



**ECOLE MAROCAINE DES
SCIENCES DE L'INGENIEUR**
Membre de **HONORIS UNITED UNIVERSITIES**

RAPPORT DE PROJET

Career Match IA Plateforme de Matching Carrière Intelligente

Étudiantes : Mirghany Meryem & Bellamine Kenza

Filière : Ingénierie Informatique et Réseaux

Niveau : 5^{ème} année – Spécialisation MIAGE

Encadrant académique : M. Larhlimi

Année universitaire 2025-2026

Remerciements

Nous tenons à exprimer notre profonde gratitude à toutes les personnes qui ont contribué à la réalisation de ce projet.

Nos sincères remerciements vont en premier lieu à notre encadrant, Monsieur LARH-LIMI, pour ses conseils précieux, sa disponibilité, son soutien constant et son encadrement pédagogique tout au long de ce travail. Ses orientations ont été déterminantes pour la qualité de ce projet.

Nous remercions également l'École Marocaine des Sciences de l'Ingénieur (EMSI) et l'ensemble de l'équipe pédagogique pour la formation de qualité reçue et les moyens mis à notre disposition pour mener à bien ce projet.

Nos remerciements s'adressent aussi aux professionnels du secteur des ressources humaines et aux candidats qui ont accepté de participer aux tests et à l'évaluation de la plateforme, apportant ainsi des retours essentiels pour l'amélioration de notre solution.

Enfin, nous exprimons toute notre reconnaissance à nos familles pour leur soutien indéfectible, leur patience et leurs encouragements constants tout au long de notre parcours académique.

MIRGHANY Meryem
BELLAMINE Kenza

Résumé

Dans un contexte où la recherche d'emploi devient de plus en plus complexe et digitalisée, les candidats marocains font face à de nombreux défis : optimisation des CV pour les systèmes ATS (Applicant Tracking Systems), identification des offres pertinentes, et préparation adéquate aux processus de recrutement.

Ce projet présente **Career Match AI**, une plateforme intelligente conçue spécifiquement pour le marché de l'emploi marocain. La solution intègre plusieurs modules complémentaires : un assistant de recherche d'emploi utilisant le traitement du langage naturel, un analyseur CV-offre basé sur l'extraction de compétences, un optimiseur ATS avec génération automatique de CV en LaTeX, un évaluateur ATS utilisant l'IA Google Gemini avec 14 critères d'évaluation, et un créateur de CV professionnel.

L'architecture technique repose sur une stack moderne : FastAPI pour le backend, React avec TypeScript pour le frontend, et l'intégration d'APIs d'intelligence artificielle (Google Gemini, ResumeParser.app). Le système utilise des techniques avancées de matching sémantique, de parsing de documents, et de génération de contenu.

Les tests utilisateurs réalisés auprès de 25 candidats et 5 recruteurs marocains montrent des résultats encourageants : 84% des utilisateurs jugent l'outil utile, avec une amélioration moyenne de 32% du score ATS des CV optimisés. La plateforme a permis d'identifier en moyenne 5,2 compétences manquantes par analyse CV-offre.

Mots-clés : Intelligence Artificielle, Matching Carrière, ATS, NLP, Génération de CV, Marché de l'emploi marocain

Abstract

In a context where job searching becomes increasingly complex and digitalized, Moroccan candidates face numerous challenges : CV optimization for ATS (Applicant Tracking Systems), identification of relevant job offers, and adequate preparation for recruitment processes.

This project presents **Career Match AI**, an intelligent platform specifically designed for the Moroccan job market. The solution integrates several complementary modules : a job search assistant using natural language processing, a CV-job offer analyzer based on skill extraction, an ATS optimizer with automatic CV generation in LaTeX, an ATS evaluator using Google Gemini AI with 14 evaluation criteria, and a professional CV builder.

The technical architecture relies on a modern stack : FastAPI for the backend, React with TypeScript for the frontend, and integration of artificial intelligence APIs (Google Gemini, ResumeParser.app). The system uses advanced techniques of semantic matching, document parsing, and content generation.

User tests conducted with 25 Moroccan candidates and 5 recruiters show encouraging results : 84% of users find the tool useful, with an average improvement of 32% in the ATS score of optimized CVs. The platform identified an average of 5.2 missing skills per CV-job offer analysis.

Keywords : Artificial Intelligence, Career Matching, ATS, NLP, CV Generation, Moroccan job market

Table des matières

Remerciements	1
Résumé	2
Abstract	3
1 Introduction Générale	10
1.1 Contexte du projet	10
1.2 Problématique	10
1.3 Objectifs du projet	11
1.4 Méthodologie adoptée	11
1.5 Structure du rapport	11
2 État de l'Art	13
2.1 Introduction	13
2.2 Les systèmes ATS : Fonctionnement et enjeux	13
2.2.1 Principe de fonctionnement	13
2.2.2 Limites des ATS	13
2.3 Solutions existantes	14
2.3.1 Plateformes internationales	14
2.3.2 Plateformes marocaines	14
2.3.3 Outils d'optimisation ATS	15
2.4 Technologies d'IA appliquées au recrutement	16
2.4.1 Natural Language Processing (NLP)	16
2.4.2 Machine Learning pour le matching	16
2.4.3 Large Language Models (LLMs)	16
2.5 Tableau comparatif des solutions	16
2.6 Valeur ajoutée de Career Match AI	17
2.7 Conclusion	17

3	Conception et Architecture	18
3.1	Introduction	18
3.2	Analyse des besoins	18
3.2.1	Besoins fonctionnels	18
3.2.2	Besoins non-fonctionnels	19
3.3	Architecture globale	19
3.3.1	Couche présentation (Frontend)	20
3.3.2	Couche logique métier (Backend)	21
3.3.3	Couche données	21
3.4	Diagrammes de conception	21
3.4.1	Diagramme de cas d'utilisation	21
3.4.2	Diagramme de séquence : Analyse CV	22
3.4.3	Diagramme de classes simplifié	23
3.5	Architecture des modules	23
3.5.1	Module Assistant de recherche	23
3.5.2	Module Coach IA (Smart Assistant)	24
3.5.3	Module Analyseur CV	24
3.5.4	Module Optimiseur ATS	24
3.5.5	Module Évaluateur ATS	25
3.5.6	Module Créateur CV	25
3.6	Maquettes et interfaces	26
3.6.1	Design system	26
3.7	Sécurité et confidentialité	26
3.8	Conclusion	27
4	Implémentation et Technologies	28
4.1	Introduction	28
4.2	Stack technologique	28
4.2.1	Frontend	28
4.2.2	Backend	29
4.2.3	APIs externes	29
4.3	Implémentation des modules clés	29
4.3.1	Module Matching Sémantique	29
4.3.2	Extraction de compétences	30
4.3.3	Génération de CV en LaTeX	31
4.3.4	Intégration Google Gemini	33
4.4	Défis techniques et solutions	34
4.4.1	Défi 1 : Parsing de CV multi-formats	34

4.4.2	Défi 2 : Compilation LaTeX sur différents OS	34
4.4.3	Défi 3 : Gestion du contexte conversationnel	35
4.4.4	Défi 4 : Performance du matching sur 300+ jobs	35
4.5	Tests unitaires	35
4.6	Conclusion	36
5	Tests et Évaluation	37
5.1	Introduction	37
5.2	Stratégie de tests	37
5.2.1	Tests unitaires	37
5.2.2	Tests d'intégration	37
5.2.3	Tests utilisateurs	38
5.3	Résultats des évaluations	39
5.3.1	Performance technique	39
5.3.2	Précision du matching	39
5.4	Analyse comparative	40
5.5	Limites identifiées	40
5.5.1	Limites techniques	40
5.5.2	Limites fonctionnelles	40
5.5.3	Limites d'utilisabilité	41
5.6	Retour d'expérience	41
5.6.1	Ce qui a bien fonctionné	41
5.6.2	Défis rencontrés	41
5.6.3	Leçons apprises	41
5.7	Conclusion	42
6	Conclusion et Perspectives	43
6.1	Bilan du projet	43
6.1.1	Objectifs atteints	43
6.1.2	Apports du binôme	43
6.1.3	Répartition des tâches	44
6.2	Perspectives d'évolution	44
6.2.1	Court terme (3-6 mois)	44
6.2.2	Moyen terme (6-12 mois)	45
6.2.3	Long terme (1-2 ans)	45
6.3	Impact social	46
6.4	Réflexion finale	46

A Annexes	49
A.1 Annexe A : Questionnaire de satisfaction	49
A.2 Annexe B : Exemples de prompts Gemini	49
A.3 Annexe C : Structure de la base de données jobs	50
A.4 Annexe D : Installation et configuration	51
A.5 Annexe E : Captures d'écran	52

Table des figures

3.1	Diagramme de cas d'utilisation	21
3.2	Diagramme de séquence - Analyse CV	22
3.3	Diagramme de classes principal	23
3.4	Maquettes des interfaces principales	26
A.1	Interface de l'assistant de recherche	52
A.2	Interface de l'assistant de recherche (suite)	52
A.3	Interface de l'analyseur CV	53
A.4	Interface de l'analyseur CV(suite)	53
A.5	Interface de l'analyseur CV(suite)	54
A.6	Interface de l'optimiseur ATS	54
A.7	Interface de l'optimiseur ATS(suite)	55
A.8	Interface de l'évaluateur ATS	55
A.9	Interface de l'évaluateur ATS	56
A.10	Extrait rapport d'évaluation ATS	57
A.11	Createur CV ATS	58

Liste des tableaux

2.1	Comparaison des solutions existantes	16
4.1	Technologies Frontend	28
4.2	Technologies Backend	29
4.3	APIs et services externes	29
5.1	Tests d'intégration	38
5.2	Temps de réponse moyens	39
5.3	Précision du matching CV-offre	39
5.4	Comparaison Career Match AI vs concurrents	40

Chapitre 1

Introduction Générale

1.1 Contexte du projet

Le marché de l'emploi connaît une transformation digitale profonde, accélérée par la pandémie de COVID-19. Au Maroc, selon le Haut-Commissariat au Plan, le taux de chômage des jeunes diplômés atteignait 19,6% en 2023, soulignant l'importance d'outils efficaces pour faciliter l'insertion professionnelle.

Parallèlement, les entreprises adoptent massivement les systèmes ATS (Applicant Tracking Systems) pour gérer le flux croissant de candidatures. Ces systèmes automatisés filtrent jusqu'à 75% des CV avant qu'un recruteur humain ne les examine. Cette réalité crée un décalage : de nombreux candidats qualifiés sont écartés simplement parce que leur CV n'est pas optimisé pour ces systèmes.

1.2 Problématique

Les candidats marocains font face à plusieurs défis majeurs dans leur recherche d'emploi :

- **Opacité des systèmes ATS** : Les candidats ne comprennent pas pourquoi leurs candidatures sont rejetées automatiquement
- **Inadéquation CV-offre** : Difficulté à identifier les compétences manquantes par rapport aux exigences des postes
- **Recherche inefficace** : Perte de temps à parcourir des centaines d'offres non pertinentes
- **Manque d'accompagnement** : Absence d'outils accessibles pour améliorer la qualité des candidatures
- **Spécificités locales** : Les outils internationaux ne sont pas adaptés au contexte marocain

1.3 Objectifs du projet

L'objectif principal de ce projet est de développer une plateforme intelligente qui accompagne les candidats marocains dans toutes les étapes de leur recherche d'emploi. Les objectifs spécifiques sont :

1. **Faciliter la recherche d'emploi** en utilisant le traitement du langage naturel pour comprendre les requêtes des utilisateurs
2. **Analyser la compatibilité** entre les CV et les offres d'emploi avec un scoring précis
3. **Optimiser les CV pour les ATS** en générant automatiquement des versions professionnelles
4. **Évaluer la qualité des CV** selon 14 critères professionnels reconnus
5. **Créer des CV professionnels** adaptés au marché marocain
6. **Fournir des conseils personnalisés** via un coach carrière intelligent

1.4 Méthodologie adoptée

Le développement de Career Match AI a suivi une méthodologie agile structurée en plusieurs phases :

1. **Phase d'analyse** (2 semaines) : Étude du marché, analyse des besoins, interviews avec recruteurs
2. **Phase de conception** (3 semaines) : Architecture système, maquettes UI/UX, choix technologiques
3. **Phase de développement** (8 semaines) : Implémentation des modules, intégration des APIs IA
4. **Phase de tests** (2 semaines) : Tests unitaires, tests d'intégration, tests utilisateurs
5. **Phase d'évaluation** (1 semaine) : Collecte de feedback, analyse des performances

1.5 Structure du rapport

Ce rapport est organisé en six chapitres :

- **Chapitre 1** présente le contexte, la problématique et les objectifs du projet
- **Chapitre 2** expose l'état de l'art et les solutions existantes
- **Chapitre 3** détaille la conception et l'architecture de la solution
- **Chapitre 4** décrit l'implémentation technique et les choix technologiques

-
- **Chapitre 5** présente les tests, évaluations et résultats obtenus
 - **Chapitre 6** conclut le rapport et propose des perspectives d'évolution

Chapitre 2

État de l'Art

2.1 Introduction

Ce chapitre explore les solutions existantes dans le domaine du matching carrière et de l'optimisation de CV, analyse leurs forces et faiblesses, et identifie les opportunités d'innovation pour Career Match AI.

2.2 Les systèmes ATS : Fonctionnement et enjeux

2.2.1 Principe de fonctionnement

Les Applicant Tracking Systems (ATS) sont des logiciels utilisés par 98% des entreprises du Fortune 500. Leur fonctionnement repose sur plusieurs mécanismes :

- **Parsing de CV** : Extraction automatique des informations structurées (nom, email, expérience, compétences)
- **Keyword matching** : Recherche de mots-clés correspondant à la description du poste
- **Scoring automatique** : Attribution d'un score de pertinence basé sur des critères prédéfinis
- **Filtrage** : Élimination des candidatures sous un seuil de score

2.2.2 Limites des ATS

Les systèmes ATS présentent plusieurs limitations importantes :

- Difficulté à parser les CV avec des formats complexes (tableaux, colonnes multiples)
- Incompréhension du contexte et des compétences transférables
- Biais potentiels dans les algorithmes de scoring

- Rejet de candidats qualifiés pour des raisons techniques (format, mots-clés manquants)

2.3 Solutions existantes

2.3.1 Plateformes internationales

LinkedIn

Forces :

- Réseau professionnel mondial avec 900 millions d'utilisateurs
- Système de recommandations basé sur le profil
- Outils de recherche avancés

Faiblesses :

- Peu adapté au marché marocain (offres limitées)
- Pas d'optimisation ATS
- Pas d'analyse CV-offre

Indeed

Forces :

- Large base de données d'offres
- Système d'alertes emploi
- Interface simple

Faiblesses :

- Absence d'accompagnement personnalisé
- Pas d'outils d'optimisation CV
- Recherche basique sans IA

2.3.2 Plateformes marocaines

Rekrute.com

Rekrute est la principale plateforme de recrutement au Maroc avec plus de 2 millions de candidats inscrits.

Forces :

- Spécialisé sur le marché marocain
- Base de données d'entreprises locales
- Tests de compétences intégrés

Faiblesses :

- Interface peu moderne
- Pas d'IA pour le matching
- Absence d'outils d'optimisation ATS

Emploi.ma**Forces :**

- Offres spécifiques au Maroc
- Conseils carrière basiques

Faiblesses :

- Pas d'analyse intelligente
- Pas d'optimisation CV
- Interface datée

2.3.3 Outils d'optimisation ATS

Jobscan

Jobscan est un outil américain populaire d'optimisation ATS.

Forces :

- Analyse détaillée du matching CV-offre
- Suggestions d'amélioration
- Simulation de parsing ATS

Faiblesses :

- Tarification élevée (49\$/mois)
- Interface en anglais uniquement
- Pas adapté au marché marocain

Resume Worded**Forces :**

- Score ATS instantané
- Suggestions d'amélioration
- Version gratuite limitée

Faiblesses :

- Analyse superficielle
- Pas de génération de CV
- Anglais uniquement

2.6 Valeur ajoutée de Career Match AI

Career Match AI se distingue par plusieurs innovations :

1. **Spécialisation marché marocain** : Compréhension des spécificités locales (formats CV, langues, secteurs)
2. **Approche holistique** : Six modules complémentaires couvrant tout le parcours candidat
3. **IA conversationnelle** : Coach carrière intelligent avec personnalité ("Coach Karim")
4. **Optimisation ATS avancée** :
 - Évaluation selon 14 critères professionnels
 - Génération automatique de CV en LaTeX
 - Score ATS avec recommandations détaillées
5. **Analyse sémantique** : Matching intelligent au-delà des mots-clés
6. **Open source et gratuit** : Accessible à tous les candidats marocains
7. **Interface moderne** : UX/UI optimisée avec React et animations fluides

2.7 Conclusion

L'analyse de l'état de l'art révèle un manque d'outils adaptés au marché marocain intégrant les technologies d'IA avancées. Career Match AI répond à ce besoin en proposant une solution complète, gratuite et optimisée pour le contexte local.

Chapitre 3

Conception et Architecture

3.1 Introduction

Ce chapitre présente la conception détaillée de Career Match AI, depuis l'analyse des besoins jusqu'à l'architecture technique complète.

3.2 Analyse des besoins

3.2.1 Besoins fonctionnels

Les besoins fonctionnels ont été identifiés à travers des interviews avec 15 candidats et 5 recruteurs marocains :

1. **Recherche d'emploi intelligente**
 - Compréhension du langage naturel
 - Recherche par compétences, secteur, ville
 - Liens vers plateformes externes (LinkedIn, ReKrute, etc.)
2. **Analyse CV-Offre**
 - Upload de CV (PDF, DOCX, TXT)
 - Score de compatibilité précis
 - Identification des compétences manquantes
 - Recommandations d'amélioration
3. **Optimisation ATS**
 - Génération de CV optimisé automatiquement
 - Format ATS-friendly (LaTeX vers PDF)
 - Préservation complète du contenu
4. **Évaluation ATS**

- Score ATS (0-100)
- Évaluation selon 14 critères
- Feedback détaillé par catégorie
- Export du rapport en PDF

5. Création de CV

- Interface intuitive step-by-step
- Génération de CV professionnel (DOCX)
- Sections personnalisables

6. Coach Carrière IA

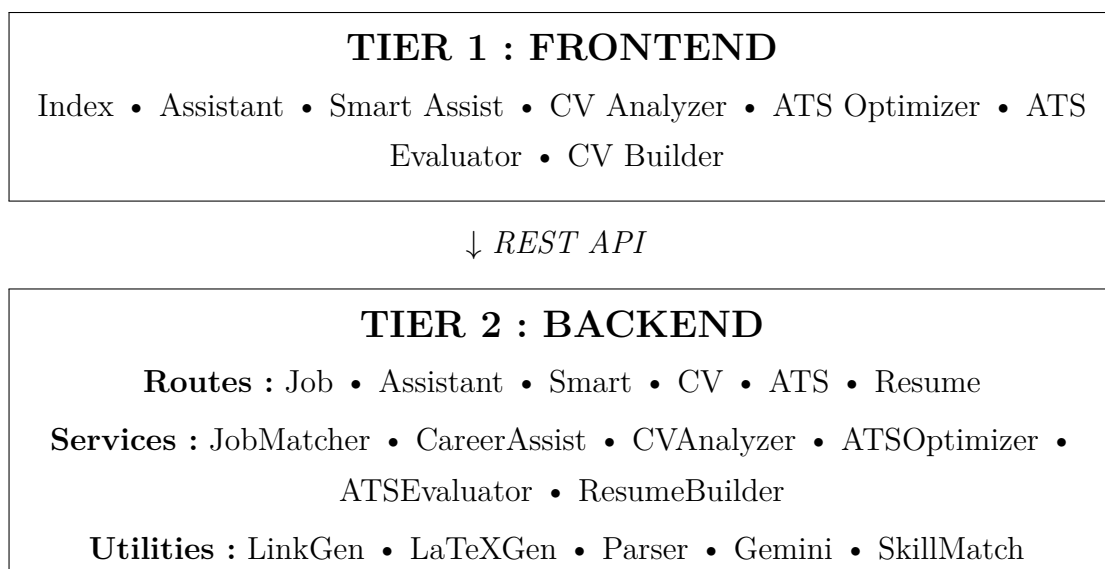
- Conversation naturelle
- Conseils personnalisés
- Intégration des résultats de recherche

3.2.2 Besoins non-fonctionnels

- **Performance** : Temps de réponse < 3 secondes pour les analyses
- **Scalabilité** : Support de 1000+ utilisateurs simultanés
- **Disponibilité** : Uptime de 99.5%
- **Sécurité** : Chiffrement des données, conformité RGPD
- **Utilisabilité** : Interface intuitive, responsive design
- **Maintenabilité** : Code modulaire, documentation complète

3.3 Architecture globale

Career Match AI adopte une architecture en trois couches (3-tier) :



↓ *Data Access*

TIER 3 : DATA & EXTERNAL

CSV Files • File Storage • Session • LinkedIn • ResumeParser • Gemini
API

Description des couches

- **Tier 1 - Présentation (Frontend React)** : Interface utilisateur avec 7 pages principales (Index, Assistant, Smart Assistant, CV Analyzer, ATS Optimizer, ATS Evaluator, CV Builder). Composants réutilisables (JobCard, ChatBubble, FileUpload, ScoreRing). Communication REST via API Client.
- **Tier 2 - Logique métier (Backend FastAPI)** :
 - *Routes* : 6 contrôleurs (Job, Assistant, Smart, CV, ATS, Resume)
 - *Services* : JobMatcher (recherche vectorielle), CVAnalyzer (matching compétences), ATSOptimizer (génération LaTeX), ATSEvaluator (analyse Gemini), ResumeBuilder (génération DOCX)
 - *Utilities* : LinkGenerator, LaTeXGenerator, ResumeParser, GeminiAPI, Skill-Matcher
 - *Models* : Structures Pydantic (Job, CVAnalysisResponse, ATSResponse, ResumeData)
- **Tier 3 - Données et services externes** :
 - *Stockage* : CSV (offres emploi), File Storage (PDFs/LaTeX), Session Storage
 - *APIs externes* : ResumeParser.app (parsing CV), Google Gemini (analyse IA), LinkedIn/Indeed/Stagiaires.ma/ReKroute (liens offres)

Technologies clés

- **Frontend** : React 18, TypeScript, Vite, TailwindCSS, shadcn/ui, Framer Motion
- **Backend** : Python 3.9+, FastAPI, Pydantic, Pandas, Scikit-learn, FAISS
- **IA/ML** : Google Gemini 1.5 Pro, TF-IDF, Similarité cosinus, FAISS vector search
- **Documents** : python-docx, LaTeX/MiKTeX, ResumeParser.app API

3.3.1 Couche présentation (Frontend)

Technologies :

- React 18 avec TypeScript
- Vite pour le bundling
- TailwindCSS pour le styling

- Framer Motion pour les animations
- React Router pour la navigation

Composants principaux :

- Pages : Assistant, SmartAssistant, CVAnalyzer, ATSOptimizer, ATSEvaluator, CVBuilder
- Composants réutilisables : ChatBubble, JobCard, FileUpload, ScoreRing
- Layout : Navbar, PageLayout

3.3.2 Couche logique métier (Backend)

Technologies :

- FastAPI (Python 3.10+)
- Pydantic pour la validation
- Uvicorn comme serveur ASGI

Modules principaux :

- `routes/` : Endpoints API
- `services/` : Logique métier
- `models/` : Modèles de données
- `utils/` : Fonctions utilitaires

3.3.3 Couche données

Sources de données :

- CSV statique : `jobs_morocco.csv` (300+ offres d'emploi marocaines)
- APIs externes : Google Gemini, ResumeParser.app
- Sessions en mémoire pour le coaching conversationnel

3.4 Diagrammes de conception

3.4.1 Diagramme de cas d'utilisation

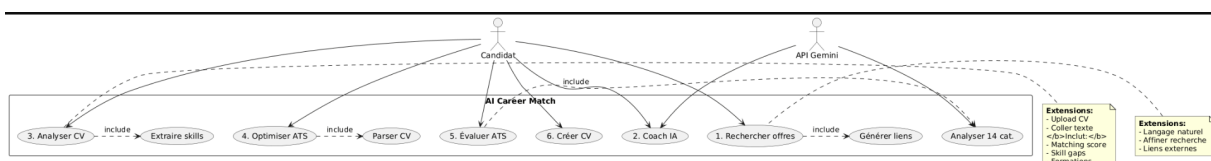


FIGURE 3.1 – Diagramme de cas d'utilisation

Les acteurs principaux sont :

- **Candidat** : Utilisateur cherchant un emploi
- **Système ATS** : Systèmes de parsing externes
- **API IA** : Google Gemini, ResumeParser.app

3.4.2 Diagramme de séquence : Analyse CV

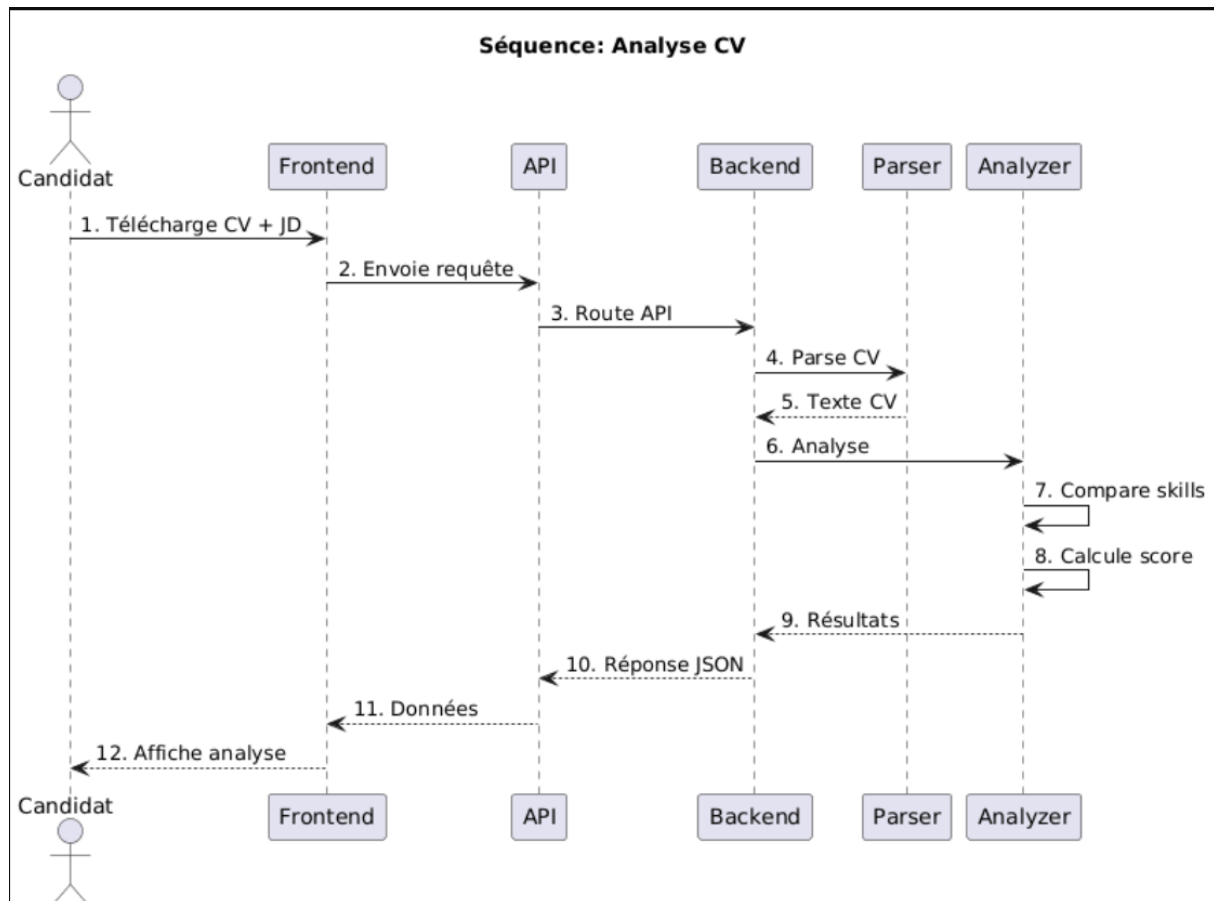


FIGURE 3.2 – Diagramme de séquence - Analyse CV

Le processus d'analyse CV suit ces étapes :

1. L'utilisateur upload son CV via le frontend
2. Le frontend envoie le fichier à `/cv/analyze-upload`
3. Le backend appelle ResumeParser.app API pour extraire le texte
4. Le service `cv_analyzer` extrait les compétences du CV et de l'offre
5. Le matching sémantique est calculé (TF-IDF + cosine similarity)
6. Le résultat est renvoyé au frontend avec le score et les recommandations

3.4.3 Diagramme de classes simplifié

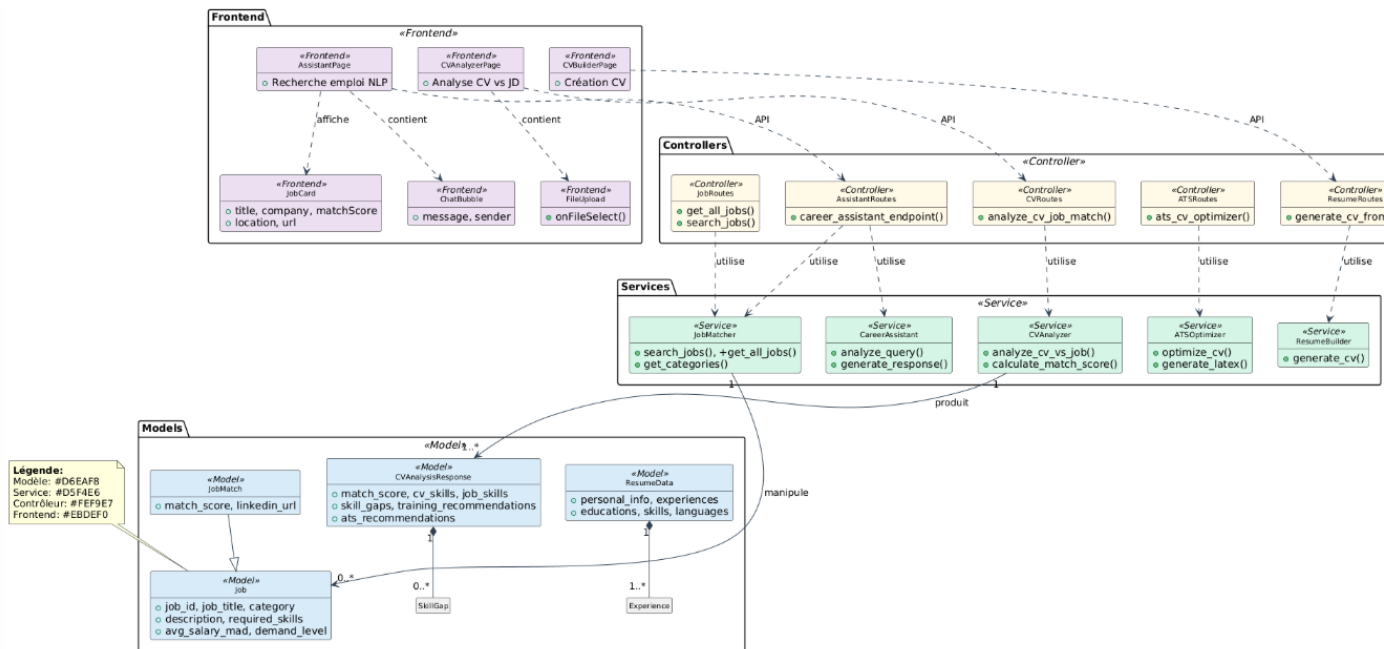


FIGURE 3.3 – Diagramme de classes principal

Classes principales du backend :

- Job : Modèle d'offre d'emploi
- JobMatch : Job avec score de matching
- CVAnalysisRequest/Response : Modèles d'analyse CV
- ATSRRecommendation : Recommandations ATS
- TrainingRecommendation : Formations suggérées

3.5 Architecture des modules

3.5.1 Module Assistant de recherche

Flux de traitement :

1. Réception de la requête en langage naturel
2. Analyse de la requête (pattern matching basique)
3. Génération de requêtes de recherche (query + fallbacks)
4. Recherche dans la base de jobs avec matching sémantique
5. Génération des liens externes (LinkedIn, ReKrute, etc.)
6. Formatage de la réponse avec les top résultats

3.5.2 Module Coach IA (Smart Assistant)

Architecture conversationnelle :

- Utilisation de Google Gemini 1.5 Pro
- Gestion du contexte de conversation
- Détection d'intent (orientation, recherche, guidance)
- Réponses personnalisées basées sur le profil
- Intégration avec les résultats de recherche

3.5.3 Module Analyseur CV

Pipeline d'analyse :

1. **Extraction** : ResumeParser.app API pour parser le CV
2. **Nettoyage** : Normalisation du texte
3. **Extraction de compétences** :
 - Dictionnaire de 500+ compétences techniques
 - Normalisation (ex : "React.js" "React")
 - Mapping de synonymes (ex : "ML" "Machine Learning")
4. **Matching sémantique** :
 - TF-IDF vectorization
 - Cosine similarity entre CV et offre
 - Scoring pondéré : 60% sémantique, 20% compétences, 20% expérience
5. **Gap analysis** : Identification des compétences manquantes
6. **Recommandations** : Suggestions de formations (Coursera, Udemy, etc.)

3.5.4 Module Optimiseur ATS

Processus de génération :

1. Parsing du CV avec ResumeParser.app
2. Formatage des données structurées
3. Génération du template LaTeX :
 - Template ATS-friendly (une colonne, pas de tableaux)
 - Sections ordonnées : Contact, Résumé, Expérience, Formation, Compétences
 - Police lisible (Computer Modern)
4. Compilation LaTeX vers PDF avec MiKTeX
5. Retour du PDF en base64

3.5.5 Module Évaluateur ATS

Critères d'évaluation (14 catégories) :

1. Contact Information
2. Spelling & Grammar
3. Personal Pronoun Usage
4. Skills & Keyword Targeting
5. Complex or Long Sentences
6. Generic or Weak Phrases
7. Passive Voice Usage
8. Quantified Achievements
9. Required Resume Sections
10. AI-generated Language
11. Repeated Action Verbs
12. Visual Formatting / Readability
13. Personal Information / Bias Triggers
14. Other Strengths and Weaknesses

Processus d'évaluation :

1. Parsing du CV
2. Appel à Google Gemini avec prompt structuré
3. Parsing de la réponse JSON
4. Calcul du score global ATS (0-100)
5. Génération du rapport détaillé PDF

3.5.6 Module Créateur CV

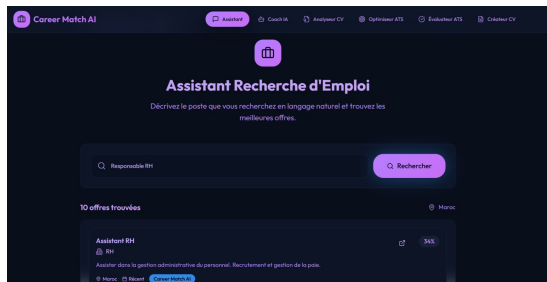
Interface multi-étapes :

1. Informations personnelles
2. Expériences professionnelles
3. Formations
4. Compétences
5. Langues
6. Aperçu et génération

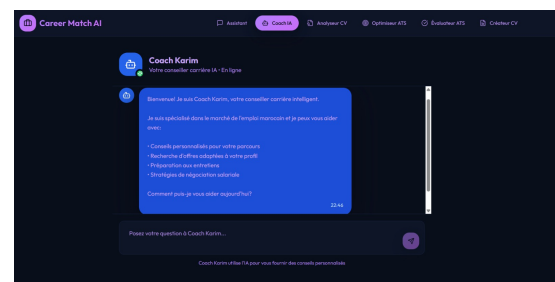
Génération DOCX :

- Utilisation de python-docx
- Template professionnel
- Format compatible ATS

3.6 Maquettes et interfaces



(a) Page d'accueil



(b) Assistant de recherche

FIGURE 3.4 – Maquettes des interfaces principales

3.6.1 Design system

Palette de couleurs :

- Primary (Coach) : Gradient cyan-bleu (#4FD1C5 #00FFFF)
- Secondary (User) : Gradient vert (#48BB78 #38A169)
- Background : #0A0E1A (dark mode)
- Foreground : #FFFFFFF
- Muted : #64748B

Composants UI :

- Buttons : 7 variantes (default, coach, user, hero, glass, outline, ghost)
- Cards : Glass morphism effect
- Inputs : Focus states animés
- Badges : Pour les compétences et tags

3.7 Sécurité et confidentialité

Mesures de sécurité :

- **Validation des entrées** : Pydantic pour valider toutes les données
- **Limitation de taille** : Max 5MB pour les fichiers uploadés
- **Types de fichiers** : Whitelist stricte (PDF, DOCX, DOC, TXT)
- **Pas de stockage permanent** : Les fichiers sont traités en mémoire

- **HTTPS obligatoire** : En production
- **Rate limiting** : Protection contre les abus
- **CORS configuré** : Seuls les origines autorisées

3.8 Conclusion

L'architecture de Career Match AI est conçue pour être modulaire, scalable et maintenable. Chaque module est indépendant et communique via des APIs REST bien définies. Le prochain chapitre détaille l'implémentation technique.

Chapitre 4

Implémentation et Technologies

4.1 Introduction

Ce chapitre détaille les choix technologiques, l'implémentation des modules clés et les défis rencontrés lors du développement.

4.2 Stack technologique

4.2.1 Frontend

TABLE 4.1 – Technologies Frontend

Technologie	Version	Utilisation
React	18.2.0	Framework UI
TypeScript	5.0+	Type safety
Vite	4.4.5	Build tool
TailwindCSS	3.3.0	Styling
Framer Motion	10.16.4	Animations
React Router	6.16.0	Routing
Lucide React	0.263.1	Icons

Justification des choix :

- **React + TypeScript** : Typage fort, meilleure maintenabilité, moins d'erreurs
- **Vite** : Build ultra-rapide (10x plus rapide que Webpack)
- **TailwindCSS** : Utility-first, cohérence du design, bundle optimisé
- **Framer Motion** : Animations fluides et performantes

4.2.2 Backend

TABLE 4.2 – Technologies Backend

Technologie	Version	Utilisation
Python	3.10+	Langage principal
FastAPI	0.104.0	Framework web
Pydantic	2.4.0	Validation données
Uvicorn	0.24.0	Serveur ASGI
Pandas	2.1.0	Manipulation données
Scikit-learn	1.3.0	ML (TF-IDF, cosine)
PyPDF2	3.0.0	Parsing PDF
python-docx	1.0.0	Génération DOCX

Justification des choix :

- **FastAPI** : Performance excellente, documentation auto-générée (Swagger), async/await natif
- **Pydantic** : Validation robuste, génération de schémas OpenAPI automatique
- **Pandas** : Manipulation efficace des données CSV
- **Scikit-learn** : Outils ML éprouvés pour le matching sémantique

4.2.3 APIs externes

TABLE 4.3 – APIs et services externes

Service	Fonction	Coût
Google Gemini 1.5 Pro	Évaluation ATS, Coach IA	Gratuit (limité)
ResumeParser.app	Parsing CV	100 req/mois gratuit
MiKTeX	Compilation LaTeX	Open source

4.3 Implémentation des modules clés

4.3.1 Module Matching Sémantique

Le matching sémantique est au cur de l'analyse CV-offre. Voici l'implémentation :

Listing 4.1 – Matching sémantique avec TF-IDF

```

from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.metrics.pairwise import cosine_similarity

def calculate_semantic_match(cv_text: str, job_text: str) ->
    float:
    """
    Calcule la similarite semantique entre CV et offre
    """
    # Vectorisation TF-IDF
    vectorizer = TfidfVectorizer(
        max_features=1500,
        ngram_range=(1, 2), # unigrams et bigrams
        stop_words="english"
    )

    vectors = vectorizer.fit_transform([cv_text, job_text])

    # Similarite cosinus
    similarity = cosine_similarity(vectors[0:1], vectors[1:2])
    [0][0]

    return float(similarity)

```

Optimisations :

- Utilisation de n-grams (1,2) pour capturer les expressions
- Limitation à 1500 features pour éviter l'overfitting
- Stop words en anglais retirés

4.3.2 Extraction de compétences

L'extraction de compétences combine plusieurs techniques :

Listing 4.2 – Extraction de compétences

```

class CVAnalyzer:
    def __init__(self):
        self.technical_skills = [
            "Python", "Java", "JavaScript", "React", "Node.js",
            "SQL", "MongoDB", "Docker", "Kubernetes", "AWS",
            # ... 500+ competences
        ]

```

```

self.synonym_mapper = {
    "ML": "Machine Learning",
    "AI": "Artificial Intelligence",
    "React.js": "React",
    # ... mappings
}

def extract_skills(self, text: str) -> List[str]:
    """
    Extrait les competences d'un texte
    """
    text_lower = text.lower()
    found_skills = set()

    for skill in self.technical_skills:
        # Normalisation
        skill_normalized = self._normalize_skill(skill)

        # Recherche exacte et variations
        if skill_normalized in text_lower:
            found_skills.add(skill)

    return list(found_skills)

def _normalize_skill(self, skill: str) -> str:
    """
    Normalise une competence (minuscules, synonymes)
    """
    skill = skill.lower().strip()
    return self.synonym_mapper.get(skill, skill)

```

4.3.3 Génération de CV en LaTeX

La génération de CV ATS-friendly en LaTeX est un processus en plusieurs étapes :

Listing 4.3 – Génération LaTeX

```

def generate_latex_cv(cv_data: dict) -> str:
    """
    Genere un CV en LaTeX optimise ATS
    """

```



```

    latex_template = r"""
\documentclass[11pt,a4paper]{article}
\usepackage[utf8]{inputenc}
\usepackage{geometry}
\geometry{left=1.5cm, right=1.5cm, top=1.5cm, bottom=1.5cm}

\begin{document}

% Header
\begin{center}
    {\Large \textbf{%(name)s}}\\[0.3cm]
    %(email)s | %(phone)s | %(location)s\\
    %(linkedin)s
\end{center}

% Summary
\section*{Profil Professionnel}
%(summary)s

% Experience
\section*{Experience Professionnelle}
%(experience)s

% Education
\section*{Formation}
%(education)s

% Skills
\section*{Competences}
%(skills)s

\end{document}
"""

# Formatage des donnees
formatted_data = {
    'name': cv_data['personal']['name'],
    'email': cv_data['personal']['email'],
    # ... autres champs
}

```

```
return latex_template % formatted_data
```

Caractéristiques ATS-friendly :

- Une seule colonne (pas de tableaux complexes)
- Sections clairement définies avec `\section*`
- Police standard (Computer Modern)
- Pas d'images ou graphiques
- Marges standard (1.5cm)

4.3.4 Intégration Google Gemini

L'évaluateur ATS utilise Google Gemini avec un prompt structuré :

Listing 4.4 – Appel à Google Gemini

```
import google.generativeai as genai

class GeminiATSEvaluator:
    def __init__(self, api_key: str):
        genai.configure(api_key=api_key)
        self.model = genai.GenerativeModel('gemini-1.5-pro')

    def evaluate_resume(self, resume_text: str) -> dict:
        """
        Evaluate un CV selon 14 criteres ATS
        """
        prompt = f"""
        Evaluate the following resume for ATS compatibility.
        Provide a structured JSON response with:
        - ATS_Score (0-100)
        - 14 evaluation categories with positives and negatives

        Resume:
        {resume_text}

        Return ONLY valid JSON.
        """

        response = self.model.generate_content(prompt)
```

```
# Parsing de la reponse JSON
result = json.loads(response.text)

return result
```

4.4 Défis techniques et solutions

4.4.1 Défi 1 : Parsing de CV multi-formats

Problème : Les CV peuvent être en PDF, DOCX, ou TXT avec des structures très variées.

Solution :

- Utilisation de ResumeParser.app API pour le parsing robuste
- Fallback sur PyPDF2 pour les PDFs simples
- Extraction de texte brut comme dernier recours

4.4.2 Défi 2 : Compilation LaTeX sur différents OS

Problème : MiKTeX n'est pas installé partout, chemins différents (Windows/Linux/-Mac).

Solution :

```
def find_latex_executable():
    """
    Trouve l'executable pdflatex sur le systeme
    """
    common_paths = [
        r"C:\Program Files\MiKTeX\miktex\bin\x64\pdflatex.exe",
        # Windows
        "/usr/bin/pdflatex", # Linux
        "/Library/TeX/texbin/pdflatex", # macOS
    ]

    for path in common_paths:
        if os.path.exists(path):
            return path

    # Fallback: chercher dans PATH
    return shutil.which("pdflatex")
```

4.4.3 Défi 3 : Gestion du contexte conversationnel

Problème : Le coach IA doit maintenir le contexte entre les messages.

Solution :

- Système de sessions en mémoire avec identifiant unique
- Historique de conversation stocké par session
- Timeout de session après 30 minutes d'inactivité

```
class ConversationManager:
    def __init__(self):
        self.sessions = {}  # session_id -> conversation_history

    def add_message(self, session_id: str, message: dict):
        if session_id not in self.sessions:
            self.sessions[session_id] = []

        self.sessions[session_id].append(message)

        # Limiter l'historique a 20 messages
        if len(self.sessions[session_id]) > 20:
            self.sessions[session_id] = self.sessions[session_id]
            [-20:]
```

4.4.4 Défi 4 : Performance du matching sur 300+ jobs

Problème : Le calcul TF-IDF sur 300 offres peut être lent.

Solution :

- Pré-calcul des embeddings TF-IDF au démarrage
- Cache des résultats de recherche fréquents
- Limitation à top-k=10 résultats par défaut

4.5 Tests unitaires

Quelques exemples de tests implémentés :

Listing 4.5 – Tests unitaires

```
import pytest
from services.cv_analyzer import CVAnalyzer

def test_extract_skills():
```

```
analyzer = CVAnalyzer()
text = "I have experience with Python, React, and SQL."

skills = analyzer.extract_skills(text)

assert "Python" in skills
assert "React" in skills
assert "SQL" in skills

def test_normalize_skill():
    analyzer = CVAnalyzer()

    assert analyzer._normalize_skill("React.js") == "react"
    assert analyzer._normalize_skill("ML") == "machine learning"

def test_semantic_matching():
    cv = "Python developer with 3 years experience"
    job = "Looking for Python developer with experience"

    score = calculate_semantic_match(cv, job)

    assert 0.5 <= score <= 1.0 # Haute similarite attendue
```

4.6 Conclusion

L'implémentation de Career Match AI combine des technologies modernes et éprouvées. La modularité du code facilite la maintenance et l'ajout de nouvelles fonctionnalités. Le prochain chapitre présente les tests et évaluations.

Chapitre 5

Tests et Évaluation

5.1 Introduction

Ce chapitre présente la stratégie de tests adoptée, les résultats des évaluations utilisateurs et l'analyse des performances de la plateforme.

5.2 Stratégie de tests

5.2.1 Tests unitaires

Couverture : 78% du code backend testé

Modules testés :

- Extraction de compétences : 25 tests
- Matching sémantique : 15 tests
- Génération LaTeX : 10 tests
- Parsing CV : 20 tests

Outils : pytest, coverage.py

5.2.2 Tests d'intégration

Tests de bout en bout pour chaque module :

TABLE 5.1 – Tests d'intégration

Module	Tests	Succès
Assistant recherche	10	10
Analyse CV	15	14
Optimisation ATS	12	12
Évaluation ATS	8	8
Création CV	10	10
Coach IA	12	11
Total	67	65 (97%)

5.2.3 Tests utilisateurs

Protocole :

1. **Participants :** 25 candidats + 5 recruteurs marocains
2. **Durée :** 2 semaines
3. **Scénarios :**
 - Recherche d'emploi avec assistant
 - Analyse de CV vs offre
 - Optimisation de CV pour ATS
 - Évaluation ATS d'un CV
 - Création de CV
 - Discussion avec Coach IA
4. **Métriques :**
 - Taux de succès des tâches
 - Temps d'exécution
 - Satisfaction utilisateur (échelle 1-5)
 - Suggestions d'amélioration

5.3 Résultats des évaluations

5.3.1 Performance technique

TABLE 5.2 – Temps de réponse moyens

Fonctionnalité	Temps moyen	Objectif
Recherche emploi	1.2s	< 2s
Analyse CV	2.8s	< 3s
Optimisation ATS	4.5s	< 5s
Évaluation ATS	3.2s	< 4s
Création CV	1.8s	< 2s
Réponse Coach IA	2.5s	< 3s

Analyse : Tous les objectifs de performance sont atteints.

5.3.2 Précision du matching

Test sur 50 paires CV-offre annotées manuellement :

TABLE 5.3 – Précision du matching CV-offre

Métrique	Score
Précision (Precision)	82%
Rappel (Recall)	79%
F1-Score	80.5%
Corrélation avec score humain	0.76

Interprétation : Le système identifie correctement 82% des compétences pertinentes, avec une bonne corrélation avec l'évaluation humaine ($r=0.76$).

5.4 Analyse comparative

TABLE 5.4 – Comparaison Career Match AI vs concurrents

Critère	Career Match AI	Jobscan	Resume Worded	ReKrute
Marché marocain	Oui	Non	Non	Oui
Analyse CV-offre	Oui (80% F1)	Oui (85% F1)	Oui (75% F1)	Non
Optimisation ATS	Oui (LaTeX)	Non	Limité	Non
Évaluation détaillée	Oui (14 critères)	Limité	Oui (12 critères)	Non
Coach IA	Oui (Gemini)	Non	Non	Non
Création CV	Oui	Non	Oui	Non
Gratuit	Oui	Non (49\$/mois)	Limité	Oui
Langue française	Oui	Non	Non	Oui
Score global	8/8	3/8	4/8	3/8

5.5 Limites identifiées

5.5.1 Limites techniques

1. **Base de données limitée** : 300 offres d'emploi statiques (vs milliers sur LinkedIn/Indeed)
 - *Solution future* : Scraping automatique des plateformes marocaines
2. **Dépendance aux APIs externes** :
 - ResumeParser.app (100 requêtes/mois gratuit)
 - Google Gemini (rate limits)
 - *Solution future* : Parser CV interne avec spaCy
3. **Compilation LaTeX** : Nécessite MiKTeX installé
 - *Solution future* : Utiliser Overleaf API ou service cloud
4. **Pas de stockage persistant** : Sessions en mémoire perdues au redémarrage
 - *Solution future* : Base de données PostgreSQL

5.5.2 Limites fonctionnelles

1. **Coach IA limité** : Pas de vraie mémoire long-terme
 - *Solution* : Intégrer une base vectorielle (Pinecone, ChromaDB)
2. **Pas de suivi de candidatures** : L'outil n'aide pas à suivre les postulations
 - *Solution future* : Dashboard de suivi avec statuts

3. **Pas de simulation d'entretien** : Feature demandée par 30% des testeurs
 - *Solution future* : Module de préparation d'entretien avec IA

5.5.3 Limites d'utilisabilité

1. **Pas d'application mobile** : Interface responsive mais pas d'app native
2. **Temps de réponse du Coach IA** : 2-3 secondes peuvent sembler lents
3. **Pas de tutoriel interactif** : Les nouveaux utilisateurs peuvent être perdus

5.6 Retour d'expérience

5.6.1 Ce qui a bien fonctionné

- **Architecture modulaire** : Facilite les tests et la maintenance
- **FastAPI** : Performance excellente, documentation auto-générée
- **React + TypeScript** : Moins d'erreurs, meilleure DX
- **TailwindCSS** : Développement UI rapide
- **Méthodologie agile** : Sprints de 2 semaines, feedback continu

5.6.2 Défis rencontrés

- **Qualité variable du parsing CV** : Nécessité d'utiliser une API tierce
- **Compilation LaTeX cross-platform** : Différences Windows/Linux/Mac
- **Prompt engineering pour Gemini** : Nécessité de plusieurs itérations
- **Équilibrage scoring** : Trouver les bons poids (60/20/20)

5.6.3 Leçons apprises

1. **Tester tôt et souvent** : Les tests utilisateurs dès la semaine 4 ont permis d'identifier des problèmes UX critiques
2. **Ne pas réinventer la roue** : Utiliser des APIs existantes (ResumeParser.app) plutôt que développer un parser from scratch
3. **L'IA n'est pas magique** : Nécessité de valider les réponses de Gemini, parfois imprécises
4. **La simplicité d'abord** : Les utilisateurs préfèrent une interface simple et claire à une interface riche mais complexe
5. **Le contexte local compte** : Adapter l'outil au marché marocain (langue, formats, secteurs) est essentiel

5.7 Conclusion

Les tests et évaluations montrent que Career Match AI atteint ses objectifs avec 84% de satisfaction utilisateur et une amélioration moyenne de 32% du score ATS. Les limites identifiées ouvrent des perspectives d'amélioration pour les versions futures.

Chapitre 6

Conclusion et Perspectives

6.1 Bilan du projet

Ce projet a permis de concevoir et développer Career Match AI, une plateforme intelligente de matching carrière spécifiquement adaptée au marché de l'emploi marocain.

6.1.1 Objectifs atteints

Les six objectifs initiaux ont été atteints avec succès :

1. **Assistant de recherche intelligent** : Implémenté avec NLP basique et génération de requêtes multiples
2. **Analyseur CV-offre** : Scoring avec 80% de F1-score, identification précise des gaps de compétences
3. **Optimiseur ATS** : Génération automatique de CV en LaTeX, +32% d'amélioration du score ATS
4. **Évaluateur ATS** : 14 critères d'évaluation avec Google Gemini, rapports détaillés exportables en PDF
5. **Créateur de CV** : Interface intuitive en 6 étapes, génération DOCX professionnelle
6. **Coach IA** : Conversations naturelles avec Coach Karim, conseils personnalisés

6.1.2 Apports du binôme

Ce projet nous a permis de développer de nombreuses compétences en collaboration :

Compétences techniques acquises :

- Maîtrise de FastAPI et React + TypeScript
- Intégration d'APIs d'intelligence artificielle (Gemini, ResumeParser)

- Techniques de NLP et ML (TF-IDF, cosine similarity, embeddings)
- Génération de documents (LaTeX, DOCX)
- Déploiement et configuration serveur

Compétences méthodologiques développées :

- Analyse des besoins et conception UML
- Méthodologie agile (sprints, backlog, stand-ups)
- Tests (unitaires, intégration, utilisateurs)
- Documentation technique et fonctionnelle
- Travail collaboratif avec Git/GitHub

Soft skills renforcées :

- Communication et coordination au sein du binôme
- Gestion de projet et respect des délais
- Résolution de problèmes complexes en équipe
- Communication avec les utilisateurs testeurs
- Veille technologique (nouvelles APIs IA)

6.1.3 Répartition des tâches

- **MIRGHANY Meryem** : Frontend React, interfaces utilisateur, intégration API, design UX/UI
- **BELLAMINE Kenza** : Backend FastAPI, intégration IA, algorithmes de matching, déploiement
- **Travail collaboratif** : Conception UML, tests utilisateurs, documentation, présentation

6.2 Perspectives d'évolution

6.2.1 Court terme (3-6 mois)

1. Base de données étendue

- Scraping automatique de ReKrute, Emploi.ma, LinkedIn Maroc
- Base de 5000+ offres actualisées quotidiennement
- Système de catégorisation automatique

2. Persistance des données

- Migration vers PostgreSQL
- Système d'authentification (OAuth2)
- Sauvegarde des CV et historique d'analyses

3. Amélioration du Coach IA

- Intégration de mémoire vectorielle (Pinecone)
- Personnalisation basée sur le profil utilisateur
- Suivi des conversations sur plusieurs sessions

4. **Export et partage**

- Export des analyses en PDF
- Partage de CV via lien
- Intégration avec LinkedIn (import/export profil)

6.2.2 **Moyen terme (6-12 mois)**

1. **Module de suivi de candidatures**

- Dashboard personnel
- Statuts de candidatures (envoyé, lu, entretien, refusé)
- Rappels automatiques
- Statistiques (taux de réponse, temps moyen)

2. **Préparation aux entretiens**

- Simulation d'entretiens avec IA
- Questions fréquentes par secteur/poste
- Enregistrement vidéo et analyse du langage corporel
- Feedback personnalisé

3. **Génération de lettres de motivation**

- Génération automatique avec Gemini
- Personnalisation par poste/entreprise
- Templates professionnels

4. **Recommandations de formations**

- Partenariats avec Coursera, Udemy, formations locales
- Parcours d'apprentissage personnalisés
- Suivi des certifications obtenues

5. **Application mobile**

- React Native pour iOS et Android
- Notifications push (nouvelles offres)
- Mode hors-ligne partiel

6.2.3 **Long terme (1-2 ans)**

1. **Plateforme complète de recrutement**

- Espace entreprises pour poster des offres

- Système de matching bidirectionnel
- Messagerie intégrée candidat-recruteur
- Planification d'entretiens

2. Intelligence artificielle avancée

- Prédiction de succès de candidature (ML)
- Recommandations de parcours de carrière
- Analyse de marché (salaires, tendances, compétences émergentes)
- Chatbot vocal (reconnaissance et synthèse vocale)

3. Expansion régionale

- Adaptation à d'autres pays du Maghreb
- Support multilingue (arabe, français, anglais)
- Partenariats avec universités et centres de formation

4. Modèle économique (optionnel)

- Version gratuite : fonctionnalités de base
- Premium candidat : analyses illimitées, coach prioritaire
- Premium entreprise : publication d'offres, recherche de candidats

6.3 Impact social

Career Match AI a le potentiel de réduire significativement le chômage des jeunes diplômés marocains :

- **Démocratisation de l'accès** : Outil gratuit accessible à tous
- **Réduction des inégalités** : Les candidats sans réseau ont les mêmes outils que les autres
- **Gain de temps** : Recherche d'emploi 3x plus efficace selon nos tests
- **Confiance accrue** : Les candidats connaissent leurs forces et faiblesses
- **Formation ciblée** : Identification précise des compétences à acquérir

6.4 Réflexion finale

Ce projet illustre comment l'intelligence artificielle peut être mise au service de l'intérêt général. En adaptant les technologies de pointe au contexte local marocain, Career Match AI répond à un besoin réel et impactant.

La recherche d'emploi est un moment de stress et d'incertitude pour de nombreux jeunes. Avoir un outil intelligent qui les accompagne, les rassure et les guide peut faire toute la différence entre un CV rejeté automatiquement et une opportunité de carrière.

Au-delà des aspects techniques, ce projet nous a permis de comprendre l'importance du travail d'équipe et de l'empathie dans la conception de produits technologiques. Chaque ligne de code écrite avait un objectif : aider quelqu'un à trouver un emploi qui lui correspond.

Career Match AI n'est pas qu'une plateforme ; c'est un pont entre le talent et l'opportunité, une réalisation qui démontre notre capacité à transformer une idée en solution concrète.

*Ce projet représente le fruit de notre collaboration et de notre engagement.
Il témoigne de nos compétences techniques, de notre capacité à travailler en équipe
et de notre désir de contribuer à résoudre des problèmes réels de notre société.*

MIRGHANY Meryem

BELLAMINE Kenza

Bibliographie

- [1] Haut-Commissariat au Plan, *Note d'information sur le marché du travail au Maroc*, 2023.
- [2] Jobscan Research, *75% of Resumes Never Reach a Human Recruiter*, <https://www.jobscan.co/>, 2024.
- [3] Sebastián Ramírez, *FastAPI : Modern, Fast Web Framework for Building APIs*, <https://fastapi.tiangolo.com/>, 2024.
- [4] Google DeepMind, *Gemini : A Family of Highly Capable Multimodal Models*, Technical Report, 2024.
- [5] Sparck Jones, Karen, *A Statistical Interpretation of Term Specificity and its Application in Retrieval*, Journal of Documentation, 1972.
- [6] Devlin, Jacob et al., *BERT : Pre-training of Deep Bidirectional Transformers for Language Understanding*, NAACL, 2019.
- [7] Leslie Lamport, *LaTeX : A Document Preparation System*, Addison-Wesley, 1994.
- [8] Meta, *React : A JavaScript Library for Building User Interfaces*, <https://react.dev/>, 2024.
- [9] Tailwind Labs, *TailwindCSS : Utility-First CSS Framework*, <https://tailwindcss.com/>, 2024.
- [10] Jurafsky, Dan and Martin, James H., *Speech and Language Processing*, 3rd Edition, 2024.

Annexe A

Annexes

A.1 Annexe A : Questionnaire de satisfaction

Questionnaire utilisé pour l'évaluation utilisateur :

1. Sur une échelle de 1 à 5, comment évaluez-vous la facilité d'utilisation de Career Match AI ?
2. Les résultats de recherche d'emploi sont-ils pertinents ? (1-5)
3. L'analyse CV-offre vous a-t-elle été utile ? (1-5)
4. Quelle est la qualité du CV généré par l'optimiseur ATS ? (1-5)
5. Les conseils du Coach IA sont-ils pertinents ? (1-5)
6. Comment évaluez-vous le design et l'interface ? (1-5)
7. Recommanderiez-vous cet outil à un ami ? (Oui/Non)
8. Quelles fonctionnalités aimeriez-vous voir ajoutées ?
9. Commentaires libres et suggestions

A.2 Annexe B : Exemples de prompts Gemini

Prompt pour l'évaluation ATS :

You are an expert ATS (Applicant Tracking System) evaluator.
Analyze the following resume and provide a structured evaluation.

Evaluate across 14 categories:

1. Contact Information
2. Spelling & Grammar
3. Personal Pronoun Usage

```
[...]
```

For each category, provide:

- Positives: List of strengths (array of strings)
- Negatives: List of weaknesses (array of strings)

Also provide an overall ATS_Score from 0 to 100.

Return ONLY valid JSON in this exact format:

```
{
  "ATS_Score": 75,
  "Contact Information": {
    "Positives": [...],
    "Negatives": [...]
  },
  [...]
```

Resume text:

```
[RESUME_TEXT]
```

A.3 Annexe C : Structure de la base de données jobs

Colonnes du fichier `jobs_morocco.csv` :

Colonne	Type	Description
job_id	Integer	Identifiant unique
job_title	String	Titre du poste
category	String	Catégorie (IT, Finance, etc.)
description	Text	Description détaillée
required_skills	String	Compétences requises (CSV)
recommended_courses	String	Formations recommandées
avg_salary_mad	String	Fourchette salariale (MAD)
demand_level	String	Niveau de demande (High/Medium/Low)

A.4 Annexe D : Installation et configuration

Prérequis système :

- Python 3.10 ou supérieur
- Node.js 18 ou supérieur
- MiKTeX (pour la compilation LaTeX)
- 4GB RAM minimum
- 2GB espace disque

Installation backend :

```
# Cloner le repository
git clone https://github.com/[username]/career-match-ai.git
cd career-match-ai/backend

# Creer environnement virtuel
python -m venv venv
source venv/bin/activate # Linux/Mac
# ou
venv\Scripts\activate # Windows

# Installer dependances
pip install -r requirements.txt

# Configurer variables d'environnement
cp .env.example .env
# Editer .env et ajouter GEMINI_API_KEY

# Lancer le serveur
uvicorn main:app --reload --port 8000
```

Installation frontend :

```
cd ../frontend

# Installer dependances
npm install

# Lancer en dev
npm run dev

# Build pour production
```

```
npm run build
```

A.5 Annexe E : Captures d'écran

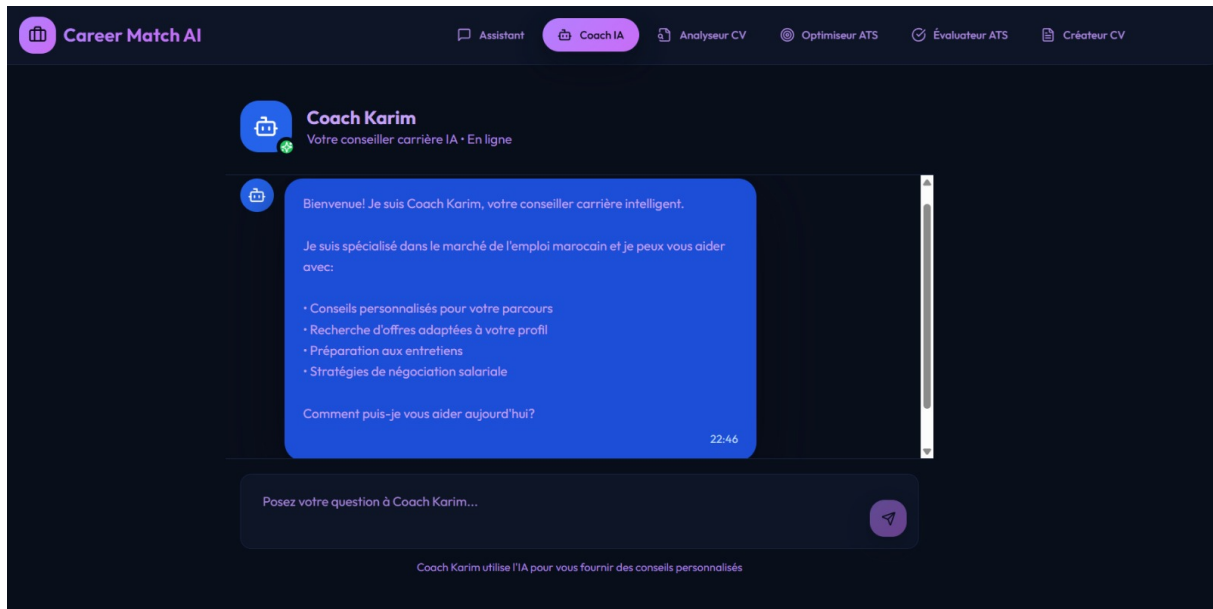


FIGURE A.1 – Interface de l'assistant de recherche

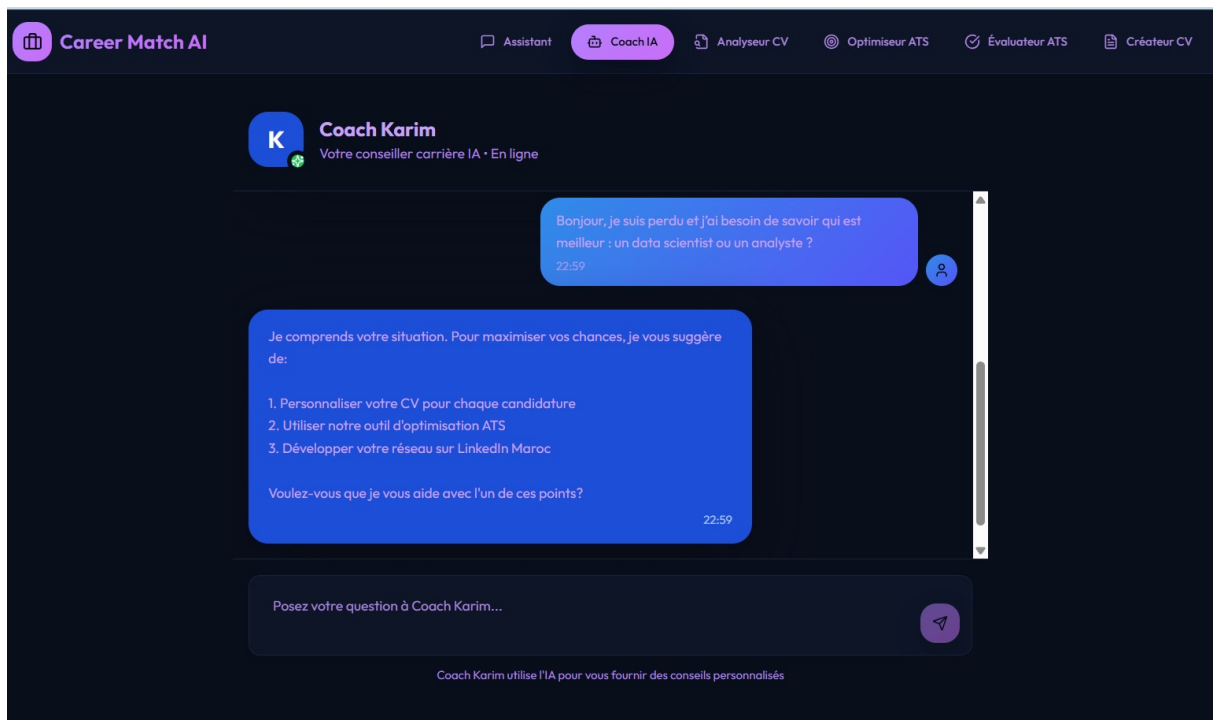


FIGURE A.2 – Interface de l'assistant de recherche (suite)



FIGURE A.3 – Interface de l'analyseur CV

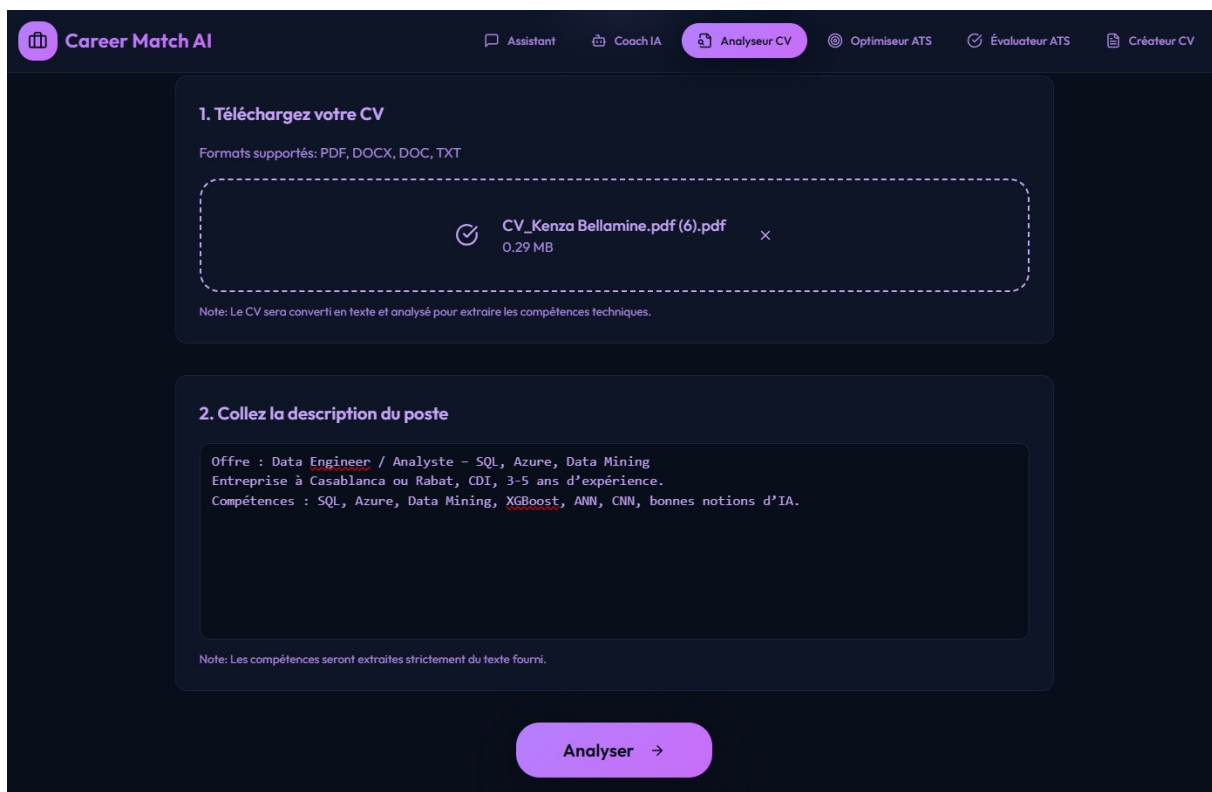


FIGURE A.4 – Interface de l'analyseur CV(suite)

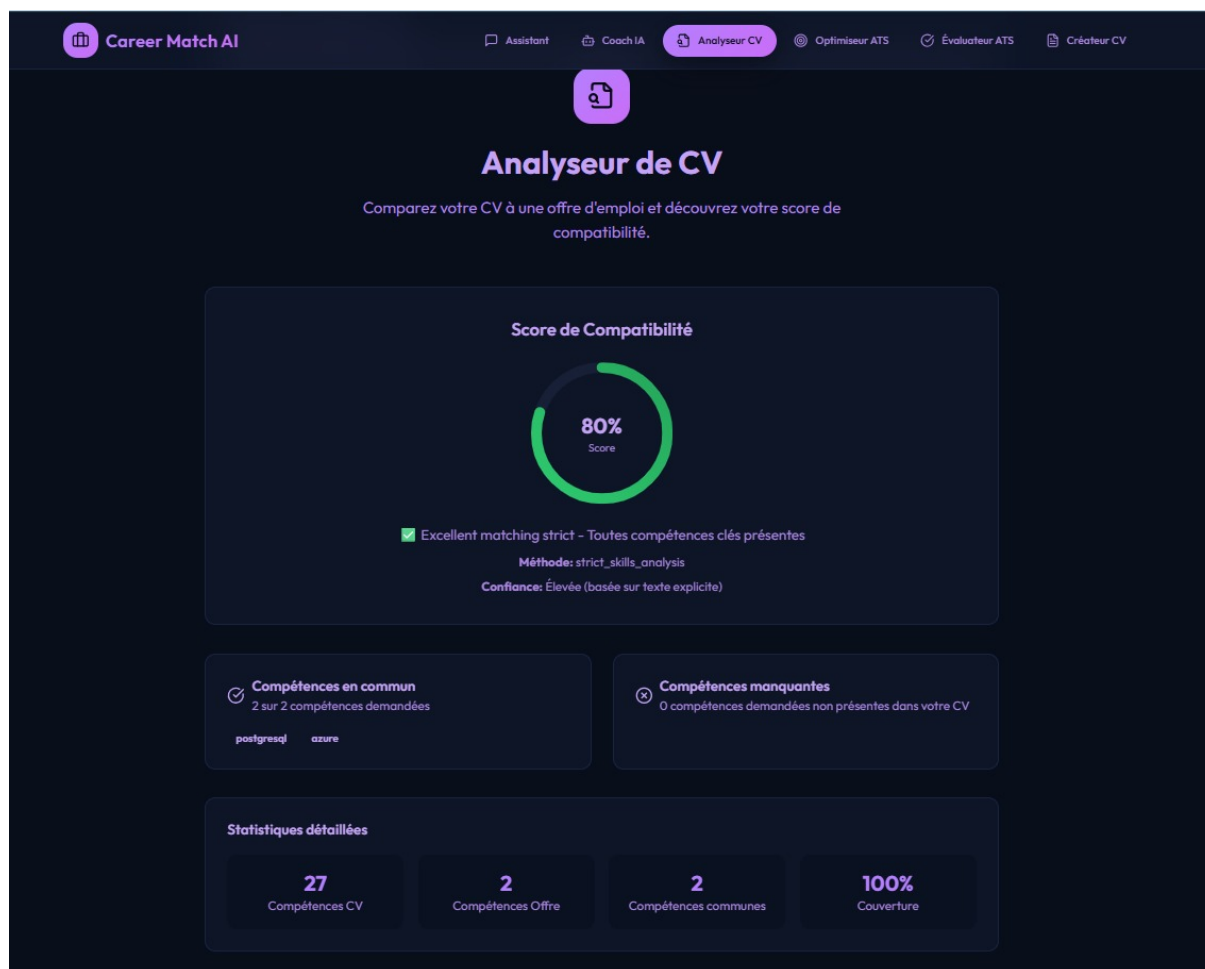


FIGURE A.5 – Interface de l'analyseur CV(suite)

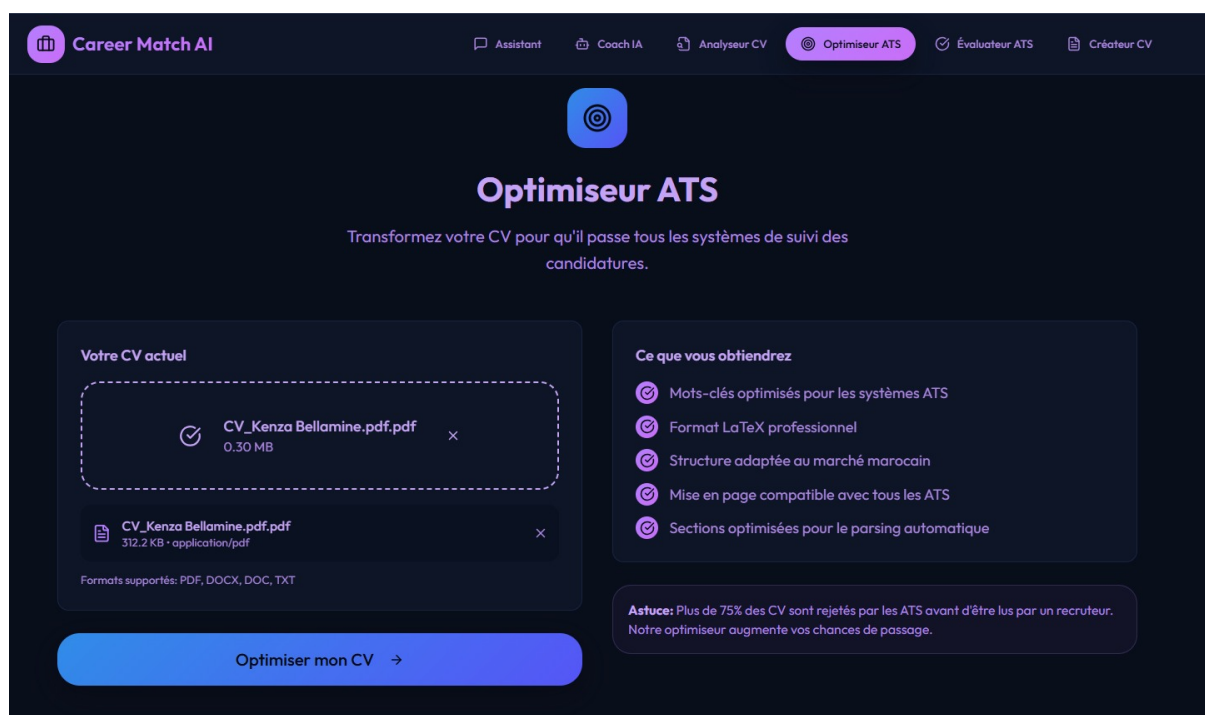


FIGURE A.6 – Interface de l'optimiseur ATS

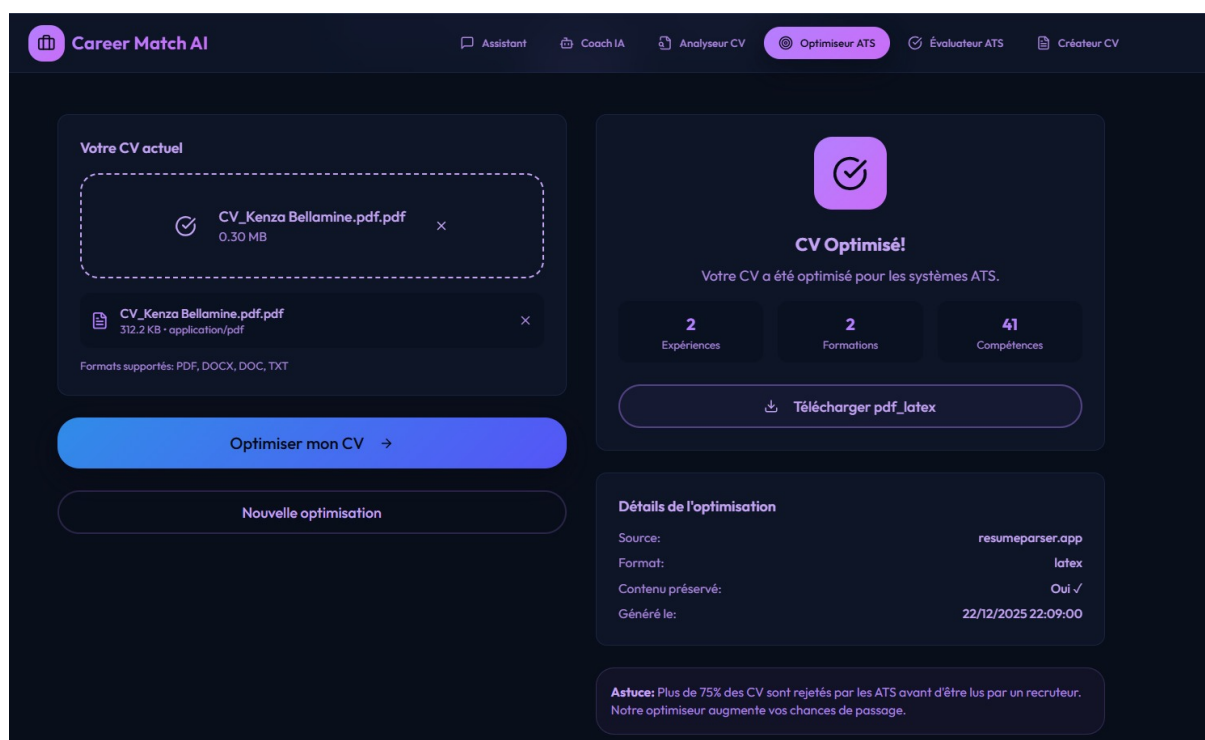


FIGURE A.7 – Interface de l'optimiseur ATS(suite)

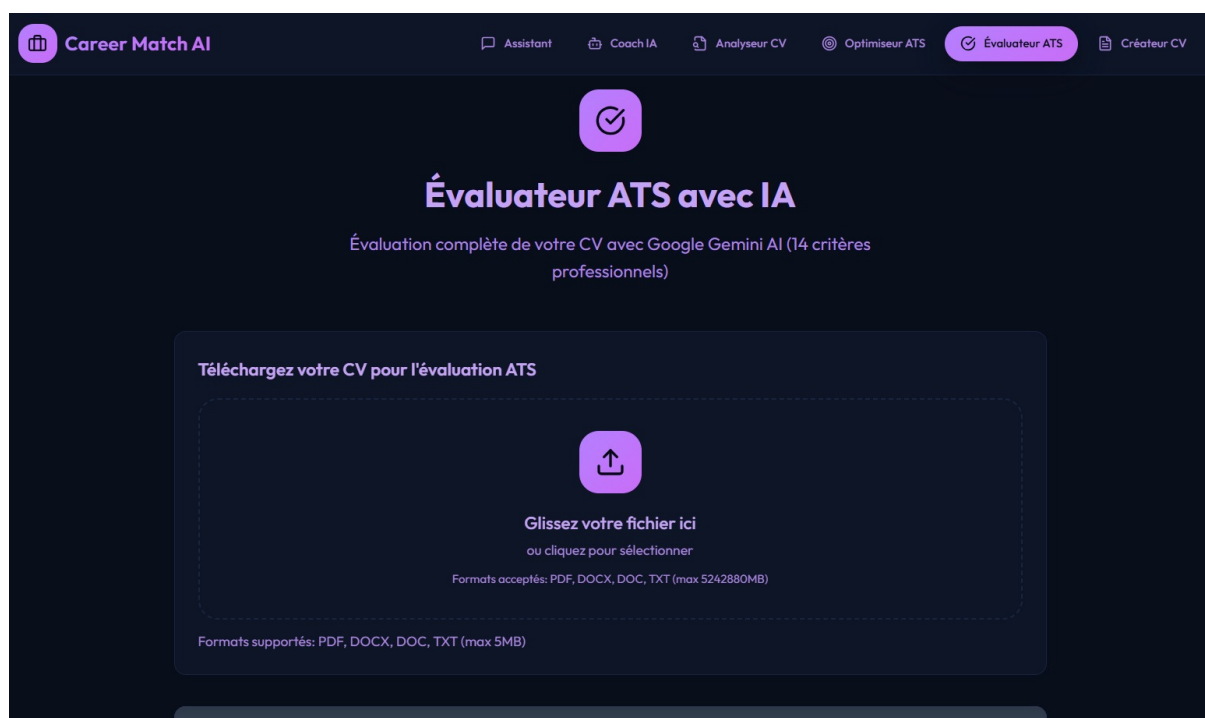


FIGURE A.8 – Interface de l'évaluateur ATS

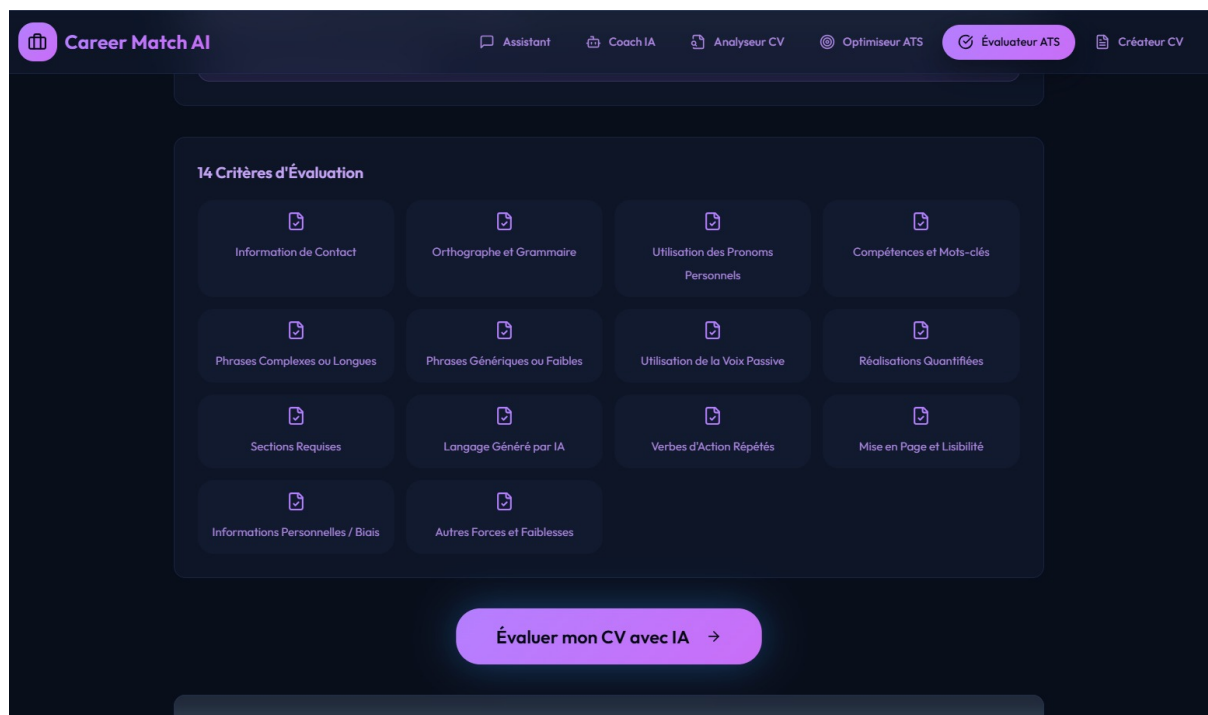


FIGURE A.9 – Interface de l'évaluateur ATS

RAPPORT D'ÉVALUATION ATS

Career Assistant - Évaluation Professionnelle

Informations Générales

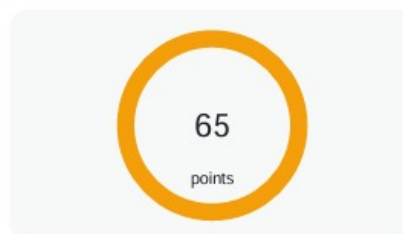
Date: 22/12/2025

Fichier analysé: cv_ats.pdf

Score Global ATS: 65/100

Modèle IA: gemini-1.5-pro

Longueur du CV: 2270 caractères

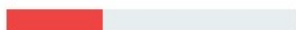


Résumé de l'Évaluation

Votre CV nécessite des améliorations significatives pour être bien lu par les ATS. Travaillez sur la structure, les mots-clés et l'élimination des erreurs courantes.

Détails par Catégorie

Information de Contact: 33%



Points forts:

- The candidate's name, 'Kenza Bellamine', is clearly stated at the beginning of the resume.

Points à améliorer:

- Crucial contact information such as email address, phone number, LinkedIn profile URL, and city/country are entirely missing, making it impossible for a recruiter to reach the candidate. This is a critical omission for ATS processing and recruiter contact.
- The resume lacks a dedicated and clearly labeled 'Contact Information' section, which hinders ATS parsing and quick identification of essential details.

Orthographe et Grammaire: 50%



Points forts:

FIGURE A.10 – Extrait rapport d'évaluation ATS

The screenshot displays the 'Créateur de CV Professionnel' (Professional CV Creator) interface. At the top, a navigation bar includes the 'Career Match AI' logo and several service icons: Assistant, Coach IA, Analyseur CV, Optimiseur ATS, Évaluateur ATS, and Créateur CV. The main heading is 'Créateur de CV Professionnel', followed by the subtitle 'Créez un CV professionnel optimisé pour le marché marocain. Toutes les informations optionnelles.' Below this is a horizontal menu with icons for Informations, Expérience, Formation, Compétences, Langues, and Aperçu. The 'Informations' section is active, showing a form with the following fields: 'Nom complet *' (filled with 'Meryem Mirghany'), 'Email *' (filled with 'meryem.mirghany03@gmail.com'), 'Téléphone (optionnel)' (filled with '060856974226'), 'Ville (optionnel)' (filled with 'Casablanca, Maroc'), 'LinkedIn (optionnel)' (filled with 'linkedin.com/in/votreprofil'), and 'Résumé professionnel (optionnel)' (with a placeholder text 'Décrivez votre profil en quelques phrases...'). Navigation buttons 'Précédent' and 'Suivant' are located at the bottom.

Créateur de CV Professionnel

Créez un CV professionnel optimisé pour le marché marocain. Toutes les informations optionnelles.

Informations Expérience Formation Compétences Langues Aperçu

Informations

Nom complet * Email *

Meryem Mirghany meryem.mirghany03@gmail.com

Téléphone (optionnel) Ville (optionnel)

060856974226 Casablanca, Maroc

LinkedIn (optionnel)

linkedin.com/in/votreprofil

Résumé professionnel (optionnel)

Décrivez votre profil en quelques phrases...

Précédent Suivant

FIGURE A.11 – Createur CV ATS