

Projet de Fin d'Année

DÉVELOPPEMENT D'UN COBOT POUR L'EXTRACTION ET L'ANALYSE DES FORMATIONS MICROSOFT LEARN



Microsoft

Réalisé par:
Hiba Kharfasse

Encadré par:
M. Hassan Fadili

Dedicace

Kharfasse Hiba

À ma chère mère et à mon cher père,

Pour votre amour inconditionnel, vos sacrifices et votre soutien constant qui m'ont donné la force d'avancer et de croire en mes rêves. Ce travail est le fruit de vos encouragements et de vos prières.

À mes chères sœurs,

Pour leur présence bienveillante, leurs encouragements constants et leur affection sincère m'ont toujours donné la force d'avancer et de croire en moi.

À mes amis proches,

Pour votre présence, vos conseils et vos moments de joie partagés, qui ont rendu ce parcours plus léger et plus agréable.

À mes professeurs et à l'ensemble du corps enseignant,

Pour votre dévouement, votre patience et votre transmission du savoir, qui ont façonné mon esprit et renforcé ma détermination.

À tous mes camarades d'école,,

Pour la collaboration, l'entraide et les souvenirs que nous avons créés ensemble au fil des années.

Remerciements

Je rends grâce à **Dieu Tout-Puissant**, pour m'avoir donné la santé, la force et la détermination nécessaires à la réalisation de ce stage.

Mes sincères remerciements vont à **mes parents et mes sœurs**, pour leur soutien inébranlable, leur patience et leur encouragement constants tout au long de mon parcours.

Je remercie chaleureusement Monsieur **Hassan FADILI**, mon encadrant de stage, pour son accompagnement attentif, ses conseils judicieux et sa disponibilité.

Je tiens également à exprimer ma gratitude à Monsieur **Anas BELEBBES**, Project Owner, pour sa confiance et ses orientations tout au long du projet.

Mes remerciements s'adressent aussi à Monsieur **Youssef GHIGANE**, responsable technique, pour son appui technique et ses précieux conseils qui m'ont permis de progresser.

Je remercie vivement l'ensemble des **professeurs et l'administration** de mon école pour la qualité de la formation dispensée et leur soutien constant. J'exprime également ma profonde gratitude à M. Othman Aghzout, Chef de filière, ainsi qu'à **M. Kamal Reklaoui**, Directeur et Professeur, pour leur accompagnement, leurs conseils précieux et leur engagement envers la réussite des étudiants.

Enfin, merci à mes **amis proches** pour leur présence et leur encouragement durant cette expérience enrichissante.

Résumé

Ce projet consiste à développer un **cobot** intelligent capable de recommander des formations adaptées issues de la plateforme **Microsoft Learn**. Face à la richesse et la complexité des contenus disponibles, ce cobot guide l'utilisateur à travers un dialogue interactif pour identifier ses besoins spécifiques en termes de domaine, niveau et objectifs professionnels. Le projet intègre plusieurs composantes : la collecte automatisée des données via l'**API Microsoft Learn** et le **scraping**, le nettoyage et la structuration des informations, ainsi qu'un traitement avancé du **langage naturel (NLP)** pour analyser les requêtes utilisateurs. Grâce à ces éléments, le cobot propose des **recommandations** personnalisées et actualisées. Cette solution innovante facilite l'accès à formation professionnelle et améliore l'expérience utilisateur en simplifiant le choix parmi un large catalogue de modules et certifications.

Mots-clés : Microsoft Learn, Cobot, Recommandation, API, Scraping, NLP.

Abstract

This project aims to develop an intelligent **cobot** capable of recommending suitable training courses from the **Microsoft Learn** platform. Faced with the richness and complexity of the available content, this cobot guides the user through an interactive dialogue to identify their specific needs in terms of domain, level, and professional objectives. The project integrates several components : automated data collection via the Microsoft Learn **API** and **scraping**, cleaning and structuring of the information, as well as advanced natural **language processing (NLP)** to analyze user queries. Thanks to these elements, the cobot provides personalized and up-to-date **recommendations**. This innovative solution facilitates access to professional training and enhances the user experience by simplifying the choice among a large catalog of modules and certifications.

Keywords : Microsoft Learn, Cobot, Recommendation, API, Scraping, NLP.

Table des matières

Dedicace Kharfasse Hiba	1
Remerciements	2
Résumé	3
Abstract	4
Liste des Abréviations	9
Introduction générale	10
1 Contexte organisationnel et technologique	11
1.1 Présentation de Dyn IT Maroc	11
1.1.1 Domaines d'expertise	11
1.1.2 Chiffres et implantations	12
1.1.3 Valeurs et mission	12
1.1.4 Méthodologie de management de projets chez Dyn IT Maroc	12
1.2 Aperçu synthétique de Big Data et Intelligence Artificielle (IA)	13
1.2.1 Définition de Big Data	13
1.2.2 Types de données	13
1.2.3 Définition de l'Intelligence Artificielle (IA)	14
1.2.4 Types d'Intelligence Artificielle (IA)	14
1.2.5 Concepts fondamentaux	15
1.2.6 Les défis auxquels sont confrontées les technologies de l'information	16
2 Analyse de la plateforme et des besoins	18
2.1 Présentation de Microsoft Learn	18
2.1.1 Qu'est-ce que Microsoft Learn ?	18
2.1.2 Origines et évolution	18
2.1.3 Objectifs et mission	19
2.1.4 Mode de fonctionnement	19
2.1.5 Types de contenus disponibles	19
2.2 Problématique et difficultés rencontrées par les utilisateurs	20
2.2.1 Volume important et diversité des formations	20
2.2.2 Limites des méthodes classiques de recherche	20
2.2.3 Difficultés d'accès rapide à une formation pertinente et personnalisée	21
2.3 Expression des besoins et objectifs du projet	22
2.3.1 Besoin d'un outil intelligent d'assistance à la recherche	22

2.3.2	Critères fonctionnels attendus	22
2.3.3	Critères techniques attendus	23
3	Proposition de solution et architecture	24
3.1	Proposition de solution : Robot de recommandation Microsoft Learn	24
3.1.1	Description générale de la solution proposée	24
3.1.2	Rôle du cobot dans l'accompagnement des utilisateurs	25
3.1.3	Avantages attendus par rapport aux méthodes classiques	25
3.2	Architecture fonctionnelle et technique	26
3.2.1	Schéma fonctionnel	26
3.2.2	Architecture technique	27
3.2.3	Gestion de la sécurité et de la confidentialité	28
3.3	Technologies utilisées	29
4	Structuration et enrichissement du fichier Excel	33
4.1	Collecte via l'API Microsoft Learn	33
4.1.1	Connexion à l'API officielle	33
4.1.2	Téléchargement des données brutes au format JSON	34
4.2	Nettoyage et enrichissement de base	34
4.2.1	Suppression des colonnes inutiles	34
4.2.2	Colonnes conservées	35
4.2.3	Suppression des contenus retirés (retired)	35
4.2.4	Scraping des informations manquantes	35
4.2.5	Création de la colonne certified	36
4.3	Remplissage automatique des sujets manquants	36
4.3.1	TF-IDF (Term Frequency – Inverse Document Frequency)	37
4.3.2	Régression Logistique pour la classification multi-étiquettes	37
4.4	Calcul de la popularité des contenus	37
4.4.1	Calcul de la popularité des sujets	37
4.4.2	Normalisation des variables	37
4.4.3	Pondération et score final	38
4.5	Identification et ajout de la dimension Technologie	38
4.5.1	Constitution d'un dictionnaire de technologies	38
4.5.2	Extraction des technologies dans les textes	38
4.6	Export final du dataset	39
5	Intégration d'OpenAI et Développement de l'Interface du Robot Microsoft Learn	41
5.1	Architecture générale de l'application	41
5.1.1	Fonctionnalités principales et ergonomie	41
5.1.2	Présentation de la structure globale du projet	42
5.1.3	Schéma de l'architecture technique	42
5.2	Intégration d'OpenAI	44
5.2.1	Mise en place et configuration de l'API	44
5.2.2	Gestion des clés d'accès et sécurité	46
5.3	Implémentation technique	47
5.3.1	Conception en HTML, CSS et JavaScript	47
5.3.2	Workflow de communication entre les composants	48

Table des matières

5.4	Tests et validation de l'application	48
5.4.1	Méthodologie de test adoptée	48
6	Démonstration et illustrations du projet	50
	Conclusion Générale	57

Table des figures

1.1	Dyn IT Logo	11
1.2	Microsift Logo	11
2.1	Microsoft Learn Logo	18
3.1	Schéma Flux d'interaction	26
3.2	Python logo	29
3.3	Excel logo	30
3.4	OpenAI logo	31
3.5	js/css/html logo	32
4.1	Extrait de Dataset	39
5.1	Architecture technique	43
5.2	flask web development framework	44
5.3	Schéma de Flux de traitement des données	46
6.1	Page d'accueil	50
6.2	Fonctionnalités Clés	51
6.3	À propos de nous	51
6.4	Contactez-nous	52
6.5	Interface du Chatbot	53
6.6	Résultats de recherche	53
6.7	Tableau de bord	54
6.8	Visualisation des données – Répartition par niveau et par type	54
6.9	Analyse des données – Popularité moyenne par type et durée totale par niveau	55
6.10	Analyse des données – Top technologies et sujets	55
6.11	Tableau des données filtrées – Détails des modules de formation	56

Liste des Abréviations

AGI : Artificial General Intelligence

AI : Artificial Intelligence

API : Application Programming Interface

ASI : Artificial Super Intelligence

AWS : Amazon Web Services

CSS : Cascading Style Sheets

GDPR : General Data Protection Regulation

HTML : HyperText Markup Language

HTTPS : HyperText Transfer Protocol Secure

IOT : Internet of Things

IT : Information Technologo

JSON : JavaScript Object Notation

JS : JavaScript

KPI : Key Performance Indicator

NLP : Natural Language Processing

REST : Representational State Transfer

RGPD : Règlement Général sur la Protection des Données

ROI : Return On Investment

SQL : Structured Query Language

UID : User Identifier

URL : Uniform Resource Locator

XML : eXtensible Markup Language

Introduction générale

Dans un monde en constante évolution, marqué par la transformation numérique et l'émergence de nouvelles technologies, la formation continue et l'acquisition de compétences actualisées représentent des enjeux majeurs pour les étudiants, les professionnels et les entreprises. Les plateformes d'apprentissage en ligne, à l'instar de **Microsoft Learn**, se sont imposées comme des solutions incontournables en mettant à disposition une vaste bibliothèque de modules, parcours d'apprentissage et certifications couvrant des domaines variés.

Cependant, **la richesse et la diversité des contenus** disponibles peuvent parfois constituer une difficulté pour les apprenants, qui peinent à identifier rapidement les formations correspondant à leurs besoins et objectifs. D'où la nécessité de concevoir des outils intelligents capables de guider l'utilisateur et de simplifier son parcours d'apprentissage.

C'est dans ce cadre que s'inscrit le présent projet de stage, qui consiste à développer **un cobot de recommandation** dédié à la plateforme Microsoft Learn. Ce cobot, basé sur **l'intelligence artificielle**, le traitement du **langage naturel (NLP)** et les modèles avancés d'**OpenAI GPT**, a pour objectif d'analyser les besoins des utilisateurs à travers des interactions en langage naturel et de leur proposer des formations personnalisées et pertinentes.

Afin de rendre ce système accessible et interactif, **une interface développée avec HTML, CSS et JavaScript** a été mise en place. Elle regroupe deux composantes principales :

1. **Un cobot conversationnel**, permettant à l'utilisateur d'interagir directement pour obtenir des recommandations de formations.
2. **Une partie de visualisation des données**, offrant des représentations graphiques (par exemple : répartition des formations par domaine, par niveau, ou par durée) pour mieux comprendre l'offre disponible sur Microsoft Learn.

Le projet s'appuie sur plusieurs volets techniques : **la collecte et le traitement automatisé des données** issues de **Microsoft Learn**, l'intégration d'**un moteur GPT** pour la compréhension et la génération de réponses, ainsi que la mise en place d'un **système de visualisation et de recommandation intelligent**. L'approche adoptée combine des techniques de **Big Data**, d'**IA**, et de **NLP**, afin de garantir la pertinence des résultats et la mise à jour régulière des contenus.

Chapitre 1

Contexte organisationnel et technologique

Introduction

Ce premier chapitre offre un aperçu de l'environnement du projet. Nous présenterons d'abord l'entreprise DynIt, où nous avons effectué notre stage de fin d'études et qui met en avant l'utilisation des solutions Microsoft dans ses activités. Ensuite, nous aborderons la problématique de l'apprentissage en ligne, la richesse mais aussi la complexité des contenus proposés sur Microsoft Learn, ainsi que les défis rencontrés par les utilisateurs dans l'identification de formations adaptées à leurs besoins.

1.1 Présentation de Dyn IT Maroc

Dyn IT Maroc est une société marocaine fondée en 2012 spécialisée dans les services informatiques et la formation, avec un fort ancrage dans l'écosystème Microsoft. Microsoft est un leader mondial des technologies, proposant des solutions logicielles et cloud qui transforment le fonctionnement des entreprises. Dyn IT Maroc exploite ces technologies pour accompagner la transformation numérique des organisations, tout en intégrant Microsoft Learn, la plateforme d'apprentissage en ligne de Microsoft. Microsoft Learn permet aux professionnels et aux étudiants de développer leurs compétences techniques, d'obtenir des certifications reconnues et de se tenir à jour avec les dernières innovations Microsoft.



FIGURE 1.1 – Dyn IT Logo



FIGURE 1.2 – Microsoft Logo

1.1.1 Domaines d'expertise

- Collaboration digitale : déploiement et gestion d'Office 365 pour optimiser la communication et la productivité interne.

- Cloud et Data : fourniture de services sur Microsoft Azure (infrastructure virtuelle, services de données) ainsi que création de tableaux de bord interactifs avec Power BI.
- Formation et certification : centre de formation agréé Microsoft (Microsoft Gold Partner), offrant des ateliers animés par des formateurs certifiés, des supports officiels, des labs en ligne et des programmes préparant aux certifications.

1.1.2 Chiffres et implantations

- En activité depuis 2012, l'entreprise se démarque par son expertise pointue dans les technologies Microsoft.
- Implantations à Rabat (Avenue Tadla et Technopark) ou j'ai effectué mon stage, et Casablanca (Technopark).
- Réputation solide : plus de 6 000 personnes formées, de nombreux projets réussis et un taux de satisfaction qui cadre avec la promesse de qualité

1.1.3 Valeurs et mission

Animée par des professionnels expérimentés en IT et en formation, Dyn IT Maroc se donne pour mission de renforcer la collaboration numérique des entreprises et de les accompagner dans leur développement grâce à des solutions technologiques et pédagogiques adaptées.

1.1.4 Méthodologie de management de projets chez Dyn IT Maroc

Dyn IT Maroc applique une approche structurée et agile pour la gestion de ses projets, en particulier ceux liés à la transformation numérique et à l'intégration des technologies Microsoft. Cette méthodologie repose sur les principes suivants :

1. Planification collaborative et rigoureuse

Chaque projet débute par une phase de planification détaillée, impliquant toutes les parties prenantes. Cette étape comprend :

- La définition claire des objectifs et des livrables.
- L'élaboration d'un échéancier précis.
- L'estimation des ressources nécessaires.
- L'identification des risques potentiels et la mise en place de plans de mitigation.

2. Approche agile et itérative

Dyn IT Maroc privilégie une gestion de projet agile, permettant une flexibilité et une adaptation continue aux besoins du client. Cette approche inclut :

- Des cycles de développement courts (sprints).
- Des revues régulières avec les parties prenantes.
- Une réévaluation constante des priorités et des ressources.

3. Suivi et contrôle constants

Tout au long du projet, un suivi rigoureux est effectué pour assurer le respect des délais, du budget et de la qualité. Les outils utilisés comprennent :

- Des réunions de suivi hebdomadaires.
- Suivi des tasks (Devops).
- Des outils de collaboration en ligne pour une communication fluide.

4. Gestion proactive des risques

La méthodologie de Dyn IT Maroc intègre une gestion proactive des risques, avec :

- L'identification précoce des risques potentiels.
- L'élaboration de stratégies de réponse adaptées.
- Une surveillance continue pour détecter et traiter rapidement les problèmes émergents.

5. Clôture et évaluation post-projet

À la fin de chaque projet, une phase de clôture est réalisée, comprenant :

- La validation des livrables avec le client.
- Une évaluation des performances du projet.
- La documentation des leçons apprises pour améliorer les futurs projets.

1.2 Aperçu synthétique de Big Data et Intelligence Artificielle (IA)

1.2.1 Définition de Big Data

Big Data désigne l'ensemble des données volumineuses, variées et générées à grande vitesse par les activités humaines et les systèmes numériques. Ces données dépassent la capacité des outils traditionnels de gestion et d'analyse et nécessitent des technologies spécifiques pour leur collecte, stockage, traitement et exploitation afin d'en extraire des informations utiles et des tendances décisionnelles.

1.2.2 Types de données

Les données peuvent être classées selon leur structure, leur nature et leur utilisation :

1. **Données structurées** : Ce sont des données organisées selon un schéma précis, généralement stockées dans des bases de données relationnelles. Elles sont facilement exploitables par les systèmes informatiques classiques. Exemples : noms, âges, numéros de téléphone, ventes, stocks.
2. **Données semi-structurées** : Ces données ont une organisation partielle mais ne suivent pas un schéma strict. Elles nécessitent des outils spécifiques pour leur traitement. Exemples : fichiers JSON, XML, logs d'activité, e-mails.

3. **Données non structurées** : Ce sont des données brutes sans format prédéfini, difficiles à analyser sans outils avancés. Exemples : texte libre, documents, images, vidéos, fichiers audio, publications sur les réseaux sociaux.
4. **Données temporelles ou séquentielles** : Il s'agit de données enregistrées au fil du temps et représentant une évolution ou une séquence d'événements. Exemples : séries temporelles de capteurs, cours boursiers, historique des ventes.
5. **Données quantitatives et qualitatives** :
 - **Quantitatives** : valeurs numériques mesurables, permettant des calculs et des analyses statistiques (ex. : température, poids, revenus).
 - **Qualitatives** : informations descriptives ou catégorielles, utilisées pour classifier ou caractériser des objets ou des événements (ex. : couleur, type de produit, catégorie sociale).

La classification des données permet de choisir les méthodes et outils d'analyse adaptés, essentiels pour tirer des informations pertinentes et prendre des décisions éclairées.

1.2.3 Définition de l'Intelligence Artificielle (IA)

L'**Intelligence Artificielle (IA)** est un domaine de l'informatique qui vise à créer des systèmes capables de reproduire certaines capacités de l'intelligence humaine, telles que l'apprentissage, le raisonnement, la résolution de problèmes et la prise de décision. L'IA s'appuie sur des données et des algorithmes pour analyser des informations, reconnaître des patterns, prédire des résultats et automatiser des tâches complexes. Ses applications incluent notamment le traitement du langage naturel, la vision par ordinateur, les systèmes de recommandation, et l'automatisation industrielle.

1.2.4 Types d'Intelligence Artificielle (IA)

L'IA peut être classée selon son niveau de capacité ou son domaine d'application :

1. **IA faible (ou étroite / Narrow AI)**
 - Conçue pour accomplir une tâche spécifique de manière efficace.
 - Elle ne possède pas de conscience ni de compréhension générale.
 - L'estimation des ressources nécessaires.
 - Exemples : assistants vocaux (Siri, Alexa), systèmes de recommandation, chatbots.
2. **IA générale (AGI – Artificial General Intelligence)**
 - Capable d'exécuter n'importe quelle tâche intellectuelle qu'un humain peut accomplir.
 - Elle peut apprendre, raisonner et transférer ses connaissances à différents domaines.
 - Elle reste aujourd'hui théorique et n'existe pas encore.
3. **IA super-intelligente (ASI – Artificial Superintelligence)**

- Dépasserait l'intelligence humaine dans tous les domaines.
- Elle est encore hypothétique et fait l'objet de recherches et débats éthiques.

4. Classification selon les techniques utilisées

- **Apprentissage automatique (Machine Learning)** : l'IA apprend à partir des données pour prédire ou classifier.
- **Apprentissage profond (Deep Learning)** : basé sur des réseaux de neurones pour traiter des données complexes comme des images, textes ou sons.
- **Systèmes experts** : utilisent des règles prédefinies pour prendre des décisions.
- **Traitement du langage naturel (NLP)** : permet à l'IA de comprendre et générer le langage humain.
- **Vision par ordinateur** : permet à l'IA d'interpréter et d'analyser des images et vidéos.

5. Clôture et évaluation post-projet

À la fin de chaque projet, une phase de clôture est réalisée, comprenant :

- La validation des livrables avec le client.
- Une évaluation des performances du projet.
- La documentation des leçons apprises pour améliorer les futurs projets.

1.2.5 Concepts fondamentaux

Pour comprendre l'Intelligence Artificielle (IA) et le traitement des données, il est essentiel de connaître certains concepts clés :

Données (Data) : informations brutes collectées à partir de diverses sources, servant de base à l'apprentissage et à l'analyse.

Algorithme : ensemble d'instructions ou de règles utilisées par un système pour traiter les données et résoudre des problèmes.

Apprentissage automatique (Machine Learning) : technique permettant à un système d'apprendre à partir des données sans être explicitement programmé pour chaque tâche.

Apprentissage profond (Deep Learning) : sous-domaine du Machine Learning utilisant des réseaux de neurones pour traiter des données complexes comme des images, des vidéos ou du texte.

Big Data : ensemble de données volumineuses, variées et générées rapidement, nécessitant des technologies spécifiques pour leur traitement et analyse.

Modèle : représentation mathématique ou statistique utilisée par l'IA pour effectuer des prédictions ou des classifications à partir des données.

Prédiction et analyse : utilisation de modèles et algorithmes pour anticiper des comportements, détecter des tendances ou prendre des décisions.

Automatisation : capacité d'un système à exécuter des tâches sans intervention humaine, en se basant sur les données et l'IA.

1.2.6 Les défis auxquels sont confrontées les technologies de l'information

Le domaine des technologies de l'information (IT) fait face à plusieurs défis majeurs liés à l'évolution rapide des technologies et aux besoins croissants des organisations :

1. Gestion et sécurité des données

- Protéger les données contre les cyberattaques, les pertes ou les fuites est un enjeu majeur.
- Assurer la conformité avec les réglementations (ex. : RGPD) est essentiel pour éviter des sanctions.

2. Big Data et traitement des volumes massifs de données

- Collecter, stocker et analyser des volumes de données de plus en plus importants nécessite des infrastructures performantes et des compétences spécialisées.

3. Évolution technologique rapide

- L'IT doit constamment s'adapter aux nouvelles technologies comme le cloud computing, l'intelligence artificielle, l'IoT et la blockchain.

4. Pénurie de compétences

- La demande en spécialistes IT dépasse l'offre, notamment dans les domaines du développement, de la cybersécurité et de l'analyse de données.

5. Interopérabilité et intégration des systèmes

- Les entreprises utilisent des systèmes et logiciels variés. Assurer leur compatibilité et leur intégration fluide est un défi constant.

6. Optimisation des coûts et ROI (Return on Investment)

- Les projets IT nécessitent des investissements importants. Il est crucial d'optimiser les coûts tout en garantissant un retour sur investissement tangible.

7. Adoption et gestion du changement

- L'introduction de nouvelles technologies implique de former les utilisateurs et de gérer la résistance au changement pour garantir le succès des projets.

Conclusion

En résumé, ce premier chapitre a permis de présenter **Dyn IT Maroc**, une entreprise spécialisée dans les services informatiques et la formation, ainsi que son intégration des solutions **Microsoft** et de la plateforme **Microsoft Learn** dans ses activités. Nous avons également défini les notions clés de Big Data et d'Intelligence Artificielle, en mettant en avant leurs types, concepts fondamentaux et défis actuels. Enfin, nous avons introduit la méthodologie de management de projet adoptée par Dyn IT, qui repose sur une planification structurée, une gestion des risques, une communication continue et un suivi rigoureux. Ce chapitre constitue donc une base solide pour comprendre le cadre technologique, stratégique et organisationnel dans lequel s'inscrit notre étude.

Chapitre 2

Analyse de la plateforme et des besoins

Introduction

Après avoir présenté dans le premier chapitre le contexte général du projet, à travers la mise en avant de l'entreprise **Dyn IT Maroc**, de son intégration de **Microsoft Learn** ainsi que des concepts fondamentaux liés au **Big Data**, à l'**Intelligence Artificielle** et à la méthodologie de management de projet, ce deuxième chapitre s'intéresse à l'analyse fonctionnelle et technique du **cobot**. Il s'agit dans un premier temps d'identifier les difficultés rencontrées par les utilisateurs et les limites des méthodes traditionnelles de recherche de contenus de formation. Cette réflexion permettra de mieux cerner les besoins auxquels le cobot doit répondre.

2.1 Présentation de Microsoft Learn

2.1.1 Qu'est-ce que Microsoft Learn ?

Microsoft Learn est une plateforme de formation en ligne gratuite développée par Microsoft, destinée aux utilisateurs finaux, aux développeurs et aux professionnels IT. Elle propose des ressources pédagogiques structurées par spécialisations, parcours d'apprentissage, modules interactifs, exercices pratiques, et certifications.



FIGURE 2.1 – Microsoft Learn Logo

2.1.2 Origines et évolution

- La plateforme Microsoft Learn a été officiellement lancée le 26 septembre 2018, en remplacement progressif de la plateforme "Microsoft Docs", elle-même issue de MSDN et TechNet.
- Quelques semaines avant le lancement officiel, Microsoft a publié un article de blog (4 mai 2016) annonçant la vision de cette plateforme comme un espace centralisé pour l'apprentissage guidé.
- En 2022, l'intégralité des contenus de Microsoft Docs (documentation technique) a été intégrée dans la plateforme Microsoft Learn, renforçant ainsi son rôle en tant que hub global.

2.1.3 Objectifs et mission

L'objectif de Microsoft Learn est de proposer un environnement d'apprentissage autonome, interactif et accessible, correspondant aux rythmes d'apprentissage des apprenants (auto-rythmé), tout en couvrant une approche pratique et concrète des technologies Microsoft.

Parmi les avantages :

- Une apprentissage « learning by doing », qui intègre des environnements de test interactifs (Azure Sandbox).
- Une plateforme multilingue à portée mondiale, accessible sans inscription obligatoire.
- Une approche gamifiée, avec badges, points et niveaux pour encourager la progression des apprenants.

2.1.4 Mode de fonctionnement

La plateforme Microsoft Learn a été conçue pour offrir une expérience d'apprentissage complète et interactive, adaptée aussi bien aux débutants qu'aux professionnels expérimentés. Elle repose sur une approche structurée et progressive, qui accompagne l'utilisateur depuis son inscription jusqu'à la validation de ses acquis. L'objectif est de permettre à chacun de développer ses compétences de manière personnalisée, tout en restant motivé grâce à un système de suivi et de récompenses.

- **Inscription et profil utilisateur** : L'apprenant se connecte avec un compte Microsoft, ce qui permet de sauvegarder ses progrès.
- **Choix d'un domaine ou rôle** : L'utilisateur sélectionne un domaine (cloud, data, IA, développement, cybersécurité...) ou un rôle (data scientist, développeur, administrateur cloud, etc.).
- **Progression guidée** : La plateforme propose un parcours personnalisé avec des modules organisés chronologiquement.
- **Pratique et validation** : Chaque module inclut des quiz, des exercices interactifs et des challenges pratiques pour valider les acquis.
- **Suivi et motivation** : L'utilisateur gagne des points, des badges, et peut suivre son évolution dans le tableau de bord.

2.1.5 Types de contenus disponibles

La plateforme Microsoft Learn propose différents types de contenus adaptés aux besoins des apprenants :

- **Cours** : Unités d'apprentissage indépendantes permettant de se concentrer sur des compétences ou concepts précis.
- **Modules** : Ensembles structurés de leçons et exercices pratiques, souvent conçus pour approfondir un sujet spécifique.

- **Learning Path (Parcours d'apprentissage)** : Parcours guidés combinant plusieurs modules, permettant d'acquérir des compétences complètes sur un domaine ou rôle particulier.
- **Certifications** : Reconnaissance officielle des compétences acquises, délivrée par Microsoft après réussite d'un examen.
- **Examens** : Tests permettant de valider les connaissances et de progresser vers l'obtention d'une certification.

2.2 Problématique et difficultés rencontrées par les utilisateurs

2.2.1 Volume important et diversité des formations

Les utilisateurs de Microsoft Learn peuvent être confrontés à un volume très important de contenus et à une grande diversité de formations, couvrant de nombreux domaines (cloud, data, IA, développement, cybersécurité...). Cette abondance rend parfois difficile la sélection du parcours le plus adapté à leurs besoins et objectifs professionnels. Les apprenants peuvent se sentir perdus face à la multiplicité des modules, learning paths et certifications disponibles, ce qui nécessite une meilleure structuration ou des recommandations personnalisées.

La plateforme Microsoft Learn propose une offre d'apprentissage vaste et diversifiée, comprenant des milliers de contenus répartis sur plusieurs types de formations :

- **Modules** : Plus de **3 000 modules** couvrant des domaines tels que le cloud, la sécurité, les données, le développement, l'IA, etc.
- **Learning Paths (parcours d'apprentissage)** : Environ **300 parcours guidés**, chacun combinant plusieurs modules pour approfondir un sujet spécifique.
- **Certifications** : Plus de **30 certifications officielles**, correspondant à des rôles techniques variés, comme administrateur Azure, ingénieur en sécurité ou analyste de données.
- **Examens** : Plus de **20 examens** associés aux certifications, permettant de valider les compétences acquises.

2.2.2 Limites des méthodes classiques de recherche

Sur Microsoft Learn, les utilisateurs disposent de plusieurs méthodes pour rechercher et accéder aux contenus : filtres, recherche par mot-clé, et navigation manuelle à travers les modules, parcours et certifications. Cependant, ces méthodes présentent des limites importantes lorsqu'il s'agit de traiter le volume massif et la diversité des formations disponibles.

- **Filtres** : Bien qu'utiles pour restreindre les résultats selon des critères comme le niveau, le domaine ou le rôle, ils restent souvent insuffisants pour identifier les contenus les plus pertinents, surtout lorsqu'un utilisateur ne connaît pas exactement le terme ou le concept qu'il recherche.
- **Recherche par mot-clé** : Cette méthode dépend fortement de la formulation exacte des requêtes. Des différences terminologiques ou des synonymes peuvent entraîner des résultats incomplets ou non pertinents, ce qui peut ralentir la progression de l'apprenant.
- **Navigation manuelle** : Explorer les modules ou parcours un par un est chronophage et peu efficace, notamment lorsque l'on souhaite comparer plusieurs options ou trouver des contenus adaptés à un objectif précis.

Ces limites montrent que les méthodes classiques de recherche ne suffisent pas toujours à orienter efficacement l'apprenant dans son parcours, surtout face à la multiplicité et la complexité des contenus proposés. Une approche plus intelligente et personnalisée pourrait ainsi améliorer significativement l'expérience utilisateur et l'efficacité de l'apprentissage.

2.2.3 Difficultés d'accès rapide à une formation pertinente et personnalisée

Malgré la richesse des contenus proposés sur Microsoft Learn, les utilisateurs rencontrent souvent des difficultés à accéder rapidement à des formations réellement pertinentes pour leurs besoins spécifiques. La multitude de modules, parcours et certifications disponibles rend la sélection chronophage, surtout pour les apprenants cherchant à atteindre des objectifs précis dans un délai limité.

De plus, la personnalisation des parcours reste limitée : les méthodes actuelles ne tiennent pas suffisamment compte du niveau initial de l'utilisateur, de ses compétences existantes ou de ses objectifs professionnels. En conséquence, certains apprenants peuvent suivre des modules trop avancés ou trop élémentaires, ce qui entraîne une perte de temps et une motivation réduite.

Ces difficultés sont accentuées par la navigation manuelle et la recherche par filtres ou mots-clés, qui ne permettent pas toujours d'identifier rapidement les contenus les plus adaptés. Ainsi, pour maximiser l'efficacité de l'apprentissage et l'aligner sur le parcours professionnel de chaque utilisateur, il serait nécessaire de disposer de mécanismes de recommandation intelligents et personnalisés, capables de guider l'apprenant vers les formations les plus pertinentes en fonction de ses besoins et de son profil.

2.3 Expression des besoins et objectifs du projet

2.3.1 Besoin d'un outil intelligent d'assistance à la recherche

Face au volume considérable et à la diversité des contenus proposés par Microsoft Learn, les utilisateurs peuvent rencontrer des difficultés pour identifier rapidement les formations les plus pertinentes pour leurs besoins et objectifs professionnels. Les méthodes classiques, telles que la navigation manuelle, les filtres ou la recherche par mots-clés, s'avèrent souvent insuffisantes, surtout lorsqu'un utilisateur ne maîtrise pas exactement les termes ou concepts recherchés. Il apparaît donc nécessaire de disposer d'un outil intelligent d'assistance à la recherche, capable de :

- **Analyser le profil et les besoins de l'utilisateur** (niveau de compétence, domaine d'intérêt, objectif professionnel).
- **Recommander des contenus pertinents et adaptés**, en priorisant les modules ou parcours les plus cohérents avec le profil de l'apprenant.
- **Optimiser le temps et l'efficacité de l'apprentissage**, en guidant rapidement l'utilisateur vers les formations qui apportent une réelle valeur ajoutée.
- **Proposer une expérience personnalisée**, adaptée au rythme, au niveau et aux objectifs spécifiques de chaque utilisateur.

L'introduction d'un tel outil permettrait de transformer l'expérience d'apprentissage, en rendant Microsoft Learn plus accessible et en maximisant la pertinence des contenus pour chaque utilisateur.

2.3.2 Critères fonctionnels attendus

Pour répondre efficacement aux besoins des utilisateurs, l'outil intelligent d'assistance à la recherche doit respecter plusieurs critères fonctionnels essentiels :

1. **Rapidité** : L'outil doit permettre à l'utilisateur d'accéder aux contenus pertinents en un minimum de temps, en réduisant les recherches manuelles et les navigations chro-nophages. Une réponse instantanée ou quasi-instantanée aux requêtes est nécessaire pour garantir une expérience fluide et efficace.
2. **Pertinence** : Les recommandations fournies doivent être cohérentes avec le profil de l'utilisateur et ses objectifs professionnels. L'outil doit prioriser les modules, parcours et certifications les plus adaptés, en évitant les contenus trop avancés ou trop élémentaires. La pertinence des résultats est cruciale pour assurer une progression efficace et motivante pour l'apprenant.
3. **Personnalisation** : L'outil doit être capable d'adapter les recommandations au niveau, aux compétences et aux préférences de chaque utilisateur. Cela inclut la prise en compte des parcours précédemment suivis, des objectifs de carrière, et du rythme d'ap-

apprentissage de l'utilisateur, afin de proposer un parcours sur mesure et évolutif.

En combinant ces trois critères, l'outil d'assistance garantit une expérience d'apprentissage optimisée, permettant à chaque utilisateur de naviguer efficacement dans l'ensemble des contenus Microsoft Learn et d'atteindre ses objectifs professionnels de manière ciblée et rapide.

2.3.3 Critères techniques attendus

Pour garantir le bon fonctionnement et l'efficacité de l'outil intelligent d'assistance à la recherche, plusieurs critères techniques doivent être respectés :

1. Intégration API : L'outil doit pouvoir s'interfacer directement avec les services de Microsoft Learn via API, afin de récupérer en temps réel les informations sur les modules, parcours et certifications. Cette intégration doit être fiable, performante et maintenable, permettant de mettre à jour les contenus et recommandations automatiquement sans intervention manuelle.

2. Accessibilité : L'outil doit être facilement accessible pour les utilisateurs, quel que soit leur environnement de travail (ordinateur, tablette ou smartphone). L'interface doit être ergonomique, intuitive et responsive, permettant une navigation fluide et une interaction simplifiée avec les fonctionnalités de recherche et de recommandation.

3. Sécurité des données : La protection des informations personnelles et des préférences des utilisateurs est primordiale. L'outil doit respecter les normes de sécurité et de confidentialité, notamment en utilisant des mécanismes tels que le stockage sécurisé des clés API via des variables d'environnement, le chiffrement des données sensibles et le respect de la législation sur la protection des données personnelles.

Ces critères techniques garantissent que l'outil d'assistance ne se contente pas d'être fonctionnel, mais qu'il soit également fiable, sûr et accessible, tout en offrant une expérience utilisateur optimale et une intégration transparente avec la plateforme Microsoft Learn.

Conclusion

La solution repose sur un cobot (collaborative robot), un assistant numérique intelligent conçu pour accompagner l'utilisateur dans l'exploration de Microsoft Learn. Ce cobot facilite la recherche de formations pertinentes en proposant des recommandations personnalisées, rapides et adaptées aux besoins de chaque profil.

Chapitre 3

Proposition de solution et architecture

Introduction

Dans ce chapitre, nous décrirons l'architecture proposée ainsi que les outils mobilisés, notamment la collecte des données les modèles de traitement du langage naturel (GPT) et les interfaces d'interaction avec HTML, CSS et JavaScript. Ainsi, ce chapitre constitue une étape essentielle de la conception, puisqu'il traduit les problématiques initiales en exigences fonctionnelles et techniques concrètes, sur lesquelles repose la mise en œuvre effective de la solution.

3.1 Proposition de solution : Robot de recommandation Microsoft Learn

3.1.1 Description générale de la solution proposée

Pour répondre aux problématiques identifiées lors de l'analyse fonctionnelle, la solution proposée consiste en la mise en place d'un cobot de recommandation intelligent dédié à la plateforme Microsoft Learn. Un cobot (ou collaborative robot) est un assistant numérique capable de travailler en collaboration avec l'utilisateur, en automatisant certaines tâches tout en s'adaptant à ses besoins et à ses préférences.

Ce cobot a pour objectif principal de faciliter l'accès aux contenus pertinents et personnalisés pour chaque utilisateur, en tenant compte de son profil, de ses compétences et de ses objectifs professionnels.

La solution repose sur plusieurs composants clés :

- **Collecte des données** : Le cobot se connecte à la plateforme Microsoft Learn via des API pour récupérer en temps réel les informations sur les modules, parcours, certifications et examens disponibles.
- **Analyse du profil utilisateur** : À partir des informations fournies par l'apprenant (niveau de compétence, domaine d'intérêt, parcours déjà suivis), le cobot construit un profil précis qui servira de base aux recommandations.
- **Moteur de recommandation intelligent** : Grâce à des algorithmes d'apprentissage automatique et à des modèles d'IA, le cobot sélectionne les contenus les plus pertinents, en tenant compte de la rapidité, de la pertinence et de la personnalisation.
- **Interface utilisateur interactive** : Les recommandations sont présentées via une interface simple et intuitive, permettant à l'utilisateur de filtrer, explorer et accéder directement aux modules ou parcours proposés.

3.1.2 Rôle du cobot dans l'accompagnement des utilisateurs

Le cobot joue un rôle central dans l'accompagnement des utilisateurs en agissant comme un assistant intelligent et interactif. Sa mission principale est de guider chaque utilisateur dans son parcours de formation sur Microsoft Learn, en proposant :

- **Recommandation personnalisée** : Le cobot analyse le profil, le niveau de compétence et les centres d'intérêt de l'utilisateur pour suggérer les modules et parcours les plus adaptés.
- **Assistance interactive** : Il répond aux questions des utilisateurs en temps réel, fournit des explications supplémentaires et clarifie les concepts complexes.
- **Suivi de progression** : Le cobot suit l'évolution de l'apprentissage, en identifiant les points forts et les lacunes, et propose des contenus ciblés pour améliorer les performances.
- **Support proactif** : Il anticipe les besoins des utilisateurs en proposant des rappels, des exercices pratiques et des contenus complémentaires selon le rythme et le niveau de chacun.

3.1.3 Avantages attendus par rapport aux méthodes classiques

L'introduction du cobot dans le processus de formation offre plusieurs bénéfices par rapport aux méthodes traditionnelles :

- **Personnalisation de l'apprentissage** : Contrairement aux cours standardisés, le cobot adapte le contenu à chaque utilisateur selon ses besoins et son niveau.
- **Gain de temps** : L'utilisateur accède directement aux informations pertinentes sans chercher manuellement dans de longs modules ou documentations.
- **Disponibilité continue** : Le cobot est accessible à tout moment, permettant un apprentissage flexible et autonome, sans contrainte d'horaires ou de disponibilité d'un formateur.
- **Amélioration de l'engagement** : L'interaction dynamique et personnalisée favorise la motivation et l'implication de l'utilisateur dans son parcours de formation.
- **Suivi précis et évolutif** : Le cobot collecte et analyse les données de progression, permettant d'ajuster le parcours en temps réel et d'optimiser les résultats.

3.2 Architecture fonctionnelle et technique

3.2.1 Schéma fonctionnel

Le schéma fonctionnel illustre les interactions principales entre l'utilisateur, le cobot et la plateforme Microsoft Learn :

- **Utilisateur** : interagit avec le cobot via une interface web intégrant un chatbot OpenAI 4.0. L'utilisateur peut poser des questions, demander des recommandations de modules ou suivre sa progression.
- **Cobot** : agit comme intermédiaire intelligent. Il analyse les demandes de l'utilisateur, consulte le fichier Excel structuré à partir de l'API Microsoft Learn (puisque l'API n'est pas directement structurée), traite les données, applique des algorithmes de recommandation et fournit des réponses personnalisées adaptées au profil de l'utilisateur.
- **Plateforme Microsoft Learn** : fournit les contenus pédagogiques, modules de formation, parcours et métadonnées nécessaires à la personnalisation des recommandations.

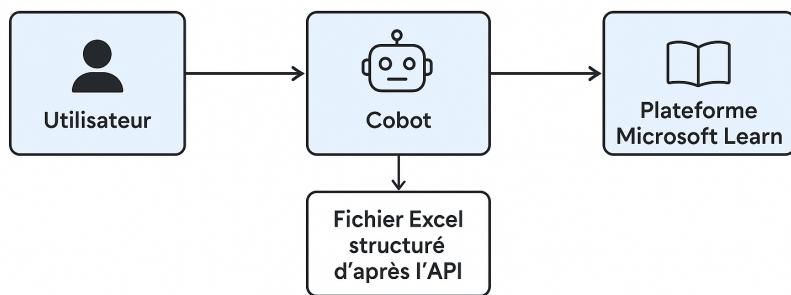


FIGURE 3.1 – Schéma Flux d'interaction

Flux d'interaction :

1. L'utilisateur envoie une requête via l'interface web.
2. Le cobot analyse la demande, consulte le fichier Excel contenant les données Microsoft Learn et applique les règles de recommandation.
3. Le cobot restitue les réponses personnalisées à l'utilisateur, incluant des recommandations de modules et des conseils adaptés.

4. La plateforme Microsoft Learn alimente le fichier Excel régulièrement pour maintenir les contenus à jour.

3.2.2 Architecture technique

L'architecture technique du cobot repose sur un enchaînement de processus allant de la collecte des données jusqu'à leur restitution à l'utilisateur. Elle intègre plusieurs composants interconnectés :

1. Collecte des données

- **Source** : les contenus pédagogiques (modules, parcours, certifications, métadonnées) sont extraits de la plateforme Microsoft Learn.
- **Transformation** : comme l'API Microsoft Learn n'est pas directement structurée, les données sont collectées, nettoyées et organisées dans un fichier Excel structuré afin de faciliter l'exploitation.
- **Mise à jour** : un mécanisme périodique alimente le fichier Excel à partir des nouvelles données disponibles via l'API.

2. Traitement des données

- **Cobot (moteur IA)** : développé en Python, il intègre un modèle NLP basé sur OpenAI (GPT-4o) capable de comprendre les requêtes utilisateurs.
- **Analyse et filtrage** : le cobot interroge le fichier Excel, applique des algorithmes de recherche et de recommandation (par mots-clés, correspondance de profils, scoring).
- **Adaptation personnalisée** : le traitement prend en compte le profil, le niveau et l'historique de l'utilisateur pour affiner les recommandations.

3. Restitution des données

- **Interface utilisateur** : une interface web (HTML/CSS/JS) permet à l'utilisateur d'interagir avec le cobot.
- **Réponses intelligentes** : le cobot formule des recommandations sous forme de messages interactifs (modules adaptés, parcours de formation, rappels de progression).
- **Visualisation** : affichage graphique (progress bars, graphiques, tableaux) pour un suivi clair et ergonomique.

3.2.3 Gestion de la sécurité et de la confidentialité

La mise en place du cobot nécessite une gestion rigoureuse de la sécurité et de la confidentialité afin de protéger les données des utilisateurs ainsi que l'accès aux services externes (API). Les mesures adoptées sont les suivantes :

1. Sécurisation des clés et accès API

- Les clés API (Microsoft Learn, OpenAI, etc.) ne sont jamais stockées en clair dans le code source.
- Elles sont sauvegardées sous forme de variables d'environnement afin d'empêcher toute exposition accidentelle lors du partage du projet (ex. GitHub).
- Les appels API utilisent le protocole HTTPS pour assurer la confidentialité et l'intégrité des échanges.

2. Protection des données des utilisateurs

- Les informations personnelles (profil, historique de formation) sont anonymisées ou pseudonymisées.
- Les données sensibles ne sont stockées que temporairement pour le traitement et sont supprimées après usage.
- En cas de stockage, une base de données sécurisée est utilisée avec un contrôle d'accès restreint.

3. Confidentialité et conformité

- Respect des bonnes pratiques de protection des données (RGPD/GDPR) : consentement explicite de l'utilisateur, transparence sur l'usage des données.
- Limitation de la collecte au strict nécessaire (principe de minimisation).

4. Sécurisation de l'application

- Authentification des utilisateurs pour accéder aux services personnalisés.

- Vérification et filtrage des entrées pour éviter les attaques de type injection (SQL, scripts).
- Sauvegardes régulières des fichiers Excel structurés et bases de données pour assurer la disponibilité.

3.3 Technologies utilisées

1. Python

Python est le langage de programmation principal utilisé dans ce projet.

- **Historique :** Créé en 1991 par Guido van Rossum, Python est rapidement devenu l'un des langages les plus populaires au monde. Sa philosophie repose sur la simplicité, la lisibilité et la rapidité de développement.
- **Pourquoi Python ici ?**
 - o Il dispose d'une vaste bibliothèque (Pandas, NumPy, Requests, etc.) adaptée au traitement des données.
 - o Il est largement utilisé pour l'intelligence artificielle et le traitement du langage naturel.
 - o Sa facilité d'intégration avec des APIs et des systèmes externes en fait un choix naturel pour la collecte et la manipulation de données.
- **Rôle dans le projet :** Python est utilisé pour se connecter à l'API Microsoft Learn, extraire les données brutes, les nettoyer (suppression de doublons, réorganisation des colonnes, normalisation des formats) et les exporter vers un fichier structuré (Excel). Il est également le point d'intégration entre les données et le modèle d'IA.



FIGURE 3.2 – Python logo

2. API Microsoft Learn

- **Définition :** Une API (Application Programming Interface) est une interface permettant à des logiciels de communiquer entre eux.

- **Historique de Microsoft Learn :** Microsoft a lancé cette plateforme en 2018 pour centraliser ses contenus de formation (modules, parcours, certifications) autour de ses technologies (Azure, Power Platform, IA, etc.). Afin de rendre ces ressources accessibles de manière automatisée, Microsoft a mis à disposition une API publique.
- Pourquoi utiliser cette API ?
 - o Accéder en temps réel aux formations disponibles.
 - o Éviter le téléchargement manuel de fichiers volumineux.
 - o Automatiser la mise à jour des contenus du cobot.

- **Rôle dans le projet :** L'API est la source officielle de toutes les données (modules, parcours, certifications). C'est par elle que Python collecte les informations avant de les nettoyer et de les transmettre au cobot.

3. Excel structuré

- **Historique :** Microsoft Excel existe depuis 1985. Initialement conçu comme un tableau pour les calculs financiers, il est devenu au fil des décennies un outil incontournable pour l'analyse et la visualisation des données.

- **Pourquoi Excel ?**

- o Facile à manipuler par les non-développeurs.
 - o Compatible avec de nombreux outils (Power BI, Python, etc.).
 - o Permet un stockage structuré avant intégration dans des systèmes plus complexes (base de données, tableau de bord, etc.).
- **Rôle dans le projet :** Après extraction et nettoyage par Python, les données sont exportées dans un fichier Excel organisé. Ce fichier sert de base intermédiaire pour valider les informations et faciliter leur exploitation future.



FIGURE 3.3 – Excel logo

4. OpenAI GPT-4o

- **Historique :** OpenAI, fondée en 2015, a développé plusieurs modèles d'IA. Le modèle GPT (Generative Pre-trained Transformer) est apparu en 2018. GPT-4o, lancé en 2024, est une version optimisée, multimodale et plus rapide.

- **Pourquoi GPT-4o dans ce projet ?**
 - Compréhension avancée du langage naturel (NLP).
 - Capacité à générer des réponses personnalisées, adaptées au profil et à la demande de l'utilisateur.
 - Intégration fluide avec Python et une interface web.
- **Rôle dans le projet :** GPT-4o constitue le cœur intelligent du cobot. Il analyse les requêtes des utilisateurs (ex. "Je cherche une formation en Data Science"), puis va chercher dans les données préparées les formations les plus pertinentes.



FIGURE 3.4 – OpenAI logo

5. Interface Web (HTML / CSS / JavaScript)

- **Historique :**
 - **HTML (HyperText Markup Language)** : créé en 1993, il est la base de la structure des pages web.
 - **CSS (Cascading Style Sheets)** : introduit en 1996, il permet de styliser et mettre en forme les pages HTML.
 - **JavaScript** : développé en 1995, il apporte l'interactivité aux sites web.
- **Pourquoi ces technologies ?**
 - Standards universels du web, compatibles avec tous les navigateurs.
 - Faciles à déployer et légères, ce qui correspond à un cobot accessible en ligne.
 - Permettent de créer une interface simple et intuitive, utilisable sans compétences techniques.
- **Rôle dans le projet :** L'interface web sert de point d'accès entre l'utilisateur et le cobot. Elle offre un champ de saisie (pour poser des questions) et une zone de réponse (où GPT-4o affiche les recommandations).



FIGURE 3.5 – js/css/html logo

Conclusion

Dans ce chapitre, nous avons présenté le cobot de recommandation Microsoft Learn comme solution aux limites identifiées lors de l'analyse fonctionnelle. Ce cobot agit comme un assistant intelligent, offrant des recommandations de formations personnalisées, rapides et pertinentes. Nous avons détaillé sa conception, son rôle dans l'accompagnement des utilisateurs, ainsi que son architecture fonctionnelle et technique, basée sur l'API Microsoft Learn, Python pour le traitement des données, et le modèle d'IA GPT-4o pour générer des recommandations adaptées. L'interface web constitue le point d'accès principal, garantissant une interaction simple et intuitive. Ce chapitre établit ainsi les fondations de la solution technique, tandis que le chapitre suivant portera sur la structuration et l'enrichissement du fichier Excel, cœur du système de recommandation.

Chapitre 4

Structuration et enrichissement du fichier Excel

Introduction

Ce chapitre présente le processus de structuration et d'enrichissement du fichier Excel qui constitue la base de fonctionnement du cobot de recommandation. L'objectif est de transformer les données brutes issues de l'API Microsoft Learn en un dataset propre, organisé et enrichi, afin de garantir des recommandations précises et pertinentes. Pour ce faire, plusieurs étapes ont été mises en place : la collecte des données, leur nettoyage et leur organisation, l'intégration d'un système de recommandation basé sur la popularité, ainsi qu'un pipeline avancé permettant d'améliorer la qualité des informations grâce au NLP, au scraping et à la détection automatique des technologies abordées.

4.1 Collecte via l'API Microsoft Learn

4.1.1 Connexion à l'API officielle

La première étape de la structuration du dataset repose sur l'exploitation du Microsoft Learn Catalog API, un service REST public accessible sans authentification. Cette API permet de récupérer l'ensemble des métadonnées relatives aux ressources de formation disponibles sur Microsoft Learn. Parmi les éléments accessibles on trouve les modules, unités (units), parcours d'apprentissage (learning paths), Applied Skills, certifications, examens.

Détails techniques :

- **Point de terminaison principal** : Un simple appel GET à l'URL <https://learn.microsoft.com/api/catalog/> permet d'obtenir le catalogue complet, une réponse JSON d'environ 13 Mo à date de novembre 2022.
- **Paramètres facultatifs disponibles pour filtrer ou affiner les données rentrées :**
 - **Locale** : pour choisir une langue (par défaut en-us).
 - **Type** : pour extraire des catégories spécifiques telles que modules, learningPaths, certifications, etc...
 - **Autres critères** comme level, subject, popularity, last_modified permettent un filtrage plus fin lors de l'appel API.

Avantages et bonnes pratiques :

- **Accessibilité immédiate** : L'API est publique, gratuite, ne requiert pas d'authentification et est conforme aux conditions d'utilisation des API Microsoft.
- **Mises à jour fréquentes** : Les données sont alimentées quotidiennement, avec au moins une actualisation par jour, voire deux fois en période de forts changements.
- **UID parfaits pour le suivi de versions** : Chaque objet dispose d'un identifiant unique (UID). En cas de modifications, un nouvel objet est généré, même si le contenu semble similaire, ce qui facilite le suivi des versions.

4.1.2 Téléchargement des données brutes au format JSON

Une fois la connexion à l'API officielle établie, la deuxième étape consiste à télécharger les données brutes. L'API Microsoft Learn renvoie les informations au format JSON (JavaScript Object Notation), un format textuel léger, largement utilisé pour l'échange et le stockage de données.

Le choix de JSON présente plusieurs avantages :

- **Lisibilité** : les données sont structurées sous forme de paires clé-valeur, ce qui facilite leur compréhension.
- **Compatibilité** : JSON est nativement supporté par Python et de nombreux langages, ce qui simplifie son intégration.
- **Flexibilité** : les objets JSON peuvent contenir des structures imbriquées (par exemple, un module avec ses unités internes), ce qui reflète fidèlement l'organisation des contenus Microsoft Learn.

En pratique, une requête GET simple vers l'URL <https://learn.microsoft.com/api/catalog> retourne un fichier JSON de plusieurs mégaoctets, contenant l'ensemble des ressources : modules, learning paths, certifications, examens et Applied Skills. Ce fichier constitue la matière brute qui sera ensuite nettoyée, filtrée et transformée dans les étapes suivantes afin de créer un dataset exploitable par le cobot de recommandation.

4.2 Nettoyage et enrichissement de base

Après la collecte des données brutes depuis l'API Microsoft Learn, une étape essentielle consiste à nettoyer et enrichir le dataset afin de le rendre exploitable pour le cobot de recommandation.

4.2.1 Suppression des colonnes inutiles

Le fichier JSON renvoyé par l'API contient un grand nombre de champs qui ne sont pas directement utiles pour l'analyse ou la recommandation. Par exemple :

- **image_url** : lien vers l'illustration visuelle du cours (inutile pour le filtrage).

- **first_unit_url** : lien interne vers la première unité d'un module.
- **last_modified, slug, typeahead_displayname, etc** : Métadonnées techniques utilisées par Microsoft Learn mais non nécessaires pour l'utilisateur final.
- **Colonnes liées au tracking interne Microsoft** (identifiants, chemins relatifs, etc.).

Ces colonnes, bien que pertinentes pour l'écosystème Microsoft, n'apportent pas de valeur ajoutée dans le cadre du projet et alourdissent inutilement le dataset.

4.2.2 Colonnes conservées

Après nettoyage, seules les colonnes essentielles pour la recommandation et la compréhension des formations sont maintenues :

- **title** : titre du cours, module, parcours ou certification.
- **summary** : résumé ou description du contenu.
- **subjects** : thématiques ou domaines abordés (IA, Cloud, Cybersécurité, etc.).
- **roles** : rôles professionnels visés (Data Scientist, Developer, Security Engineer, etc.).
- **levels** : niveau de difficulté (Beginner, Intermediate, Advanced).
- **duration_in_minutes** : durée estimée de la formation.
- **type** : nature de la ressource (module, course, exam, certification, learning path).
- **url** : lien direct vers la ressource sur Microsoft Learn.
- **uid** : identifiant unique pour chaque formation.
- **popularity (ajoutée plus tard)** : indicateur calculé pour prioriser les formations les plus suivies.

4.2.3 Suppression des contenus retirés (retired)

- Certains cours ou certifications sont marqués comme retired (retirés de la plate-forme).
- Pour éviter d'avoir des données obsolètes dans le dataset, le script vérifie chaque URL :
 - o Une requête HTTP est envoyée via la fonction `make_request`.
 - o Si la page contient des mentions comme "retired" ou "has been retired", l'entrée correspondante est automatiquement supprimée du DataFrame.

Résultat : seules les formations actives restent dans le dataset.

4.2.4 Scraping des informations manquantes

Après l'élimination des entrées classées comme retired, dans un souci d'assurer la cohérence et la fiabilité du corpus de données, une étape d'enrichissement a été mise en œuvre afin de combler les champs restés incomplets. Cette opération repose sur l'utilisation de la bibliothèque **BeautifulSoup**, reconnue pour ses capacités avancées de web scraping

et d'extraction d'informations structurées à partir de documents HTML. Le script développé a pour objectif d'identifier et d'extraire automatiquement des attributs pertinents tels que **la durée en minutes (duration_in_minutes)**, **le nombre d'enfants (number_of_children)** ainsi que **le résumé (summary)** associé à chaque enregistrement. L'intégration de ces informations manquantes permet non seulement d'accroître la complétude et la richesse sémantique du dataset, mais également d'améliorer la robustesse des traitements analytiques ultérieurs.

En outre, l'automatisation de ce processus garantit une homogénéité dans la collecte, réduit le risque d'erreurs humaines et optimise le temps consacré aux tâches de prétraitement des données, ce qui constitue un atout considérable dans un contexte de manipulation de données à grande échelle.

4.2.5 Création de la colonne certified

L'objectif est d'indiquer si un contenu Microsoft Learn correspond à une formation certifiante ou non. Cette information permet de distinguer les cours ou modules « officiels » qui conduisent à une certification des contenus plus généraux ou exploratoires.

- La colonne certified est calculée à partir de la colonne type.
- Les types considérés comme certifiants sont :
 - o **cert (certification)**
 - o **exam (examen)**
 - o **Microsoft Applied Skills (formations spécifiques Microsoft)**

Résultat :

- o Si le type du contenu est dans certified_types → certified = "yes"
- o Sinon → certified = "no"

4.3 Remplissage automatique des sujets manquants

Lors de l'extraction des données depuis l'API Microsoft Learn, nous avons constaté que de nombreux enregistrements ne contenaient pas d'information dans la colonne **subjects**, qui correspond aux thématiques ou domaines principaux d'un cours ou d'un module (exemple : Cloud, Intelligence Artificielle, Développement Web).

Or, cette information est essentielle pour construire un système de recommandation pertinent. En effet, si une formation n'est associée à aucun sujet, elle risque de ne pas être sélectionnée lors du filtrage, ce qui réduit la qualité globale du dataset et appauvrit les suggestions proposées par le cobot.

Afin de résoudre ce problème, nous avons mis en place une approche basée sur le traitement automatique du **langage naturel (NLP)**. Plus précisément, nous avons utilisé une combinaison de deux techniques complémentaires :

4.3.1 TF-IDF (Term Frequency – Inverse Document Frequency)

C'est une méthode statistique largement utilisée en text mining. Elle permet de mesurer l'importance d'un mot dans un document relativement à l'ensemble du corpus.

Concrètement, les mots très fréquents dans un texte mais rares dans le reste du corpus reçoivent un poids élevé. Cela permet d'identifier les termes les plus représentatifs d'un cours donné.

Dans notre cas, les colonnes **title (titre)** et **summary (résumé)** de chaque cours ont été combinées et transformées en vecteurs numériques grâce au TF-IDF.

4.3.2 Régression Logistique pour la classification multi-étiquettes

Un même cours peut appartenir à plusieurs thématiques (par exemple Data Science et Machine Learning). Pour gérer ce cas, nous avons utilisé une approche dite **multi-label classification**. Concrètement, nous avons entraîné une série de classificateurs (un par sujet possible) afin de prédire, pour chaque formation sans sujet, si elle appartient ou non à chacune des catégories connues. Pour encoder ces sujets multiples, nous avons utilisé la classe **MultiLabelBinarizer** de la bibliothèque Scikit-learn, qui transforme chaque ensemble de sujets en vecteurs binaires.

4.4 Calcul de la popularité des contenus

Après avoir complété et enrichi le dataset, une étape essentielle consiste à attribuer un **score de popularité** à chaque contenu (cours, certification, examen, etc.). Ce score permet de mesurer l'attractivité d'un module par rapport aux autres et servira plus tard à la recommandation personnalisée.

4.4.1 Calcul de la popularité des sujets

La colonne **subjects** indique les thématiques principales de chaque formation (ex. Cloud, Data, AI).

- Pour estimer l'importance de chaque sujet, on calcule sa fréquence d'apparition dans l'ensemble du dataset.
- Exemple : si Azure apparaît dans 300 cours et Python dans 50, alors Azure est considéré comme plus populaire.
- Chaque ligne reçoit un score **subject_popularity** basé sur la somme des fréquences de ses sujets.

4.4.2 Normalisation des variables

Comme les variables utilisées pour le score global (durée, nombre d'unités, popularité des sujets) n'ont pas la même échelle, elles sont normalisées avec la méthode Min-Max Scaling de scikit-learn.

- Cela ramène toutes les valeurs dans un intervalle commun [0, 1].

4.4.3 Pondération et score final

Pour obtenir un score de popularité global, on combine les trois critères avec des poids définis empiriquement :

- 20 % → Durée (duration_in_minutes)
- 40 % → Nombre d'unités (number_of_children)
- 40 % → Popularité des sujets (subject_popularity)

La formule est donc :

$$\text{popularity} = 0.2 \times \text{duration} + 0.4 \times \text{children} + 0.4 \times \text{subjects}$$

4.5 Identification et ajout de la dimension Technologie

L'une des limites du dataset initial de Microsoft Learn est l'absence d'une information explicite sur **les technologies couvertes par chaque contenu**. Or, cette dimension est essentielle pour améliorer la pertinence des recommandations et faciliter le filtrage par domaine technique. L'objectif de cette étape est donc **d'ajouter une colonne Technologie** à partir des informations textuelles disponibles dans le dataset.

4.5.1 Constitution d'un dictionnaire de technologies

Une liste de technologies pertinentes et fréquemment utilisées dans l'écosystème Microsoft et au-delà a été définie manuellement. Elle couvre différents domaines :

- **Langages de programmation** : Python, Java, C#, JavaScript, C++, PHP, etc.
- **Bases de données** : SQL, MongoDB, PostgreSQL, MySQL, etc.
- **Frameworks et bibliothèques** : React, Angular, Django, TensorFlow, PyTorch, etc.
- **Outils de data et BI** : Power BI, Tableau, Excel, Pandas, NumPy, etc.
- **Cloud et DevOps** : Azure, AWS, Docker, Kubernetes, Git, Jenkins, etc.
- **Autres domaines transverses** : Intelligence Artificielle, Data Science, Blockchain, Cybersécurité, etc.

4.5.2 Extraction des technologies dans les textes

Pour chaque ligne du dataset, les colonnes textuelles suivantes sont fusionnées :

- **title (titre de la formation)**,
- **summary (résumé)**,
- **subjects (sujets)**.

Un algorithme basé sur des expressions régulières (regex) recherche ensuite la présence de chaque technologie de la liste dans ce texte combiné.

Exemple :

- **Titre** : "Introduction to Machine Learning with Python"
- **Détection** → Python, Machine Learning

. Importance pour le pipeline

L'ajout de cette colonne apporte une plus-value stratégique :

- Elle permet de filtrer les contenus selon les technologies recherchées par l'utilisateur (ex. "cours sur Azure" ou "formations Python").
- Elle contribue à une meilleure personnalisation des recommandations du cobot.
- Elle enrichit le dataset avec une dimension métier qui n'existe pas initialement.

4.6 Export final du dataset

Après avoir appliqué toutes les étapes du pipeline (restructuration, enrichissement web, remplissage des sujets manquants, calcul de popularité et ajout de la colonne Technologie), le DataFrame final est complet, nettoyé et enrichi.

- Il contient désormais toutes les informations essentielles :

o title, summary, levels, roles, products, uid, type, duration_in_minutes, number_of_children, subjects, Technologie, certified, popularity, etc.

- Les valeurs manquantes ont été comblées, les contenus retirés supprimés, et des colonnes calculées (popularity, Technologie) ajoutées.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
1	summary	levels	roles	products	uid	type	title	icon_in_mi	popularity	url	firstUnitUrl	units	per_of_chi	subjects	certified	technologie			
2	This mode	'intermed	'administr	'windows learn.vwl.module		Deploy dev	48	0.077049		https://lea https://lea	'learn.vwl	8	['data', 'Big no	Azure					
3	Learn to b	'beginner	'develop	'dotnet', 'learn.langt module		Add decisiv	56	0.086464		https://lea https://lea	'learn.lan	7	['app-deve no	Azure					
4	Implement	'intermed	'develop	'azure-sdl learn.vwl.module		Implement	30	0.049439		https://lea https://lea	'learn.vwl	6	['identity-n	Azure					
5	Learn how	'beginner	'develop	'dotnet', 'learn.vwl.module		Get starte	58	0.090615		https://lea https://lea	'learn.vwl	8	['dates', 'd no	Azure					
6	You learn	'intermed	'administr	'azure-ex learn.vwl.module		Design arc	57	0.093315		https://lea https://lea	'learn.vwl	11	['data-eng no	Azure					
7	Learn abou	'beginner	'data-sci	'azure-de learn.vwl.module		Introducti	33	0.053568		https://lea https://lea	'learn.vwl	6	['devops' no	DevOps, Machine Learning					
8	How comp	'beginner	'develop	'azure'		learn.work module	37	0.06461		https://lea https://lea	'learn.wor	10	['guidance no	AI					
9	As the am	'intermed	'ai-engine	'ai-service learn.vwl.module		Use AI resq	70	0.105792		https://lea https://lea	'learn.vwl	7	['cloud-coi no	AI, Azure					
10	Monitor W	'intermed	'administr	'azure', 'a learn.vwl.module		Monitor W	46	0.074114		https://lea https://lea	'learn.vwl	8	['virtual-m no	Windows Server					
11	Use mode	'intermed	'ai-engine	'azure-op learn.vwl.module		Generate i	33	0.054963		https://lea https://lea	'learn.vwl	7	['artificial' no	AI, Azure					
12	Manage A	'intermed	'devops-e	'azure', 'learn.vwl.module		Manage A	66	0.105775		https://lea https://lea	'learn.vwl	11	['cloud-coi no	Azure, Kubernetes					
13	Describe c	'beginner	'business-	'azure', 'learn.vwl.module		Describe c	24	0.04252		https://lea https://lea	'learn.vwl	7	['blockcha no	Blockchain					
14	We will lea	'intermed	'develop	'azure-co', 'learn.vwl.module		Monitor re	51	0.082444		https://lea https://lea	'learn.vwl	9	['cloud-co no	Azure, NoSQL					
15	Learn how	'beginner	'data-sci	'azure', 'a learn.tensio module		Introducti	48	0.072704		https://lea https://lea	'learn.ten	5	['classifica no	TensorFlow					
16	Learn how	'beginner	'develop	'azure', 'a learn.desig module		Design a Pi	151	0.219498		https://lea https://lea	'learn.des	9	['data-mov no	Azure, SQL					
17	In this mot	'beginner	'administr	'intunet', 'learn.vwl.module		Discover N	30	0.054034		https://lea https://lea	'learn.vwl	9	['data', 'es no	Azure					
18	Learn how	'intermed	'develop	'azure', 'a learn.vwl.module		Develop A	38	0.061877		https://lea https://lea	'learn.vwl	7	['cloud-co no	Azure					
19	Learn how	'beginner	'develop	'azure', 'learn.celci module		Develop p	99	0.148395		https://lea https://lea	'learn.celi	9	['design', 'i no	Azure					
20	Implement	'intermed	'administr	'azure', 'a learn.vwl.module		Implement	64	0.101609		https://lea https://lea	'learn.vwl	10	['cloud-sec no	Windows Server					
21	Share kno	'advanced	'administr	'azure', 'a learn.vwl.module		Share kno	57	0.087804		https://lea https://lea	'learn.vwl	7	['communi no	Azure					
22	Learn how	'beginner	'ai-engine	'azure', 'learn.vwl.module		Get starte	31	0.052245		https://lea https://lea	'learn.vwl	7	['artificial' no	AI, Azure					
23	Monitor ai	'intermed	'develop	'azure-mc learn.vwl.module		Monitor aj	44	0.072842		https://lea https://lea	'learn.vwl	9	['cloud-co no	Azure					
24	Learn how	'beginner	'data-ana	'fabric', 'learn.vwl.module		Train and t	54	0.083741		https://lea https://lea	'learn.vwl	7	['data-mat no	Machine Learning					
25	Explore hc	'beginner	'solution-	'azure', 'a learn.aks- module		Applicatio	29	0.046775		https://lea https://lea	'learn.aks	5	['cloud-co no	Azure					
26	Learn mult	'intermed	'administr	'azure-sql learn.vwl.module		Assess SQL	50	0.081221		https://lea https://lea	'learn.vwl	9	['Assess', 'l no	Azure, SQL					
27	In this mci	'beginner	'k-12-edu	'm365-ed learn.dpu.module		Introducti	24	0.052665		https://lea https://lea	'learn.dpu	14	['K-12', 'm no	Azure					

FIGURE 4.1 – Extrait de Dataset

Le dataset final est sauvegardé sous le fichier :

Microsoft_learn_data_final.xlsx

Ce fichier constitue la base prête à l'emploi pour toutes les analyses ultérieures ou pour l'intégration dans le cobot de recommandation Microsoft Learn.

Conclusion

Ce chapitre a présenté le pipeline de traitement du dataset Microsoft Learn, incluant le nettoyage des données, la suppression des contenus retirés, l'enrichissement web, le remplissage des sujets manquants par NLP, le calcul de la popularité et l'ajout de la colonne Technologie. Le dataset final, complet et structuré, a été exporté dans Microsoft_learn_data_final.xlsx, prêt à être utilisé pour les analyses et recommandations du cobot.

Chapitre 5

Intégration d'OpenAI et Développement de l'Interface du Robot Microsoft Learn

Introduction

Dans ce chapitre, nous présentons la structure technique du projet Cobot Microsoft Learn, en détaillant l'architecture globale, les technologies utilisées, et le rôle de chaque composant. L'objectif est de montrer comment les différentes parties du système ,depuis la collecte des données jusqu'à l'affichage des résultats pour l'utilisateur ,interagissent de manière cohérente et efficace.

5.1 Architecture générale de l'application

5.1.1 Fonctionnalités principales et ergonomie

Le projet Cobot Microsoft Learn propose plusieurs fonctionnalités clés visant à accompagner les utilisateurs dans l'exploration des parcours de formation :

- **Chatbot intelligent** : une interface interactive permettant de poser des questions en langage naturel et de recevoir des recommandations personnalisées de cours et certifications.
- **Dashboard analytique** : un espace regroupant des indicateurs clés (KPIs), des graphiques interactifs (via Plotly) et un aperçu des données filtrées, facilitant l'analyse et la compréhension des formations disponibles.
- **Filtres avancés** : possibilité de sélectionner les formations par niveau, type, rôle ou produit pour obtenir des résultats plus précis.
- **Pagination et affichage dynamique** : les résultats sont organisés par pages pour garantir fluidité et clarté dans l'affichage des recommandations.
- **Intégration responsive** : l'interface a été conçue pour s'adapter aux différents supports (ordinateurs, tablettes, mobiles) tout en gardant une navigation simple et intuitive.

Sur le plan ergonomique, l'accent a été mis sur :

- une navigation claire via des menus et sections bien structurés (Accueil, Chatbot, Dashboard, À propos, Contact),
- un design sobre et professionnel, inspiré de l'identité visuelle de Microsoft,
- une expérience utilisateur fluide, grâce à l'automatisation des interactions et à une mise en page optimisée.

5.1.2 Présentation de la structure globale du projet

Pages principales :

- **index.html** : page d'accueil présentant le projet, ses fonctionnalités et les informations sur l'équipe et l'entreprise.
- **chatbot.html** : interface du Cobot pour poser des questions et obtenir des recommandations personnalisées, avec pagination des résultats.
- **dashboard.html** : tableau de bord analytique affichant des KPIs, des graphiques interactifs et un aperçu des données filtrées.

Fonctionnalités principales :

- Navigation fluide entre les sections grâce à des onglets.
- Chatbot intelligent fournissant des recommandations détaillées (titre, type, niveau, durée, popularité, résumé, lien).
- Dashboard interactif avec filtres, KPIs, graphiques Plotly et aperçu des données.

Technologies utilisées :

- **Front-end** : HTML, CSS, JavaScript
- **Visualisation** : Plotly.js pour les graphiques
- **Backend / API** : endpoints pour récupérer les filtres, recommandations et réponses du chatbot.

5.1.3 Schéma de l'architecture technique

L'architecture technique du projet Cobot Microsoft Learn est organisée de manière modulaire, avec trois niveaux principaux :

1. Front-End (Interface Utilisateur)

- o **Pages HTML** : index.html, chatbot.html, dashboard.html
- o **CSS** : style et mise en page
- o **JavaScript (script.js)** : interactions, gestion des filtres, affichage dynamique des recommandations et des graphiques
- o **Plotly.js** : visualisation interactive des KPIs et des données

2. Back-End / API

- o Endpoints pour :
Fournir les filtres (/api/filters)
Traiter les requêtes du chatbot (/api/chat)
Fournir les recommandations et données analytiques (/api/recommendations)
- o Logique de filtrage et pagination des résultats

3. Base de données / Source de données

- o Contient les informations des formations (titre, type, niveau, popularité, durée, résumé, etc.)
- o Sert à générer les recommandations et à alimenter le dashboard analytique

Flux de données :

- L'utilisateur interagit avec l'interface → requête envoyée à l'API → traitement dans le back-end → réponse JSON → affichage dynamique côté front-end.

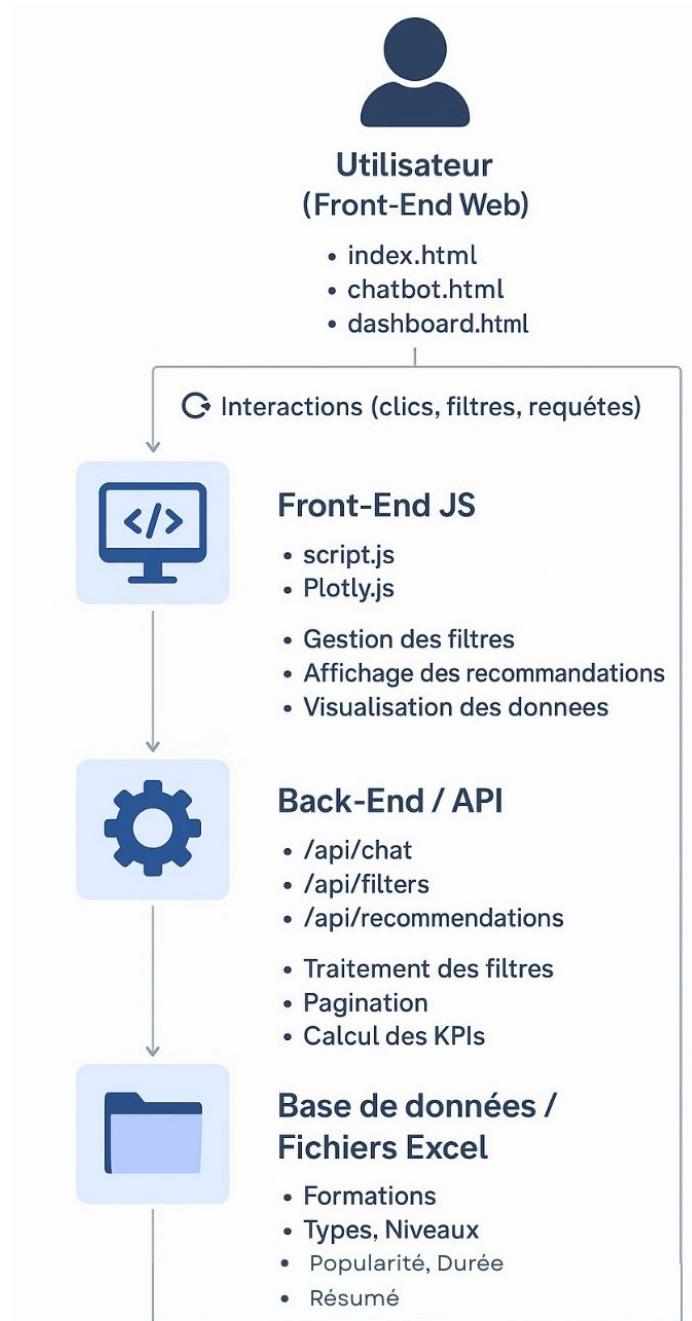


FIGURE 5.1 – Architecture technique

- Les utilisateurs interagissent avec l'interface.
- Le JavaScript gère la logique côté client et affiche les données.
- Les API du back-end traitent les requêtes et appliquent les filtres, calculent les KPIs et la pagination.
- Les données sont stockées dans des fichiers Excel ou une base de données et alimentent à la fois les recommandations et le dashboard analytique.

5.2 Intégration d'OpenAI

5.2.1 Mise en place et configuration de l'API

Dans le cadre de ce projet, l'API joue un rôle central pour relier le front-end aux données des formations Microsoft Learn et permettre l'affichage dynamique des recommandations et des indicateurs sur le dashboard.

1. Choix technologique

- **Langage / Framework :** Python avec Flask pour sa légèreté et sa simplicité dans la création de routes API.



FIGURE 5.2 – flask web development framework

- **Endpoints principaux :**

- **/api/filters** : Fournit les types et niveaux disponibles pour les filtres du dashboard.
 - **/api/recommendations** : Retourne les formations filtrées, les statistiques (KPIs) et les données pour les graphiques.
 - **/api/chat** : Gère les requêtes du chatbot, la pagination des résultats et les réponses interactives.

2. Configuration

- L'API est configurée pour recevoir des requêtes JSON et répondre en JSON, ce qui permet une intégration facile avec le front-end en JavaScript.
- Cors est activé pour autoriser les requêtes provenant du front-end, même si le serveur est séparé

- Les endpoints utilisent des méthodes POST pour récupérer les filtres, les requêtes du chatbot et les paramètres sélectionnés dans le dashboard.

3. Traitement des données

- Les données sources proviennent d'un fichier Excel contenant les formations, leurs niveaux, types, durées, popularité, résumé et statut de certification.
- L'API lit ces données, applique les filtres sélectionnés par l'utilisateur et calcule :

- **Les KPIs** : nombre total de formations, durée totale, popularité moyenne, pourcentage de certifications.

- **Les données pour les graphiques** : statistiques de répartition par type, niveau, durée, popularité, etc.

- **Les résultats paginés** pour le chatbot, avec 10 résultats par page.

4. Exemple de fonctionnement

1. L'utilisateur sélectionne des filtres sur le dashboard.
2. Le front-end envoie une requête POST à /api/recommendations avec les filtres.
3. L'API traite les données, calcule les indicateurs et retourne :

```
{  
  "kpis" : ...,  
  "charts" : ...,  
  "data_preview" : [...],  
  "recommendations" : [...]  
}
```

4. Le front-end affiche dynamiquement les résultats dans le tableau, les graphiques et les KPI.

5. Sécurité et bonnes pratiques

- Les entrées utilisateur sont validées côté serveur pour éviter les erreurs et injections.
- Les fichiers Excel sont chargés et manipulés en mémoire, limitant les accès directs et améliorant les performances.

Flux de traitement des données (Front-End → API → Excel → JSON → Dashboard)

1. **Front-End** : L'utilisateur interagit avec le dashboard ou le chatbot et envoie une requête via JavaScript.
2. **API (Flask)** : Reçoit la requête, applique les filtres et traite les données.
3. **Fichier Excel** : Contient toutes les données de formations Microsoft Learn. L'API le lit pour récupérer les informations nécessaires.
4. **Traitement** : L'API calcule les KPI, prépare les graphiques et sélectionne les recommandations correspondant aux filtres ou à la requête du chatbot.
5. **Réponse JSON** : Les données calculées sont envoyées au front-end sous forme de

JSON.

6. Front-End : Affiche dynamiquement les résultats dans le dashboard (KPIs, graphiques, aperçu des données) ou dans l'interface du chatbot.

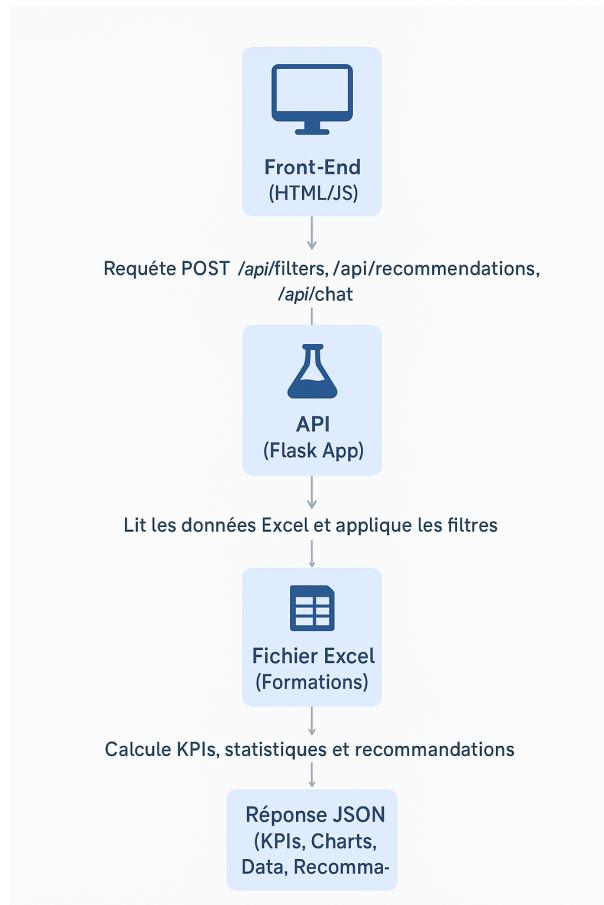


FIGURE 5.3 – Schéma de Flux de traitement des données

5.2.2 Gestion des clés d'accès et sécurité

Accès à l'API

- L'API du projet est construite avec Flask, exposant des endpoints tels que /api/filters, /api/recommendations et /api/chat.
 - Pour sécuriser ces endpoints, seules les requêtes provenant du front-end de l'application sont autorisées à accéder aux données.
 - Des contrôles côté serveur vérifient les entrées afin de prévenir les injections malveillantes ou requêtes non conformes.

Gestion des données sensibles

- Les informations sensibles comme les adresses e-mail ou les données internes de l'entreprise ne sont jamais exposées dans le front-end.
 - Les fichiers Excel contenant les formations sont stockés côté serveur et ne sont accessibles que via l'API, qui renvoie uniquement les informations nécessaires pour le front-end.

Sécurisation des échanges

- Les requêtes entre le front-end et l'API utilisent le protocole HTTPS pour chiffrer les échanges et protéger les données contre les interceptions.
- Les réponses JSON sont validées et nettoyées avant d'être envoyées au client afin de prévenir les attaques XSS ou l'injection de scripts.

Bonnes pratiques côté front-end

- Validation des entrées utilisateur pour le chatbot et les filtres du dashboard afin de limiter les erreurs et potentielles attaques.
- Mise en place d'une pagination et limitation du nombre de résultats par requête pour éviter la surcharge du serveur.

5.3 Implémentation technique

5.3.1 Conception en HTML, CSS et JavaScript

La conception du Cobot Microsoft Learn repose sur une architecture front-end développée en **HTML, CSS et JavaScript**.

- **HTML** est utilisé pour définir la structure des différentes pages :
 - o index.html : page d'accueil avec présentation des fonctionnalités, accès au chatbot et au dashboard.
 - o chatbot.html : interface de conversation interactive avec le cobot.
 - o dashboard.html : espace d'analyse des données avec filtres, KPIs et graphiques.
- **CSS** permet de gérer la mise en forme et l'esthétique de l'interface :
 - o Uniformisation de la charte graphique avec des couleurs sobres et professionnelles.
 - o Mise en place de grilles et de styles responsives afin d'assurer une bonne expérience utilisateur sur différents écrans.
 - o Utilisation d'icônes et d'animations légères pour rendre l'application plus intuitive.
- **JavaScript** assure la dynamique et l'interactivité du projet :
 - o Gestion des onglets de navigation et transitions entre sections.
 - o Intégration des filtres pour le dashboard et affichage en temps réel des KPIs, graphiques (via Plotly) et tableaux.
 - o Fonctionnalité du chatbot permettant l'envoi de requêtes à l'API et l'affichage des recommandations.
 - o Mise en place de la pagination pour gérer efficacement le volume de résultats affichés.

5.3.2 Workflow de communication entre les composants

Le projet suit un enchaînement logique de communication entre ses différentes parties (frontend, backend et APIs). Le workflow peut être résumé ainsi :

Interaction utilisateur (Frontend) :

- L'utilisateur accède à l'application via l'une des interfaces (index.html, chatbot.html, dashboard.html).
- Il saisit une question dans le chatbot ou applique des filtres dans le dashboard.

Traitements par le JavaScript (Côté client) :

- Le fichier script.js intercepte l'action de l'utilisateur.
- Les données (texte du chatbot, filtres du dashboard) sont préparées pour être envoyées vers le serveur ou l'API.

Communication avec les APIs :

- Une requête HTTP est envoyée via fetch() ou AJAX au serveur ou directement aux APIs (OpenAI et Microsoft Learn).
- Microsoft Learn fournit les données sur les cours, parcours et certifications.
- OpenAI génère des réponses intelligentes en fonction du contexte de la question de l'utilisateur.

Retour des données (Backend → Frontend) :

- Les résultats (sous forme de JSON) sont renvoyés au fichier script.js.
- Le JavaScript analyse et traite ces données.

Affichage dynamique (Interface utilisateur) :

- Les recommandations sont affichées dans la zone de chat.
- Les indicateurs, graphiques (Plotly) et tableaux du dashboard sont mis à jour de manière interactive.

En résumé, le workflow suit ce cycle :

Utilisateur → Interface (HTML/CSS/JS) → API (OpenAI/Microsoft Learn) → Traitement → Retour JSON → Affichage dynamique.

5.4 Tests et validation de l'application

5.4.1 Méthodologie de test adoptée

Afin d'assurer la fiabilité et la performance du projet RoboMicrosoft Learn, une méthodologie de test progressive a été mise en place. Elle repose sur différents niveaux de

vérification, de la validation unitaire jusqu'aux tests utilisateurs :

1. Tests unitaires

- o Vérification du bon fonctionnement des composants individuels (fonctions JavaScript, formulaires HTML, éléments CSS).
- o Exemples : test de la saisie dans le chatbot, vérification du rendu d'un graphique simple via Plotly.

2. Tests d'intégration

- o Validation de la communication entre les différentes parties du système (Frontend → API → Backend).
- o Exemple : vérifier que la question saisie dans le chatbot déclenche bien un appel à l'API OpenAI et que la réponse est affichée correctement.

3. Tests fonctionnels

- o Vérification des fonctionnalités principales par rapport au cahier des charges.
- o Exemples :
 - Recommandations de formations pertinentes via le chatbot.
 - Application correcte des filtres dans le dashboard.
 - Génération dynamique des indicateurs (KPIs).

4. Tests de performance

- o Mesure du temps de réponse lors des requêtes API et du chargement des graphiques.
- o Vérification que l'application reste fluide même avec un grand volume de données.

5. Tests de sécurité

- o Validation de la gestion des accès (ex : protection des clés API).
- o Vérification que les données sensibles ne sont pas exposées dans le code client.

6. Tests utilisateurs (recette)

- o Mise en situation avec des utilisateurs finaux (stagiaires et encadrants).
- o Recueil des retours sur l'ergonomie, la clarté des réponses et l'expérience globale.

Conclusion

En résumé, le projet repose sur une architecture modulaire et claire, combinant stockage de données, backend via Flask, et interface utilisateur dynamique en HTML, CSS et JavaScript. Cette organisation permet d'assurer un flux de données fluide, une gestion efficace des requêtes et une expérience utilisateur optimisée, tout en offrant une base solide pour l'évolution et l'amélioration du système.

Chapitre 6

Démonstration et illustrations du projet

Introduction

Ce chapitre présente la démonstration concrète du projet à travers des captures d'écran et des illustrations des différentes interfaces et fonctionnalités. L'objectif est de visualiser le fonctionnement réel du Cobot Microsoft Learn, ainsi que la manière dont les utilisateurs peuvent interagir avec l'application pour obtenir des recommandations personnalisées et consulter les analyses de données.

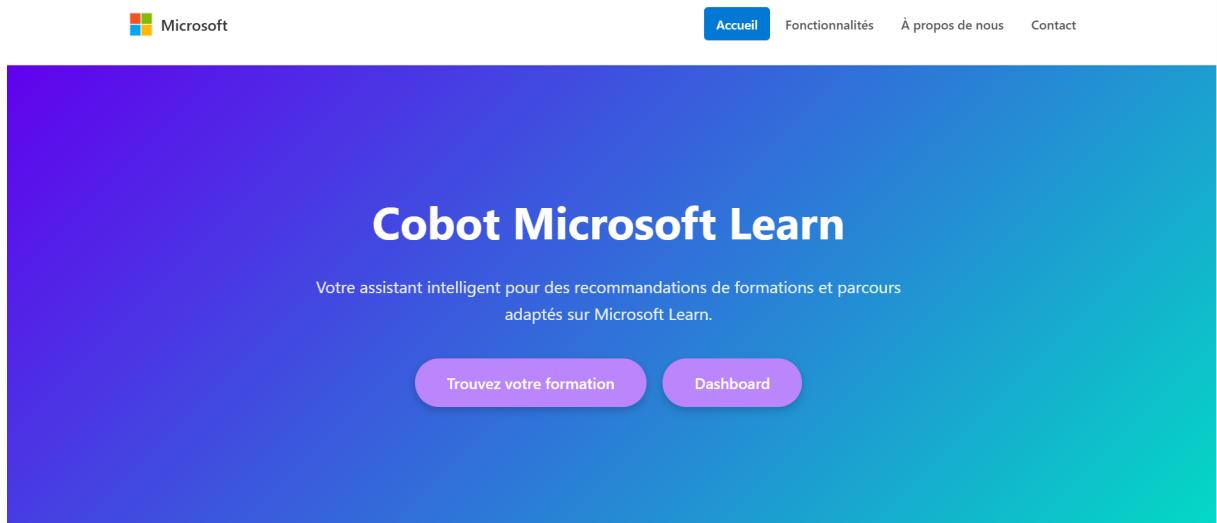


FIGURE 6.1 – Page d'accueil

Cette capture montre la page d'accueil de l'application Cobot Microsoft Learn. On y voit un fond dégradé bleu/violet, le logo Microsoft en haut à gauche, et un menu de navigation avec les onglets "**Accueil**", "**Fonctionnalités**", "**À propos de nous**" et "**Contact**".

Au centre, le titre "**Cobot Microsoft Learn**" est affiché en gros caractères blancs, accompagné d'un sous-titre explicatif : "**Votre assistant intelligent pour des recommandations de formations et parcours adaptés sur Microsoft Learn.**" Deux boutons interactifs sont proposés : "**Trouvez votre formation**" et "**Dashboard**", invitant l'utilisateur à explorer les fonctionnalités principales.

Chapitre 6. Démonstration et illustrations du projet

The screenshot shows the application's main page with a navigation bar at the top featuring the Microsoft logo, Accueil, Fonctionnalités (highlighted in blue), À propos de nous, and Contact. Below the navigation bar is a large teal header bar. The main content area is titled "Fonctionnalités Clés". It contains three cards:

- Recommandations Personnalisées** (Personalized Recommendations): An icon of a graduation cap. Description: "Obtenez des suggestions de cours et de parcours basées sur vos intérêts et votre niveau."
- Intégration Microsoft Learn**: An icon of a cube. Description: "Accédez directement aux ressources officielles de Microsoft Learn pour un apprentissage structuré."
- Analyse des données**: An icon of a line graph. Description: "Visualisez des statistiques et des tendances pour mieux comprendre les données de formation."

FIGURE 6.2 – Fonctionnalités Clés

Cette capture présente la section "Fonctionnalités Clés" de l'application. On y trouve trois cartes illustrant les principales fonctionnalités :

- **Recommandations Personnalisées** (suggestions de cours et parcours basées sur les intérêts et le niveau de l'utilisateur).
- **Intégration Microsoft Learn** (accès direct aux ressources officielles pour un apprentissage structuré).
- **Analyse des données** (visualisation des statistiques et tendances pour mieux comprendre les données de formation).

Chaque carte est accompagnée d'une icône et d'une brève description. Le menu de navigation en haut de page est mis en évidence avec l'onglet "Fonctionnalités" sélectionné.

The screenshot shows the "À propos de nous" (About Us) page. The navigation bar at the top is identical to Figure 6.2, with Fonctionnalités selected. The main content area is titled "À propos de nous". It contains the following text and information:

Ce projet a été réalisé par Hiba Kharfasse, étudiante en Big Data et Intelligence Artificielle à l'ENSA de Tétouan et stagiaire au sein de l'entreprise DYN IT.

DYN IT est une entreprise spécialisée dans les solutions numériques et technologiques innovantes. Elle accompagne ses clients dans la transformation digitale, le développement d'applications sur mesure et l'intégration de systèmes intelligents afin de répondre aux besoins croissants en matière de digitalisation et d'efficacité opérationnelle.

La réalisation de ce projet a également bénéficié du soutien et de l'accompagnement de :

- Monsieur Hassan Fadili
- Anas Belebbes
- Youssef Ghigane

FIGURE 6.3 – À propos de nous

Cette capture d'écran montre la section "**À propos de nous**" du projet Cobot Microsoft Learn. Elle détaile la réalisation du projet par **Hiba Kharfasse**, étudiante en **Big Data et Intelligence Artificielle à l'ENSA de Tétouan** et stagiaire chez **DYN IT**. Une description de l'entreprise **DYN IT** est également fournie, mettant en avant son expertise dans les solutions numériques et technologiques, notamment la transformation digitale, le développement d'applications sur mesure et l'intégration de systèmes intelligents. Enfin, les remerciements sont adressés à Monsieur **Hassan Fadili**, **Anas Belebbes**, et **Youssef Ghigane** pour leur soutien et accompagnement dans la réalisation du projet.

The screenshot shows the 'About Us' section of a website. At the top, there is a navigation bar with links for 'Accueil', 'Fonctionnalités', 'À propos de nous' (which is highlighted in blue), and 'Contact'. Below the navigation bar, there is a text box containing information about the realization of the project and the names of the supporters: Hassan Fadili, Anas Belebbes, and Youssef Ghigane. At the bottom of the screenshot, there is a 'Contactez-nous' section with contact details for Hiba Kharfasse and Dyn IT, along with a description of the company's location in Rabat.

Microsoft

Accueil Fonctionnalités À propos de nous Contact

La réalisation de ce projet a également bénéficié du soutien et de l'accompagnement de :

- Monsieur Hassan Fadili
- Anas Belebbes
- Youssef Ghigane

Contactez-nous

Hiba Kharfasse : hibakharfasse8@gmail.com
Telephone Dyn IT : 05377-31201
L'entreprise DYN IT MAROC est située à l'adresse principale 51 Avenue Tadla à Rabat, dans le quartier d'Aviation. Elle dispose également d'une présence au TechnoPark de Rabat.

FIGURE 6.4 – Contactez-nous

Cette capture d'écran présente la section "**Contactez-nous**" du projet Cobot Microsoft Learn. Elle inclut les coordonnées de **Hiba Kharfasse** (hibakharfasse8@gmail.com) et le numéro de téléphone de **DYN IT** (**05377-31201**). Une description de la localisation de l'entreprise **DYN IT MAROC** est également fournie : située au 51 Avenue Tadla à Rabat, dans le quartier d'Aviation, avec une présence au TechnoPark de Rabat. Cette section permet aux utilisateurs de contacter facilement l'équipe pour toute question ou collaboration.

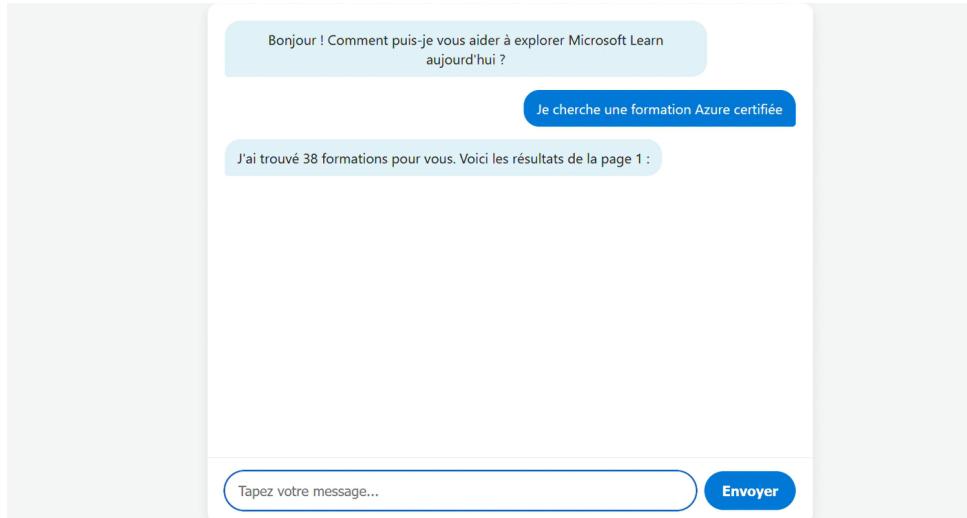


FIGURE 6.5 – Interface du Chatbot

Cette capture montre l'interface du chatbot Cobot Microsoft Learn en action, qui s'apparaît après une clique sur '**Trouvez une formation**'. Le chatbot répond à une demande utilisateur : "Bonjour ! Comment puis-je vous aider à explorer Microsoft Learn aujourd'hui ?". L'utilisateur a saisi une requête ,ex : "**Je cherche une formation Azure certifiée**". Le chatbot indique avoir trouvé **38 formations** et propose d'afficher les résultats de la première page. En bas, un champ de texte permet à l'utilisateur de taper un nouveau message, avec un bouton "**Envoyer**" pour soumettre sa demande.

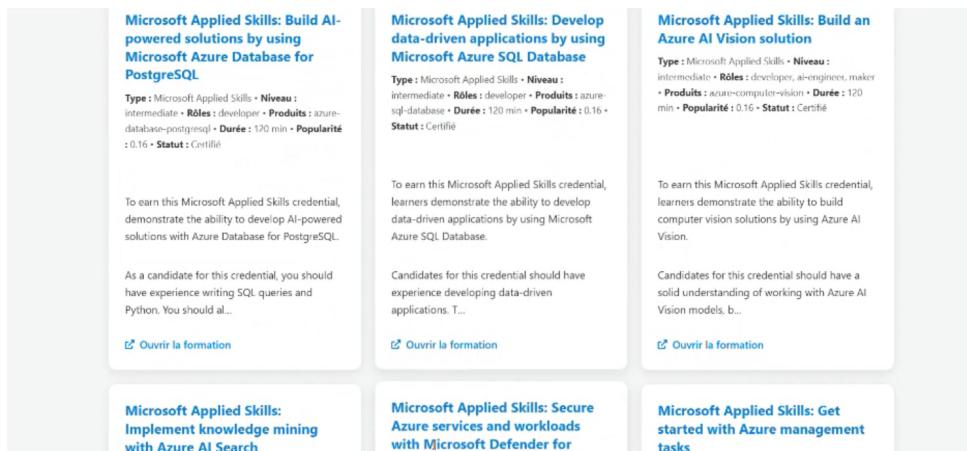


FIGURE 6.6 – Résultats de recherche

Cette capture d'écran affiche **les résultats de recherche** proposés par le chatbot pour des formations Azure certifiées. Dix cartes de formation sont visibles par page, chacune indiquant **le type**, **le niveau**, **le rôle**, **le produit**, **la durée**, **la popularité**, **et le statut**. Chaque carte contient un titre, une brève description des compétences à démontrer, et un bouton "**Ouvrir la formation**" pour accéder au détail.



FIGURE 6.7 – Tableau de bord

Cette capture montre le tableau de bord d'analyse des données du projet Cobot Microsoft Learn, qui s'apparaît après une clique sur 'Dashboard'. À gauche, des filtres permettent de sélectionner les niveaux (beginner, intermediate, advanced) et les types (Microsoft Applied Skills, cert, course, exam, learningPath, module) comme filtre. En dessous, la section "Indicateurs Clés" affiche quatre métriques : Total items (4706), Durée totale (heures) (12056), Popularité moyenne (0.22), et % Certifiés (5.0%). Un bouton "Mettre à jour l'analytique" permet de rafraîchir les données en fonction des filtres appliqués.



FIGURE 6.8 – Visualisation des données – Répartition par niveau et par type

Cette capture d'écran montre deux graphiques du tableau de bord d'analyse des données. À gauche, un histogramme illustre la répartition par niveau (beginner, intermediate, advanced), avec un axe vertical indiquant le nombre d'items (Count) et un axe horizontal listant les niveaux. À droite, un diagramme circulaire présente la répartition par type (module, learningPath, cert, course, Microsoft Applied Skills, exam) avec des pourcentages : les modules dominent à 73,8%, suivis des learningPath à 18,1%, et les autres types en plus petites proportions.

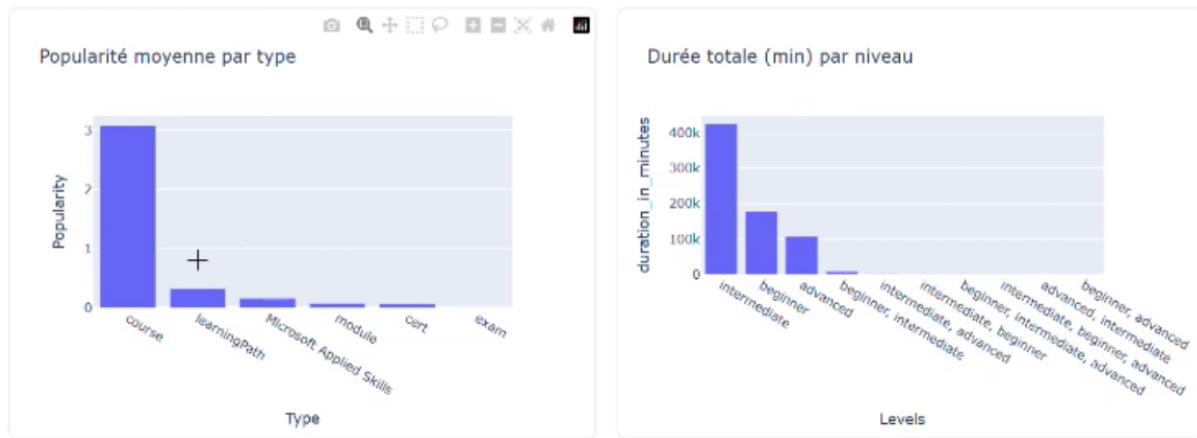
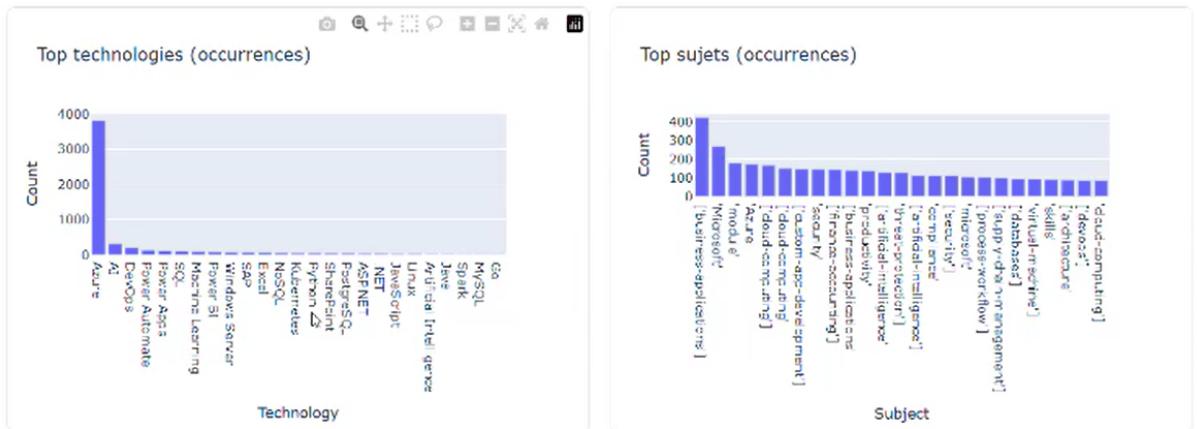


FIGURE 6.9 – Analyse des données – Popularité moyenne par type et durée totale par niveau

Cette capture d'écran présente deux graphiques supplémentaires du tableau de bord. À gauche, un histogramme montre la popularité moyenne par type (course, learningPath, Microsoft Applied Skills, module, cert, exam), où les "course" ont la popularité la plus élevée. À droite, un histogramme illustre la durée totale en minutes par niveau (beginner, intermediate, advanced), mettant en évidence que le niveau "beginner" a la durée totale la plus longue, suivie par "intermediate" et "advanced".



Aperçu des données filtrées								
cls	Popularity	Products	Roles	Title	Type	Url		duration_i
inner	0.0864642169884599	dotnet, vs-code	higher-ed-educator, lc-12-educator, student	Add decision logic to your code using 'if', 'else', and 'else if' statements in C#	module	https://learn.microsoft.com/en-us/training/modules/csharp-if-elseif-else/?WT.mc_id=api_CatalogApi		56
immediate	0.04943898053954388	azure sdks	developer	Implement authentication by using the Microsoft Authentication Library	module	https://learn.microsoft.com/en-us/training/modules/implement-authentication-by-using-microsoft-authentication-library/?WT.mc_id=api_CatalogApi		30
inner	0.09061535591939508	dotnet	developer, student	Get started with dates, times, and time zones	module	https://learn.microsoft.com/en-us/training/modules/get-started-with-dates-times-time-zones/?WT.mc_id=api_CatalogApi		58
immediate	0.09331548521558378	azure-expressroute, azure-virtual-networks	administrator, network-engineer	Design and implement Azure ExpressRoute	module	https://learn.microsoft.com/en-us/training/modules/design-implement-azure-expressroute/?WT.mc_id=api_CatalogApi		57

FIGURE 6.11 – Tableau des données filtrées – Détails des modules de formation

Cette capture d'écran montre **un extrait du dataset Excel** utilisé pour le projet Cobot Microsoft Learn. Le tableau affiche les données filtrées des modules de formation.

Conclusion :

À travers cette démonstration, il est possible d'apprécier l'ergonomie, la clarté et l'efficacité du projet. Les captures d'écran permettent de confirmer que les fonctionnalités développées répondent aux besoins définis et illustrent le résultat final de l'intégration des différents composants techniques.

Conclusion Générale

Le présent projet s'inscrit dans une démarche de conception et de mise en œuvre d'un système intelligent de recommandation exploitant l'API Microsoft Learn, associé à un ensemble de technologies modernes telles que Flask, HTML, CSS et JavaScript. L'objectif principal consistait à développer une solution capable d'intégrer, de traiter et de présenter des données issues de Microsoft Learn de manière ergonomique, sécurisée et exploitable.

L'architecture proposée, articulée autour d'une logique modulaire et cohérente, a permis de couvrir l'ensemble des étapes du processus, depuis la collecte et la préparation des données jusqu'à la visualisation et l'interaction avec l'utilisateur final. L'intégration des mécanismes de sécurité, notamment la protection des données, a renforcé la fiabilité du système, tandis que l'interface utilisateur a été pensée pour favoriser l'accessibilité et l'efficacité d'utilisation.

Sur le plan scientifique et technique, ce travail a démontré la pertinence de l'approche adoptée, en combinant l'exploitation d'APIs, les outils de développement web et les méthodologies de test, pour parvenir à une solution robuste et extensible. Toutefois, des perspectives d'amélioration demeurent envisageables, telles que l'implémentation de modèles de machine learning plus avancés, l'optimisation des performances ou encore l'intégration de nouvelles sources de données afin d'élargir le champ d'application du robot.

En définitive, ce projet constitue une contribution significative à la mise en place de systèmes intelligents de recommandation, tout en offrant un cadre solide pour de futures évolutions et innovations dans le domaine.

Bibliographie

@misc{microsoftlearn2025, author=Microsoft, title=Microsoft Learn API Documentation, year=2025, howpublished=<https://learn.microsoft.com/>, note=Accessed : 2025-09-07}

@iscpandas2024, author=Pandas Development Team, title=Pandas Documentation, year=2024, howpublished=<https://pandas.pydata.org/docs/>, note=Accessed : 2025-09-07

@article{pedregosa2011scikit, title=Scikit-learn : Machine learning in Python, author=Pedregosa, Fabian and Varoquaux, Gaël and Gramfort, Alexandre and others, journal=Journal of Machine Learning Research, volume=12, pages=2825–2830, year=2011}

@miscoython2024, author=Python Software Foundation, title=Python Documentation, year=2024, howpublished=<https://docs.python.org/3/>, note=Accessed : 2025-09-07

@misc{wikipedia2025, author=Wikipedia contributors, title=Wikipedia, The Free Encyclopedia, year=2025, howpublished=<https://www.wikipedia.org/>, note=Accessed : 2025-09-07}