnalités qui nous rendent uniques :

- **Conversation naturelle** : Discutez de vos besoins en restauration comme vous le feriez avec un conseiller. Notre assistant comprend le langage courant et les nuances de vos demandes.
- **Informations fiables** : Grâce à la technologie RAG, chaque réponse est étayée par les données actualisées de Belmokhtar Traiteur.
- **Recommandations personnalisées** : L'assistant vous aide à formuler vos demandes, vous guide dans le choix des menus et vous propose des solutions adaptées à votre événement.

Pour mettre en place cette solution, on présente tout d'abord l'architecture de l'application qui représente les composantes internes ainsi que le fonctionnement et la succession des actions dans notre solution.

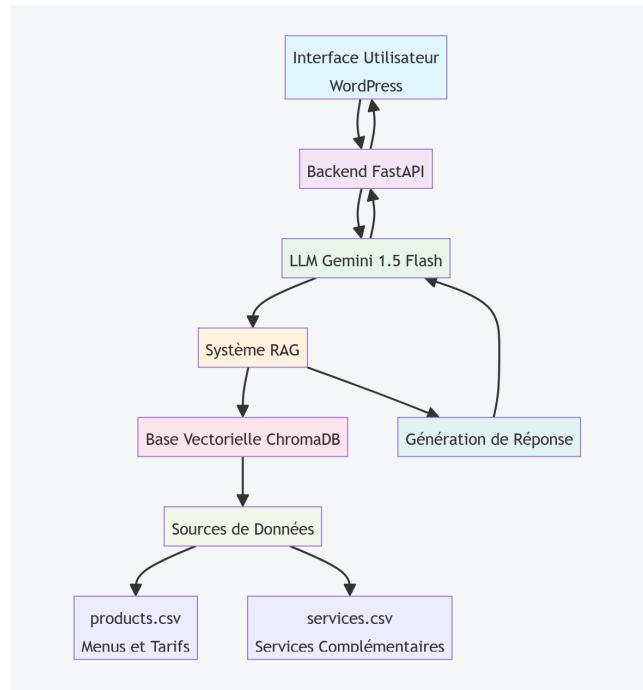


FIGURE 3.2 – Architecture de l’Application

Notre architecture est bien conçue pour répondre aux besoins des utilisateurs selon la succession suivante :

- L’**Interface Utilisateur** constitue le point de contact entre l’utilisateur et le système, développé en tant qu’application web intégrée à WordPress, qui permet à l’utilisateur d’écrire librement son besoin ou de poser des questions en langage naturel, puis affiche des réponses selon le cas de cet utilisateur, et permet ainsi des fonctionnalités comme l’historique des conversations ou des suggestions de questions types en un format intuitif et interactif.
- En **Backend**, on gère les flux de données entre les différentes composantes, puisqu’il reçoit des requêtes de l’interface utilisateur, il gère l’authentification des utilisateurs en cas de besoin pour personnaliser l’expérience et permet le stockage des conversations dans la base de données. Ainsi, il est responsable du déploiement dans le serveur cloud.
- le**LLM** constitue le coeur de l’application puisqu’il est responsable de comprendre la requête client et de la reformuler pour optimiser la recherche du **RAG** qui ajoute les données spécifiques de Belmokhtar Traiteur et assure la réponse à l’utilisateur

### 3.3 Sources de Données

Pour garantir la fiabilité et l’actualisation des informations, le Groupe HS & Belmokhtar Traiteur nous a directement fourni l’ensemble des données nécessaires, notamment les fichiers

structurés **products.csv** et **services.csv** contenant les menus détaillés, tarifs et options de services. Cette collaboration étroite avec le client assure que notre assistant virtuel s'appuie sur des informations officielles, précises et régulièrement mises à jour, répondant ainsi à l'exigence de crédibilité essentielle dans le domaine commercial de la restauration événementielle.

ID	Type	SKU	Nom	Catégories	Tags	Description	Regular price	Regular price numeric	Sale price numeric	In stock?	Stock level
7595	simple	Uniquer_Cocktail	Buffet	"Lorem ipsum sit amet, consectetur adipisciing elitsi edo eiusmod senteas tempor incididunt ut labore et dolore magna aliqua."							
7634	simple	Buffet_de_soutenance	Buffet standard	"Buffet, Buffet > Buffet de soutenance"							
7635	simple	Deco	Buffet, Matériel Chic	"Buffet, Le buffet chic apporte une touche d'élégance et de gourmandise à votre événement.."							
7637	simple	Pastilla	Pastilla au	"Cuisine marocaine et internationale,Pastilla, poisson, poulet", "C'est une préparation de deux pastillas"							
7704	simple	Mariage ou Réception	Cuisine marocaine et internationale,Mariage	"Mariage et fiançailles,Viande, poisson d'agneau marinées et rôties, viande ou poisson mariné"							
7785	simple	Marie	Cuisine marocaine et internationale,Mariage	"Mariage et fiançailles,poulet, la plats appétissants pour toutes gastronomiques"							
7799	simple	Pastilla_mouton	Cuisine marocaine et internationale,Mariage et fiançailles	"poulet, la préférence de la cuisine marocaine"							
7796	variable	menu	Classification	"cuisine marocaine et internationale, Marocaine et internationale, Marocaine et fiançailles"							

FIGURE 3.3 – Base de données produits

Notre première source de données est le fichier **products.csv**, qui contient l'ensemble des menus et services proposés par Belmokhtar Traiteur. Cette base de données structurée comprend des informations détaillées sur :

- Les différents types de menus (mariage, entreprise, soutenance)
- Les descriptions détaillées de chaque prestation
- Les tarifs et options disponibles
- Les catégories de services (buffets, réceptions, événements spéciaux)

Ces données sont régulièrement mises à jour par l'équipe de Belmokhtar pour garantir l'actualité des informations proposées aux clients.

nom_service	type_service	résumé_service	info_produit	produits_disponibles	taux_disponibilité	prix_minimum	prix_maximum	prix_moyen
Buffet,Buffet	Service de buffet professionnel avec 3 formules différentes. Parfait pour tous types d'événements, nos buffets s'adaptent à vos besoins.							
Soutenance,Soutenance	Service spécialisé pour soutenances et célébrations académiques avec 11 formules. Nous comprenons l'importance de ces moments importants et nous nous assurons que tout va bien.							
Cuisine Marocaine,Cuisine Marocaine	Service de cuisine marocaine authentique avec 2 spécialités. Découvrez les saveurs traditionnelles et les ingrédients de saison.							
Repas,Repas	Service de repas pour tous types d'événements, de petits déjeuners aux repas de groupe.							
Catering Général,Catering Général	Service de traiteur général avec 5 options variées. Solution complète pour tous vos besoins culinaires.							
Anniversaire,Anniversaire	Service de traiteur pour anniversaires et fêtes privées avec 1 option. Créez des souvenirs mémorables avec nous.							
Pâtisserie,Pâtisserie	Service de pâtisserie fine avec 2 créations. Nos chefs pâtissiers créent des desserts exceptionnels pour sublimer vos événements.							

FIGURE 3.4 – Base de données services

Une deuxième source de données est le fichier **services.csv**, qui contient des informations complémentaires sur les services proposés :

- Les options de livraison et de service
- Les conditions particulières pour chaque type d'événement
- Les disponibilités et contraintes horaires
- Les informations sur le personnel et l'équipement nécessaire

Ces bases offrent l'avantage d'être fiables et mises à jour régulièrement, ce qui assure à notre assistant de fournir des réponses sourcées et pertinentes, tout en minimisant les hallucinations du modèle IA. Cette approche correspond parfaitement aux objectifs de notre projet.

### 3.4 LLM Choisi



FIGURE 3.5 – Enter Caption

Le choix judicieux du modèle de base du LLM constitue aussi une étape cruciale pour le développement de notre assistant virtuel. Certes il existe une variété de modèles capables de gérer efficacement les besoins et les dépendances de notre projet, mais on a dû mettre plusieurs critères pour enfin de compte choisir le plus idéal et adapté à notre solution. Pour se faire, on devait éliminer les LLM qui ne sont pas disponibles gratuitement pour notre application. Notre évaluation prenait en compte non seulement les performances brutes, mais aussi la facilité d'intégration avec l'architecture RAG et les exigences matérielles pour s'en sortir avec le tableau comparatif suivant :

Caractéristiques	GEMINI 1.5 FLASH	PHI-3-MINI	LLAMA 3.2 1B INSTRUCT	MISTRAL 7B
Taille du modèle	Cloud (API)	3.8B params	1.0B params	7B params
Architecture	Transformer	Decoder-only	Decoder-only	Transformer
Données d'entraînement	Multimodal, Web	Web, Instructions	Web, Common Crawl, Code	Web, Code
Spécialisation catering	Bonne	Limitée	Bonne	Moyenne
Performance générale	Excellente	Bonne	Bonne	Bonne
Capacités conversationnelles	Excellente	Très bonnes	Très bonnes	Bonnes
Compréhension contextuelle	Excellente	Bonne	Bonne	Bonne
Facilité d'utilisation	Excellente (via API)	Bonne	Bonne	Moyenne
Quantification supportée	Non (Cloud)	Bonne (4 <sup>4</sup> /QL8-Ra)	Bonne (Q4/Q8)	Bonne (Q4/Q8)
Intégration RAG	Excellente (via prompt)	Bonne	Bonne	Bonne

TABLE 3.1 – Comparaison des caractéristiques des LLMs

D'après les résultats fournis dans le tableau, le choix est tombé sur le modèle **Google Gemini 1.5 Flash** qui représente le choix optimal pour notre assistant virtuel. C'est un grand modèle multimodal développé par Google, capable de comprendre texte, code et contexte commercial. Ainsi, il n'exige pas de matériel lourd ni de serveurs locaux ce qui fait de lui un modèle idéal pour les projets avec peu de ressources.

L'intégration de ce modèle avec une architecture RAG bien conçue permettra d'obtenir un assistant capable de fournir des informations précises sur les services de restauration, contextuellement pertinentes et actualisées, tout en maintenant une interaction conversationnelle naturelle avec les utilisateurs.

## Conclusion

Cette analyse comparative des assistants virtuels existants pour la restauration montre que si les solutions actuelles apportent un réel soutien dans la gestion des demandes clients, elles restent limitées par des approches algorithmiques rigides et des bases de connaissances statiques. Notre projet propose une alternative innovante en intégrant les LLMs avec la technologie RAG. Cette combinaison permet d'offrir des réponses précises, sourcées et formulées en langage naturel, tout en évitant les hallucinations grâce à l'appui des données spécifiques de Belmokhtar Traiteur. Ainsi, l'utilisation du LLM de Gemini nous permet de présenter une interface intuitive, un système de personnalisation et des fonctionnalités adaptées, notre assistant virtuel se positionne comme une solution accessible, performante et orientée utilisateur. Il constitue une base solide pour le futur des outils d'assistance client dans le secteur de la restauration.

# Chapitre 4

## Réalisation

### Introduction

Après avoir défini les théories de notre solution et son architecture générale dans les chapitres précédents, nous attaquons maintenant à la partie réalisation, où nous allons explorer l'implémentation concrète de notre application d'assistant virtuel pour la restauration basé sur les LLM et la RAG. Dans ce chapitre, nous allons détailler les étapes de mise en œuvre technique de la solution.

### 4.1 Technologies Utilisées

La réalisation de notre solution d'assistant virtuel pour la restauration repose sur plusieurs technologies, soigneusement sélectionnées pour permettre la fiabilité des informations sur les services de catering avec une bonne performance selon les ressources disponibles, une interactivité via une interface web intuitive et une intégration transparente avec WordPress.

#### 4.1.1 Collecte des Données

La phase de collecte a été réalisée à partir des sources fournies directement par Belmokhtar Traiteur, qui garantissent des informations officielles et actualisées sur les produits et services. Pour traiter ces données, nous avons utilisé :

## Pandas



FIGURE 4.1 – Pandas

**Pandas** est une bibliothèque Python spécialisée dans la manipulation et l'analyse de données. Elle permet de lire, traiter et structurer efficacement les fichiers CSV contenant les produits et services de Belmokhtar Traiteur. Cette bibliothèque est utilisée pour nettoyer les données, les standardiser et les préparer pour l'indexation dans notre système RAG.

### 4.1.2 Embedding et Indexation

Afin de pouvoir rechercher efficacement les produits et services pertinents en fonction des questions posées, nous générerons des embeddings textuels pour chaque élément du catalogue.

#### Modèle d'Embedding : all-MiniLM-L6-v2

**all-MiniLM-L6-v2** est un modèle d'embedding de la série **Sentence-BERT** utilisé depuis la bibliothèque **sentence-transformers** de Python. Il mappe des phrases et des paragraphes à un espace vectoriel dense de 384 dimensions et peut être utilisé pour des tâches telles que le clustering ou la recherche sémantique. Il est très rapide, compatible CPU, et offre de bonnes performances pour notre RAG appliqué à la restauration.

### Base Vectorielle : ChromaDB



FIGURE 4.2 – ChromaDB

**ChromaDB** est une base vectorielle open-source, facile à utiliser localement. Elle permet de stocker des embeddings et de faire des recherches par similarité. Dans notre contexte, elle permet de trouver les produits et services pertinents quand l'utilisateur pose une question, ce qui est essentiel pour le RAG.

#### 4.1.3 Génération de Réponse



FIGURE 4.3 – Google AI Studio

La génération de réponse se fait via **Google Gemini Flash**, un modèle de langage large accessible via **API**, obtenu depuis **Google AI Studio**, une plateforme qui permet aux développeurs d'explorer et d'intégrer les modèles d'intelligence artificielle de Google, notamment la famille Gemini.

#### 4.1.4 Développement de l'Interface Utilisateur

Pour rendre l'assistant accessible et interactif, nous avons développé une interface web intuitive intégrée à WordPress, rassemblant simplicité et fonctionnalités pour présenter l'ensemble des fonctionnalités de l'assistant.

## Intégration WordPress



FIGURE 4.4 – WordPress

L'intégration dans WordPress se fait via un plugin personnalisé développé spécifiquement pour Belmokhtar Traiteur. Ce plugin permet d'ajouter un panneau de chat flottant sur toutes les pages du site, offrant une expérience utilisateur transparente sans nécessiter de navigation vers une page externe.

## 4.2 Présentation de la Solution

L'intégration des différentes technologies présentées précédemment nous a permis de concevoir une solution web simple d'utilisation, bien structurée et intuitive. Celle-ci met en avant l'ensemble des fonctionnalités développées, chacune étant clairement expliquée pour être accessible à tout utilisateur souhaitant obtenir des informations sur les services de restauration.

### 4.2.1 Collecte de Données, Vectorisation et Pipeline du RAG

Avant d'aborder les fonctionnalités clés de notre système, il est essentiel de présenter la base de données utilisée, ainsi que les étapes de vectorisation et d'indexation qui ont permis de rendre le système performant et précis.

## Collecte des données produits et services

La collecte des données constitue une étape fondamentale dans la mise en place de notre assistant virtuel. Nous avons travaillé directement avec les fichiers fournis par Belmokhtar Traiteur :

- **Étape 1 : Traitement du fichier products.csv** : Nous avons commencé par analyser la structure du fichier products.csv contenant l'ensemble des produits et services proposés par Belmokhtar Traiteur. Ce fichier contient des informations détaillées sur les menus, tarifs, descriptions et catégories.
- **Étape 2 : Nettoyage et structuration des données** : Pour chaque produit, nous avons extrait les informations essentielles : nom, description, catégories, tags, prix et disponibilité. Nous avons créé une colonne **rag\_description** optimisée pour le traitement RAG.
- **Étape 3 : Préparation pour l'indexation** : Les données structurées ont été préparées pour être converties en embeddings et stockées dans la base vectorielle.

Une fois les données traitées, nous les avons structurées pour garantir leur qualité et leur utilité dans le moteur RAG. Nous avons obtenu en fin de compte un fichier qui contient pour chaque produit une description optimisée pour la recherche sémantique.

## Vectorisation et Création de la base de données vectorielle

Afin de rendre ces données exploitables pour le modèle, nous les avons converties en représentations vectorielles (embeddings) grâce au modèle **sentence-transformers/all-MiniLM-L6-v2** qui représente une taille d'Embedding de 384 dimensions et qui permet l'encodage de chaque produit/service en vecteurs numériques.

Puis, nous avons créé la base de données vectorielle qui permet de rechercher rapidement les produits et services les plus pertinents en fonction de la requête utilisateur. Pour cette tâche, nous avons utilisé **ChromaDB** en tant que solution légère et rapide. Elle permet de stocker les embeddings et de faire des requêtes de similarité efficacement.

## Intégration dans le pipeline RAG

Le moteur RAG utilise désormais cette base vectorielle pour fournir des réponses sourcées à l'utilisateur. En effet, lorsqu'un utilisateur pose une question sur les services de restauration, le système **encode la question en embedding, recherche les k documents les plus similaires (top-k retrieval)**, dans notre cas  $k=3$  pour retrouver le contexte et **envoie ces**

**documents comme contexte au modèle Gemini Flash pour génération de réponse en utilisant l'API.** Gemini reçoit alors la requête de l'utilisateur ainsi que les produits trouvés, ce qui lui permet de générer une réponse précise, sourcée, et contextuelle.

Grâce à cette chaîne de traitement, nous disposons d'un catalogue riche, structuré et indexé, permettant à l'assistant virtuel de fournir des réponses fiables, rapides et basées sur les informations officielles de Belmokhtar Traiteur.

#### 4.2.2 Présentation de l'Application

Après avoir expliqué le fonctionnement général de l'application qui comprend les sources de données, la mise en place de la base données vectorielle et de la pipeline du RAG et du LLM qui cherche le contexte de la requête de l'utilisateur depuis la base de données pour trouver des similarités et générer les réponses, nous présentons ici l'interface web intuitive qui permet à l'utilisateur final d'interagir facilement avec cet assistant virtuel.

L'ensemble de ces traitements est géré depuis une interface web captivante et réactive qui présente toutes les fonctionnalités dans un enchaînement logique et simple d'utilisation, tout en s'intégrant parfaitement au site existant de Belmokhtar Traiteur.

#### Interface du Chatbot Intégré

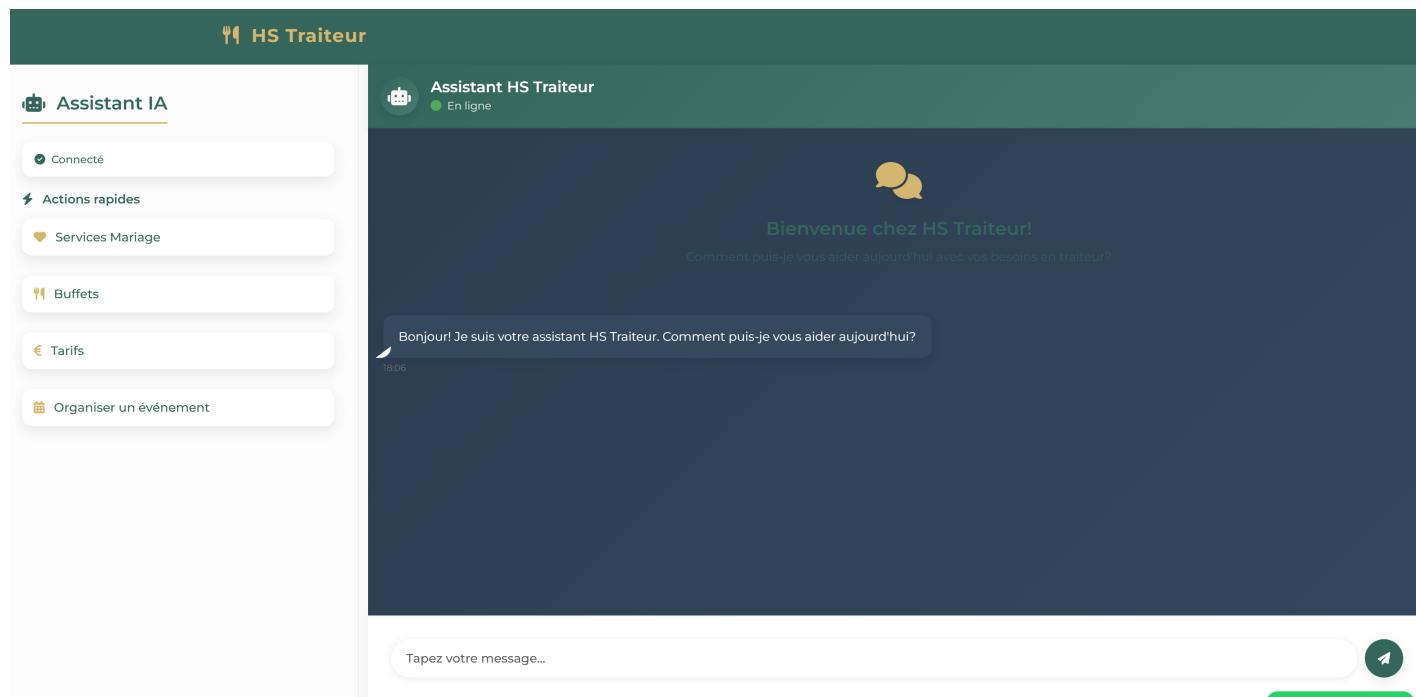


FIGURE 4.5 – Interface du chatbot intégrée au site web

L'interface du chatbot est conçue pour être accessible depuis n'importe quelle page du site web de Belmokhtar Traiteur. Un bouton flottant, représenté par une icône de chatbot, est positionné en bas à droite de l'écran. Ce bouton reste visible lors du défilement de la page, permettant aux utilisateurs d'acc

Lorsque l'utilisateur clique sur l'icône du chatbot, le panneau s'ouvre pour révéler l'interface de conversation. Cette interface permet aux clients de :

- Poser des questions en langage naturel sur les menus, services et tarifs
- Recevoir des recommandations personnalisées selon le type d'événement
- Obtenir des réponses précises basées sur le catalogue actualisé de Belmokhtar
- Voir les sources des informations (produits ou services concernés)

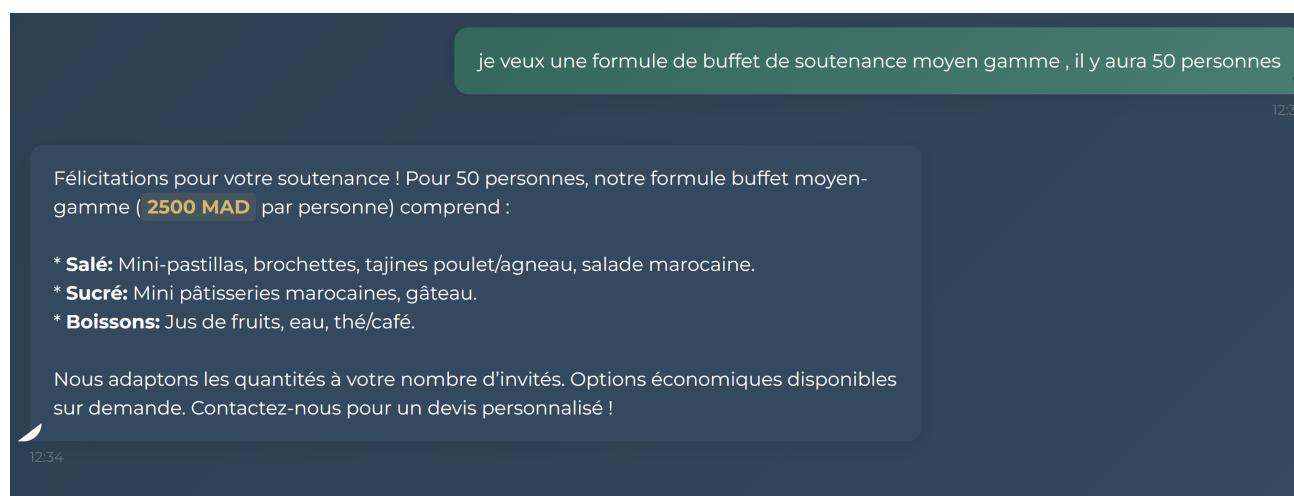


FIGURE 4.6 – Interface de conversation du chatbot

## Fonctionnalités Clés du Chatbot

Notre assistant virtuel intègre plusieurs fonctionnalités essentielles pour améliorer l'expérience client :

**Recommandations Personnalisées** Le chatbot est capable de fournir des recommandations personnalisées en fonction du type d'événement, du nombre de personnes et du budget. Par exemple, pour un mariage de 50 personnes, le système peut suggérer le menu complet classique avec les options appropriées.

**Gestion des Demandeurs d'Information** Le chatbot qualifie automatiquement les leads en collectant les informations essentielles :

- Type d'événement (mariage, entreprise, soutenance, etc.)
- Nombre de convives

- Budget approximatif
- Date souhaitée

Ces informations sont ensuite transmises à l'équipe commerciale de Belmokhtar Traiteur pour un suivi personnalisé.

**Transfert vers WhatsApp** Pour les demandes complexes nécessitant une intervention humaine (que ça soit une question complexe ou le client affirme directement qu'il veut contacter un des responsables), le chatbot propose un transfert direct vers WhatsApp où un conseiller client prendra en charge la demande.

**Indicateur de Saisie** L'interface inclut un indicateur de saisie animé (trois points qui se déplacent) lorsque le chatbot prépare sa réponse, améliorant ainsi l'expérience utilisateur en donnant un retour visuel immédiat.

**Support du Markdown** Le chatbot prend en charge le markdown pour afficher les réponses de manière enrichie, avec mise en forme du texte, listes, et autres éléments visuels pour une meilleure lisibilité.

## Intégration Technique avec WordPress

L'intégration technique du chatbot avec WordPress se fait via un plugin personnalisé développé spécifiquement pour ce projet. Ce plugin assure :

- **Chargement asynchrone** : Le chatbot se charge sans ralentir le reste du site
- **Communication sécurisée** : Les échanges entre le frontend et le backend sont cryptés
- **Compatibilité** : Le plugin est compatible avec la plupart des thèmes WordPress
- **Mises à jour** : Le contenu du chatbot se met à jour automatiquement lorsque les données produits sont modifiées

## Conclusion

Ce chapitre a permis de présenter en détail les différentes fonctionnalités clés de notre assistant virtuel pour la restauration, basé sur une architecture RAG (Retrieval-Augmented Generation) efficace. L'application développée offre aux clients de Belmokhtar Traiteur une interface intuitive et interactive pour poser des questions sur les menus et services, obtenir

des réponses précises basées sur le catalogue officiel, et bénéficier de recommandations personnalisées. Chaque fonctionnalité intègre des outils interactifs tels que la recherche sémantique, les suggestions contextuelles et le transfert vers WhatsApp pour les demandes complexes, en mettant l'accent sur la précision des informations et l'amélioration de l'expérience client, garantissant ainsi une utilisation efficace et professionnelle.

# Conclusion Générale

L'accès aux informations fiables sur les services de restauration constitue un enjeu majeur dans l'amélioration de l'expérience client. Dans ce contexte, notre projet a consisté à concevoir un assistant catering basé sur les technologies LLM et RAG, permettant de répondre aux questions liées aux services de restauration avec des sources vérifiées et une interface intuitive. L'objectif principal était de combiner ces outils pour offrir aux clients une assistance personnalisée, tout en optimisant les processus internes de l'entreprise. Pour atteindre cet objectif, nous avons mis en place une architecture modulaire et efficace. La collecte des données s'est appuyée sur les bases de produits et services de Bel Mokhtar Catering. Ces documents ont ensuite été indexés grâce au modèle d'embedding léger `all-MiniLM-L6-v2`, puis stockés dans une base vectorielle locale via ChromaDB. Ce système RAG a permis de générer des réponses sourcées via le LLM **Google Gemini Flash**, garantissant ainsi une réponse cohérente, précise et présentant des informations validées. L'application finale, développée avec une intégration **WordPress**, propose une interface web interactive comprenant plusieurs fonctionnalités : pose de questions sur les menus et services, historique des échanges, recommandations personnalisées et transfert vers WhatsApp pour une assistance humaine. Durant la réalisation, plusieurs défis ont été relevés, notamment l'intégration efficace des données produits et services, la personnalisation des recommandations selon le type d'événement et le nombre de convives, l'intégration fluide de l'historique conversationnel dans les prompts envoyés au LLM et l'utilisation d'un modèle d'intelligence artificiel performant mais nécessitant peu de ressources. Ces obstacles ont conduit à l'intégration de fonctions de nettoyage des données, de gestion des sessions utilisateurs et à une interface bien structurée et accessible. À travers ces étapes, notre assistant catering a démontré sa capacité à fournir des réponses sourcées et à simplifier l'accès à l'information sur les services de restauration, tout en restant transparent quant à ses limites. Enfin, cette application ouvre la voie à plusieurs perspectives futures que nous souhaitons intégrer telles que son déploiement en ligne pour une utilisation multi-utilisateurs, l'ajout de nouvelles sources de données (avis clients, tendances culinaires), le développement d'un système de gestion cloud des profils uti-

lisateurs pour plus de personnalisation et l'utilisation d'un LLM plus performant. Ainsi, nous avons l'intention de réaliser une version mobile de l'assistant pour un usage quotidien facilité, une amélioration du traitement multilingue notamment la langue arabe et une intégration plus poussée des systèmes de recommandation basés sur les préférences alimentaires.

Ce projet illustre comment une solution d'intelligence artificielle peut être conçue de manière responsable, légère et efficace, tout en restant centrée sur les besoins réels des utilisateurs dans le domaine de la restauration.

# Références Bibliographiques

- [1] *Transformers in Machine Learning*, [https://www.geeksforgeeks.org/getting-started-with-transformers/?ref=header\\_outind](https://www.geeksforgeeks.org/getting-started-with-transformers/?ref=header_outind), Consulté le : 10 mars 2025.
- [2] *What is Retrieval-Augmented Generation (RAG) ?* <https://www.geeksforgeeks.org/what-is-retrieval-augmented-generation-rag/>, Consulté le : 16 mars 2025.
- [3] *Understanding RAG : 6 Steps of Retrieval Augmented Generation (RAG)*, <https://www.acorn.io/resources/learning-center/retrieval-augmented-generation/>, Consulté le : 16 mars 2025.