



UNIVERSITÉ DE FIANARANTSOA

ÉCOLE NATIONALE D'INFORMATIQUE

RAPPORT DE STAGE EN DEUXIEME ANNEE DE LICENCE PROFESSIONNELLE

Mention: Informatique

Parcours: Génie Logiciel et Base de Données

Intitulé:

**Développement et implémentation d'une plateforme web de
classification de CV basée sur l'analyse des critères du poste.**

Présenté le: 18 décembre 2023

Par: ANDRIANTSILAVINA Tsiferana Heritsilavo

N°2632

Membre de Jury:

Rapporteurs: BERTIN Andry (Encadreur pédagogique)

RAFANOMANANA Emmanuel Josoa (Encadreur professionnel)

Année-Universitaire:2022-2023

CURICULUM VITAE



ANDRIANTSILAVINA
TSIFERANA HERITSILAVO

Developpeur Fullstack

CONTACT

Tel. 0342631778
Lot 331A/3704, Fianarantsoa
heritsilavo4835@gmail.com

EDUCATION

“ENI Fianarantsoa”
-L1 : 2021-2022
-L2 : 2022-2023

DIPLOME

Baccalauréat 2020-2021

LANGUE

Malagasy
Francais
Anglais

Competences en informatique

Developpement web

- NextJs
- ReactJs
- VueJs
- HTML/CSS/JS
- NestJS
- NodeJS
- PHP
- MySQL
- PostgreSQL
- MongoDB

Programmation

- JS
- C++
- C
- C#
- Python

Méthodologie de developpement

- Méthode Merise
- Méthode Agile(Scrum)

Gestion de version et outil de collaboration

- Github
- Gitlab

Outil de gestion de projet

- Docker
- ClickUp

Système d'exploitation

- Windows
- Linux

Loisirs

- Basketball
- Guitare

REMERCIEMENTS

Nous exprimons par ces quelques lignes nos remerciements ainsi que nos gratitude envers tous ceux, en qui par leur présence, leur soutien, leur disponibilité et leurs conseils nous avons trouvé courage afin d'accomplir ce projet.

En particulier,

- Monsieur Thomas Mahatody, Directeur de l'École Nationale d'Informatique qui nous a donné l'opportunité de réaliser ce stage;
- Monsieur BERTIN Andry, notre tuteur de stage, qui nous a encadrés et s'est toujours montré à l'écoute et très disponible, tout au long de la réalisation de ce stage. Son aide et le temps qu'il a bien voulu nous consacrer, nous ont été précieux et indispensables à la compréhension de notre mission et ont très largement contribué au bon déroulement de notre stage.
- Monsieur **Fetra**, pour avoir accepté de me prendre en tant que stagiaire et aussi à **RAFANOMANANA Emmanuel Josoa** d'être notre encadreur au sein de la société Webdevin.

Nous ne pourrions achever ce projet sans exprimer également nos gratitude, à tous les enseignants de la Formation Hybride de l'École Nationale d'Informatique, pour leur dévouement et leur assistance tout au long de l'année. Enfin nous exprimons notre gratitude à ceux qui nous ont rencontrés lors de ce stage, qui nous ont acceptés de répondre à nos questions avec gentillesse, qui nous ont conseillés à la rédaction du rapport de stage.

LISTE DES FIGURES

Figure 1 : Organigramme actuel de l'École.....	5
Figure 2 : Architecture des études correspondant au système LMD.....	8
Figure 3 : Organigramme de l'entreprise.....	17
Figure 4 : Diagramme de flux de données.....	31
Figure 5 : Modele conceptuel de données.....	32
Figure 6 : Modele conceptuel de traitement.....	33
Figure 7 : Design Figma.....	39
Figure 8 : candidat.controller.ts.....	40
Figure 9 : candidat.service.ts.....	41
Figure 10 : candidat.module.ts.....	41
Figure 11 : candidat.dto.ts	42
Figure 12 : candidat.schemas.ts.....	43
Figure 13 : Gérer l' upload de fichier.....	44
Figure 14 : mail.module.ts.....	45
Figure 15 : Modèle du mail a envoyer 'templates.hbs'.....	46
Figure 16 : fichier de configuration de nestjs.....	46
Figure 17 : mail.service.ts.....	47
Figure 18 : un extrait de test unitaire.....	47
Figure 19 : extrait de test d'integration.....	48
Figure 20 : Lancement des test.....	49
Figure 21 : test réussi.....	49
Figure 22 : gestion de route de l'application.....	51
Figure 23 : fichier ./app/postuler/form/[id_poste].page.jsx.....	52
Figure 24 : le fichier ./app/postuler/form/layout.jsx.....	52
Figure 25 : Déclaration des fonction pour faire les requêtes vers les api backend...53	
Figure 26 : Exécution des requêtes vers la base de données.....	54
Figure 27 : Déclaration du composant dropzone.....	55
Figure 28 : test unitaire sur le composant dropzone.....	55
Figure 29 : formulaire de recrutement.....	56
Figure 30 : interface pour remplir la description de l'offre.....	57
Figure 31 : Publier un offre d'emploi.....	57
Figure 32 : formulaire de candidature.....	58
Figure 33 : Interface pour enregistres la vidéo:.....	58
Figure 34 : Modèle d' email accusée de réception.....	59

LISTE DES TABLEAUX

Table 1 : Comparaison des formation theorique et formation pratique.....	7
Table 2 : Formation en Licence vs Master.....	9
Table 3 : Débauche professionnelle.....	15
Table 4 : Conception avant projet.....	22
Table 5 : Dictionnaire de données.....	27

LISTE ABRÉVIATIONS

- ✓ ENI : École Nationale d'Informatique
- ✓ CRUD : Create Read Update Delete
- ✓ SQL : Structured Query Language
- ✓ JS : Javascript
- ✓ CV: Curriculum Vitae
- ✓ DFD: Diagramme de flux de données
- ✓ MCD : Modèle conceptuel de données
- ✓ MCT : Modèle conceptuel de traitement
- ✓ RH : Relations Humaines
- ✓ API : Application Programming Interface
- ✓ IA : Intelligence Artificiel
- ✓ HTML: HyperText Markup Language

SOMMAIRE

CURICULUM VITAE.....	I
REMERCIEMENTS.....	II
LISTE DES FIGURES.....	III
LISTE DES TABLEAUX.....	IV
LISTE ABRÉVIATIONS.....	V
SOMMAIRE.....	VI
INTRODUCTION GENERALE.....	1
PARTIE I. PRESENTATION.....	2
chap 1: L'ENI:.....	2
chap 2: Présentation de L' Entreprise.....	17
chap 3: Description du projet.....	18
PARTIE II. ANALYSE ET CONCEPTION.....	21
chap 4: Analyse Préalable.....	21
chap 5: Analyse conceptuel.....	27
chap 6: Conception détaillée.....	31
PARTIE III. REALISATION.....	34
chap 7: Mise en place de l'environnement de développement.....	34
chap 8: Développement de l'application.....	38
CONCLUSION.....	60
BIBLIOGRAPHIE.....	VII
WEBOGRAPHIE.....	VIII
GLOSSAIRE.....	IX
ANEXES.....	X
TABLE DE MATIERE.....	XI
RÉSUMÉ.....	VII
ABSTRACT.....	VIII

INTRODUCTION GENERALE

Dans le cadre de ma deuxième année de Licence Professionnelle en Informatique, j'ai effectué un stage d'une durée de trois mois, au sein de la société WebDevin, une boîte informatique à Antananarivo. Cette société m'a accepté du 28 août 2023 au 28 novembre 2023 afin de m'aider à développer une plateforme web de classification de CV basée sur l'analyse des critères du poste.

WebDevin est une entreprise spécialisée dans le développement de logiciels informatiques. Elle propose une large gamme de services, notamment la conception, le développement, la maintenance de sites web, d'applications web et mobiles. L'entreprise emploie une vingtaine de personnes, dont une dizaine de développeurs.

La plateforme web de classification de CV est destinée aux entreprises qui souhaitent recruter. Elle permet aux recruteurs de créer des offres d'emploi, de publier ces offres sur les réseaux sociaux et de recevoir les candidatures des candidats. La plateforme utilise un algorithme d'intelligence artificielle pour classer les candidatures en fonction des critères du poste.

Ce rapport est divisé en trois parties principales : présentation, analyse et conception, puis réalisation de l'application.

PARTIE I. PRESENTATION

chap 1: L'ENI:

1. 1) Information d'ordre général

L'École Nationale d'Informatique, en abrégé ENI, est un établissement d'enseignement supérieur rattaché académiquement et administrativement à l'Université de Fianarantsoa.

Le siège de l'École se trouve à Tanambao-Antaninarenina à Fianarantsoa.

L'adresse pour la prise de contact avec l'École est la suivante : École Nationale d'Informatique (ENI) Tanambao, Fianarantsoa. Le numéro de sa boîte postale est 1487 avec le code postal 301. Téléphone : 020 75 508 01. Son adresse électronique est la suivante : *eni@univ-fianar.mg*. Site Web : www.univ-fianar.mg/eni

1. 2) Missions et historique

L'ENI se positionne sur l'échiquier socio-éducatif malgache comme étant le plus puissant secteur de diffusion et de vulgarisation des connaissances et des technologies informatiques.

Cette École Supérieure peut être considérée aujourd'hui comme la vitrine et la pépinière des élites informaticiennes du pays.

L'École s'est constituée de façon progressive au sein du Centre Universitaire Régional (CUR) de Fianarantsoa.

De façon formelle, l'ENI était constituée et créée au sein du (CUR) par le décret N° 83185 du 24 Mai 1983, comme étant le seul établissement Universitaire Professionnalisé au niveau national, destiné à former des techniciens et des Ingénieurs de haut niveau, aptes à répondre aux besoins et exigences

d'Informatisation des entreprises, des sociétés et des organes implantés à Madagascar.

L'ENI a pour conséquent pour mission de former des spécialistes informaticiens compétents et opérationnels de différents niveaux notamment :

- En fournissant à des étudiants des connaissances de base en informatique ;
- En leur transmettant le savoir-faire requis, à travers la professionnalisation des formations dispensées et en essayant une meilleure adéquation des formations par rapport aux besoins évolutifs des sociétés et des entreprises.
- En initiant les étudiants aux activités de recherche dans les différents domaines des

Technologies de l'information et de la communication (TIC).

L'implantation de cette École Supérieure de technologie de pointe dans un pays en développement et dans une Province (ou Faritany) à tissu économique et industriel faiblement développé ne l'a pourtant pas défavorisée, ni empêchée de former des spécialistes informaticiens de bon niveau, qui sont recherchés par les entreprises, les sociétés et les organismes publics et privés sur le marché de l'emploi. La filière de formation d'Analystes Programmeurs a été mise en place à l'École en 1983, et a été gelée par la suite en 1996, tandis que la filière de formation d'ingénieurs a été ouverte à l'École en 1986.

Dans le cadre du Programme de renforcement en l'Enseignement Supérieur (PRESUP), la filière de formation des Techniciens Supérieurs en Maintenance des Systèmes des informatiques a été mise en place en 1986 grâce à l'appui matériel et financier de la Mission Française de coopération auprès de l'Ambassade de France à Madagascar.

Une formation pour l'obtention de la certification CCNA et / ou NETWORK + appelée « CISCO Networking Academy » a été créée à l'École en 2002-2003 grâce au partenariat avec CISCO SYSTEM et l'École Supérieure Polytechnique d'Antananarivo (ESPA). Cependant, cette formation n'avait pas duré longtemps.

Une formation de troisième cycle a été ouverte à l'École a été ouverte à l'École depuis l'année 2003 – 2004 grâce à la coopération académique et scientifique entre l'Université de

Fianarantsoa pour le compte de l'ENI et l'Université Paul Sabatier de Toulouse (UPST). Cette filière avait pour objectif de former certains étudiants à la recherche dans les différents domaines de l'Informatique, et notamment pour préparer la relève des Enseignants-Chercheurs qui étaient en poste.

Pendant l'année 2007-2008, la formation en vue de l'obtention du diplôme de
Licence
Professionnelle en Informatique a été mise en place à l'ENI avec les deux options suivantes de formation :

- Génie Logiciel et base de Données.
- Administration des Système et réseaux.

La mise en place à l'École de ces deux options de formation devait répondre au besoin de basculement vers le système Licence – Master – Doctorat (LMD).

Mais la filière de formation des Techniciens Supérieurs en Maintenance des Systèmes Informatiques a été gelée en 2009.

En vue de surmonter les difficultés de limitation de l'effectif des étudiants accueillis à l'École, notamment à cause du manque d'infrastructures, un système de « Formation Hybride » a été mise en place à partir de l'année 2010. Il s'agit en effet d'un système de formation semi présentielle et à distance avec l'utilisation de la visioconférence pour la formation à distance. Le système de formation hybride a été ainsi créé à Fianarantsoa ainsi qu'Université de Toliara.

1. 3) Organigramme institutionnel de l'ENI

Cet organigramme de l'École est inspiré des dispositions du décret N° 83-185 du 23 Mai 1983.

L'ENI est administrée par un conseil d'École, et dirigée par un directeur nommé par un décret adopté en conseil des Ministres.

Le Collège des enseignants regroupant tous les enseignants-chercheurs de l'École est chargé de résoudre les problèmes liés à l'organisation pédagogique des enseignements ainsi qu'à l'élaboration des emplois du temps.

Le Conseil Scientifique propose les orientations pédagogiques et scientifiques de l'établissement, en tenant compte notamment de l'évolution du marché de travail et de l'adéquation des formations dispensées par rapport aux besoins des entreprises.

Trois départements de formation caractérisent l'organigramme :

- Le département de formation théorique à l'intérieur de l'École ;
- Le département de formation pratique pour la coordination et la supervision des stages en entreprise et des voyages d'études ;

Le département de formation doctorale pour l'organisation de la formation de 3ème cycle.

La figure 1 présente l'organigramme actuel de l'École.

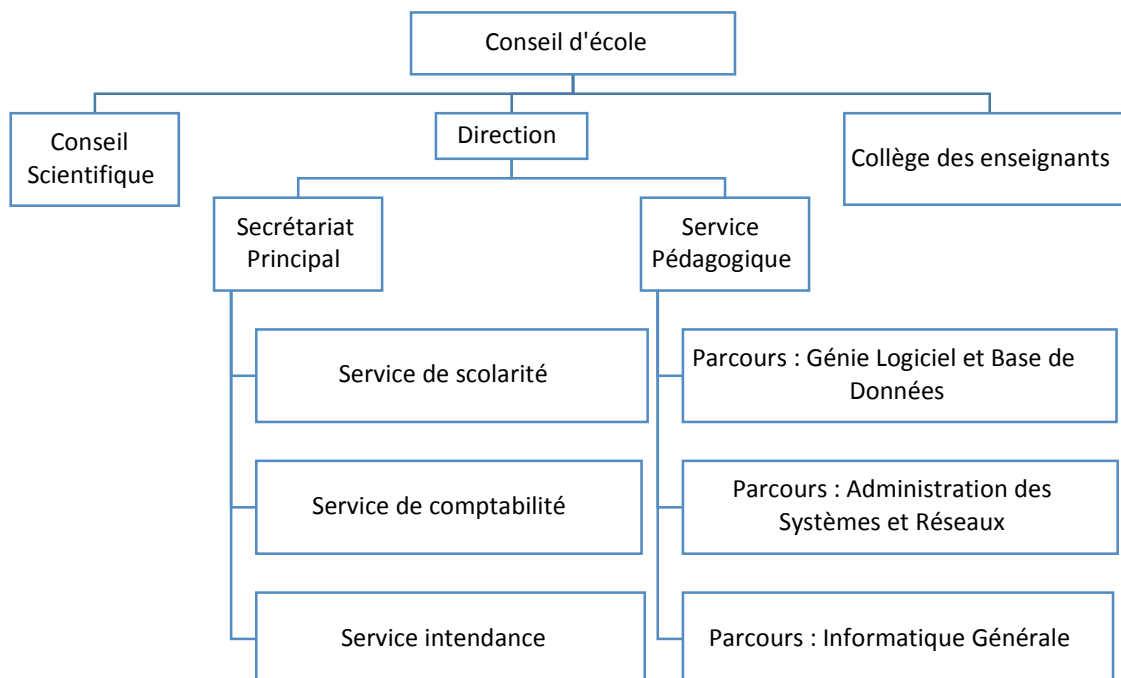


Figure 1 : Organigramme actuel de l'École

Sur cet organigramme, l'École placée sous la tutelle académique et administrative de l'Université de Fianarantsoa, et dirigée par un Directeur élu par

les Enseignants – Chercheurs permanents de l'Établissements et nommé par un décret pris en Conseil des ministres pour un mandat de 3 ans.

Le Conseil de l'École est l'organe délibérant de l'École.

Le Collège des Enseignants propose et coordonne les programmes d'activités pédagogiques.

Le Conseil scientifique coordonne les programmes de recherche à mettre en œuvre à l'École.

Le Secrétariat principal coordonne les activités des services administratifs (Scolarité, Comptabilité, et Intendance).

Conformément aux textes en vigueur régissant les Établissements malgaches d'Enseignement Supérieur, qui sont barrés sur le système LMD, les Départements de Formation pédagogique ont été ainsi remplacés par des Mentions et des parcours. Et les chefs des Départements ont été ainsi remplacés par des responsables des mentions et les responsables des parcours.

Un administrateur des Réseaux et Systèmes gère le système d'information de l'École et celui de l'Université.

1. 4) Domaine de spécialisation

Les activités de formation et de recherche organisées à l'ENI portent sur les domaines suivants :

- Génie logiciel et Base de Données ;
- Administration des Systèmes et Réseaux ;
- Informatique Générale
- Modélisation informatique et mathématique des Systèmes complexes.

D'une manière plus générale, les programmes des formations sont basés sur l'informatique de gestion et sur l'informatique des Systèmes et Réseaux. Et les

modules de formation intègrent aussi bien des éléments d'Informatique fondamentale que des éléments d'Informatique appliquée.

Table 1 : Comparaison des formation theorique et formation pratique

Formation théorique	Formation pratique
<ul style="list-style-type: none">- Enseignement théorique- Travaux dirigés- Travaux pratiques	<ul style="list-style-type: none">- Etude de cas- Travaux de réalisation- Projets / Projets tutorés- Voyage d'études- Stages

1. 5) Architecture des formations pédagogiques

Le recrutement des étudiants à l'ENI se fait uniquement par voie de concours d'envergure nationale en première année.

Les offres de formation organisées à l'École ont été validées par la Commission Nationale d'Habilitation (CNH) auprès du Ministères de l'Enseignement Supérieur et de la Recherche Scientifique selon les dispositions de l'Arrêté N°31.174/2012-MENS en date du 05 Décembre 2012.

Au sein de l'ENI, il existe une seule mention (INFORMATIQUE) et trois parcours :

- o Génie logiciel et Base de Données ;
- o Administration des Systèmes et Réseaux ;
- o Informatique Générale

L'architecture des études à trois niveaux conformément au système Licence-Master-Doctorat (LMD) permet les comparaisons et les équivalences académiques des diplômes au niveau international.

- L = Licence (Bac + 3) = L1, L2, L3 = 6 semestres S1 à S6
- M = Master (Bac + 5) = M1, M2 = 4 semestres S7 à S10

Le diplôme de licence est obtenu en 3 années des études après Baccalauréat. Et le diplôme de Master est obtenu en 2 ans après obtenu du diplôme de LICENCE. Le MASTER PROFESSIONNEL est un diplôme destiné à la recherche emploi au terme des études.

Le MASTER RECHERCHE est un diplôme qui remplace l'ancien Diplôme d' Études Approfondies (DEA), et qui permet de s'inscrire directement dans une École Doctorale. au terme des études.

- D = Doctorat (Bac +8)

Le Doctorat est un diplôme qu'on peut obtenir en 3 ans après l'obtention du diplôme de MASTER RECHERCHE.



Figure 2 : Architecture des études correspondant au système LMD

BTS : Brevet de Technicien Supérieur

DUT : Diplôme Universitaire de Technicien

La licence peut avoir une vocation générale ou professionnelle.

Le master peut avoir une vocation professionnelle ou de recherche.

Table 2 : Formation en Licence vs Master

	FORMATION EN	
	LICENCE PROFESSIONNELLE ET HYBRIDE	MASTER
Condition d'admission	Par voie de concours Formation Professionnelle : 100 candidats Formation hybride : 150 candidats	
Condition d'accès	Bac de série C, D ou Technique	Être titulaire de licence professionnelle
Durée de formation	3 années	2 années
Diplôme à délivrer	Diplôme de Licence Professionnelle en Informatique	Diplôme de Master Professionnel Diplôme de Master Recherche

L'accès en première année de MASTER se fait automatiquement pour les étudiants de l'École qui ont obtenu le diplôme de Licence Professionnelle.

Le Master Recherche permet à son titulaire de poursuivre directement des études en doctorat et de s'inscrire directement dans une École Doctorale.

Les Écoles Doctorales jouissent d'une autonomie de gestion par rapport aux Établissements de formation universitaire.

Il convient de signaler que par arrêté ministériel N° 21.626/2012 – MESupRES publié le 9 Août 2012 par la Commission National d'habilitation (CNH), l'École Doctorale « Modélisation – Informatique » a été habilitée pour l'Université de Fianarantsoa.

Depuis l'année universitaire 2010-2011, l'ENI s'est mise à organiser des formations hybrides en informatique dans les différentes régions (Fianarantsoa, Toliara) en raison de l'insuffisance de la capacité d'accueil des infrastructures logistiques. En effet, le système de formation hybride semi - présentielle utilise la visioconférence pour la formation à distance.

Bien qu'il n'existe pas encore au niveau international de reconnaissance écrite et formelle des diplômes délivrés par l'ENI, les étudiants diplômés de l'École sont plutôt bien accueillis dans les instituts universitaires étrangères (CANADA, Suisse, France...)

1. 6) Relations de l'ENI avec les entreprises et les organismes

Les stages effectués chaque année par les étudiants mettent l'École en rapport permanent avec plus de 300 entreprises et organismes publics, semi-publics et privés, nationaux et internationaux.

L' École dispose ainsi d'un réseau d'entreprises, de sociétés et d'organismes publics et privés qui sont des partenaires par l'accueil en stage de ses étudiants, et éventuellement pour le recrutement après l'obtention des diplômes par ces derniers.

Les compétences que l'École cherche à développer chez ses étudiants sont l'adaptabilité, le sens de la responsabilité, du travail en équipe, le goût de l'expérimentation et l'innovation.

En effet, la vocation de l'ENI est de former des techniciens supérieurs de niveau LICENCE et des ingénieurs de type généraliste de niveau MASTER avec des qualités scientifiques, techniques et humaines reconnues, capables d'évoluer professionnellement dans des secteurs d'activité variés intégrant l'informatique.

Les stages en milieu professionnel permettent de favoriser une meilleure adéquation entre les formations à l'École et les besoins évolutifs du marché de l'emploi.

Les principaux débouchés professionnels des diplômés de l'École concernent les domaines suivants :

- ✓ L'informatique de gestion d'entreprise
 - ✓ Les technologies de l'information et de la communication (TIC)
 - ✓ La sécurité informatique des réseaux
 - ✓ L'administration des réseaux et des systèmes
 - ✓ Les services bancaires et financiers, notamment le Mobile Banking
 - ✓ Les télécommunications et la téléphonie mobile
 - ✓ Les Big Data
 - ✓ Le commerce, la vente et l'achat, le Marketing
 - ✓ L'ingénierie informatique appliquée
 - ✓ L'écologie et le développement durable
- Parmi les sociétés, entreprises et organismes partenaires de l'École, on peut citer : ACCENTURE Mauritius, Air Madagascar, Ambre Associates, Airtel, Agence Universitaire de la Francophonie (AUF), B2B, Banque Centrale, BFG-SG, BIANCO, BLUELINE, CNaPS, Bureau National de Gestion des Risques et des Catastrophes (BNGRC), CEDII-Fianarantsoa,

Data Consulting, Central Test, Centre National Antiacridien, CNRE, CHU, CNRIT, COLAS, Direction Générale des Douanes, DLC, DTS/Moov, FID, FTM, GNOSYS, IBONIA, INGENOSIA, INSTAT, IOGA, JIRAMA, JOUVE, MADADEV, MAEP, MEF, MEN, MESupRES, MFB, MIC, MNINTER, Min des

postes/Télécommunications et du Développement Numérique, NEOV MAD, Ny Havana, Madagascar National Parks, OMNITEC, ORANGE, OTME, PRACCESS, QMM Fort-Dauphin, SMMC, SNEDADRS

Antsirabe, Sénat, Société d'Exploitation du Port de Toamasina (SEPT), SOFTWELL, Strategy Consulting, TELMA, VIVETEC, Société LAZAN'I BETSILEO, WWF ...

L'organisation de stage en entreprise continue non seulement à renforcer la professionnalisation des formations dispensées, mais elle continue surtout à accroître de façon exceptionnelle les opportunités d'embauche pour les diplômés de l'École.

1. 7) Partenariat au niveau international

Entre 1996 et 1999, l'ENI avait bénéficié de l'assistance technique et financière de la Mission Française de Coopération et d'action culturelle dans le cadre du Programme de Renforcement de l'Enseignement Supérieur (PRESUP) consacré à l'École a notamment porté sur :

- Une dotation en logiciels, micro-ordinateurs, équipements de laboratoire de maintenance et de matériels didactiques
- La réactualisation des programmes de formation assortie du renouvellement du fonds de la bibliothèque
- L'appui à la formation des formateurs
- L'affectation à l'École d'Assistants techniques français

De 2000 à 2004, l'ENI avait fait partie des membres du bureau de la Conférence Internationale des Écoles de formation d'Ingénieurs et Techniciens d'Expression Française (CITEF).

Les Enseignants-Chercheurs de l'École participent régulièrement aux activités organisées dans le cadre du Colloque Africain sur la Recherche en Informatique (CARI).

L'ENI avait également signé un accord de coopération interuniversitaire avec l'Institut de Recherche en Mathématiques et Informatique Appliquées (IREMIA) de

l'Université de la Réunion, l'Université de Rennes 1, l'INSA de Rennes, l'Institut National Polytechnique de Grenoble (INPG).

A partir du mois de Juillet 2001, l'ENI avait abrité le Centre de Réseau Opérationnel (Network Operating Center) du point d'accès à Internet de l'École ainsi que de l'Université de Fianarantsoa. Grâce à ce projet américain qui a été financé par l'USAID Madagascar, l'ENI de l'Université de Fianarantsoa avait été dotées d'une ligne spécialisée d'accès permanent au réseau Internet.

L'ENI avait de même noué des relations de coopération avec l'Institut de Recherche pour le Développement (IRD).

L'objet du projet de coopération avait porté sur la modélisation environnementale du Corridor forestier de Fandriana jusqu'à Vondrozo (COFAV). Dans ce cadre, un atelier scientifique international avait été organisé à l'ENI en Septembre 2008. Cet atelier scientifique avait eu pour thème de modélisation des paysages.

Et dans le cadre du programme scientifique PARRUR, l'IRD avait financé depuis 2010 le projet intitulé « Forêts, Parcs et Pauvreté dans le Sud de Madagascar (FPPSM). Des étudiants en DEA et des Doctorants issus de l'ENI avaient participé à ce Programme.

Par ailleurs, depuis toujours la même année 2010, l'ENI de Fianarantsoa avait été sélectionnée pour faire partie des organismes partenaires de l'Université de Savoie dans le cadre du projet TICEVAL relatif à la certification des compétences en TIC ;

Le projet TICEVAL avait été financé par le Fonds Francophone des Inforoutes pour la période allant de 2010 à 2012, et il avait eu pour objectif de généraliser la certification des compétences en Informatique et Internet du type C2i2e et C2imi.

Dans le cadre du projet TICEVAL, une convention de coopération avec l'Université de Savoie avait été signée par les deux parties concernées. La mise en œuvre de la Convention de Coopération avait permis d'envoyer des étudiants de l'ENI à Chambéry pour poursuivre des études supérieures en Informatique.

Enfin et non des moindres, l'ENI avait signé en Septembre 2009 un protocole de collaboration scientifique avec l'ESIROI – STIM de l'Université de la Réunion.

Comme l'ENI constitue une pépinière incubatrice de technologie de pointe, d'emplois et d'entreprises, elle peut très bien servir d'instrument efficace pour renforcer la croissance économique du pays, et pour lutter contre la Pauvreté.

De même que le statut de l'École devrait permettre de renforcer la position concurrentielle de la Grande Ile sur l'orbite de la modélisation grâce au développement des nouvelles technologies.

1. 8) Débouchés professionnels avec des diplômés

Le chômage des jeunes diplômés universitaires fait partie des maux qui gangrènent

Madagascar. L'environnement socio-politique du pays depuis 2008 jusqu'à ce jour a fait que le chômage des diplômés est devenu massif par rapport aux établissements de formation supérieure existants.

Cependant, les formations proposées par l'École permettent aux diplômés d'être immédiatement opérationnels sur le marché du travail avec la connaissance d'un métier complet lié à l'informatique aux TIC.

L'École apporte à ses étudiants un savoir-faire et un savoir-être qui les accompagnent tout au long de leur vie professionnelle. Elle a une vocation professionnalisante.

Les diplômés en LICENCE et en MASTER issus de l'ENI peuvent faire carrière dans différents secteurs.

L'École bénéficie aujourd'hui de 34 années d'expériences pédagogiques et de reconnaissance auprès des sociétés, des entreprises et des organismes. C'est une École Supérieure de référence en matière informatique.

Par conséquent, en raison de fait que l'équipe pédagogique de l'École est expérimentée, les enseignants-chercheurs et les autres formateurs de l'École sont dotés d'une grande expérience dans l'enseignement et dans le milieu professionnel.

L'École est fière de collaborer de façon régulière avec un nombre croissant d'entreprises, de sociétés et d'organismes publics et privés à travers les stages des

étudiants. Les formations dispensées à l'École sont ainsi orientées vers le besoin et les attentes des entreprises et des sociétés.

L'École fournit à ses étudiants de niveau LICENCE et MASTER des compétences professionnelles et métiers indispensables pour les intégrer sur le marché du travail.

L'École s'efforce de proposer à ses étudiants une double compétence à la fois technologique et managériale combinant l'informatique de gestion ainsi que l'administration des réseaux et systèmes.

D'une manière générale, les diplômés de l'ENI n'éprouvent pas de difficultés particulières à être recrutés au terme de leurs études. Cependant, l'ENI recommande à ses diplômés de promouvoir l'entrepreneuriat en TIC et de créer des cybercafés, des SSII ou des bureaux d'études.

Table 3 : Débauche professionnelle

LICENCE	<ul style="list-style-type: none"> - Analyste - Programmeur - Administrateur de site web/de portail web - Assistant Informatique et internet - Chef de projet web ou multimédia - Hot liner/Hébergeur Internet - Agent de référencement - Technicien/Supérieur de help desk sur Informatique - Responsable de sécurité web - Administrateur de réseau
MASTER	<ul style="list-style-type: none"> - Administrateur de réseau et système - Architecture de système d'information - Développeur d'applications - Ingénieur réseau - Webmaster /web designer

	<ul style="list-style-type: none"> - Concepteur Réalisateur d'applications - Directeur de projet informatique - Chef de projet informatique - Consultant fonctionnel ou freelance
--	---

D'une manière générale, les diplômés de l'ENI n'éprouvent pas de difficultés particulières à être recrutés au terme de leurs études. Cependant, l'ENI recommande à ses diplômés de promouvoir l'entrepreneuriat en TIC et de créer des cybercafés, des SSII ou des bureaux d'études.

1. 9) **Ressources humaines**

- Directeur de l'École : Docteur MAHATODY Thomas, Docteur HDR
- Responsable de Mention : Monsieur RABETAFIKA Louis Haja, Maître de Conférences
- Responsable de Parcours « Génie Logiciel et Base de Données » : Monsieur RALAIVAO Jean Christian, Assistant d'Enseignement Supérieur et de Recherche
- Responsable de Parcours « Administration Systèmes et Réseaux » : Monsieur SIAKA, Assistant d'Enseignement Supérieur et de Recherche
- Responsable de Parcours « Informatique Générale » : Monsieur Gilante GESAZAFY, Assistant d'Enseignement Supérieur et de Recherche
- Nombre d'Enseignants permanents : 12 dont un (01) Professeur Titulaire, deux (02) Professeurs, cinq (05) Maîtres de Conférences et quatre (04) Assistants d'Enseignement Supérieur et de Recherche
- Nombre d'Enseignants vacataires : 10

Personnel Administratif : 23

chap 2: Présentation de L' Entreprise

2. 1) Historique

Tout a commencé en Juillet 2021. Nous étions une équipe de cinq personnes et nous nous sommes retrouvés dans une salle dans l'une de nos maisons. Nous avons commencé par la distribution de la tâche. Les jours passaient et nous faisions des appels avec des réunions de temps en temps.

Au fil du temps, tout travail mérite d'être récompensé, et nous avons eu notre premier client. Très désireux de commencer et de donner un travail satisfaisant, nous avons tout de suite travaillé sur son projet après avoir analysé le cahier des charges. Par la suite, nous avons décidé qu'il était temps de trouver un nom pour notre entreprise. Nous avons fini par nous mettre d'accord sur le nom « Webdevin ».

À cette époque, nous avions déjà un lieu de travail avec accès à Internet. Nous avons commencé à travailler à cet endroit, dans une salle où nous faisons tout : manger, travailler et nous réunir pour nous organiser. La salle n'était pas grande, mais nous nous en sommes débrouillés.

2. 2) Notre Mission

Notre mission est de simplifier et d'améliorer la vie professionnelle de nos clients en fournissant des solutions informatiques fiables et performantes. Nous nous efforçons constamment d'anticiper les évolutions technologiques pour offrir à nos clients les outils nécessaires pour exceller dans un environnement en constante mutation.

2. 3) Organigramme

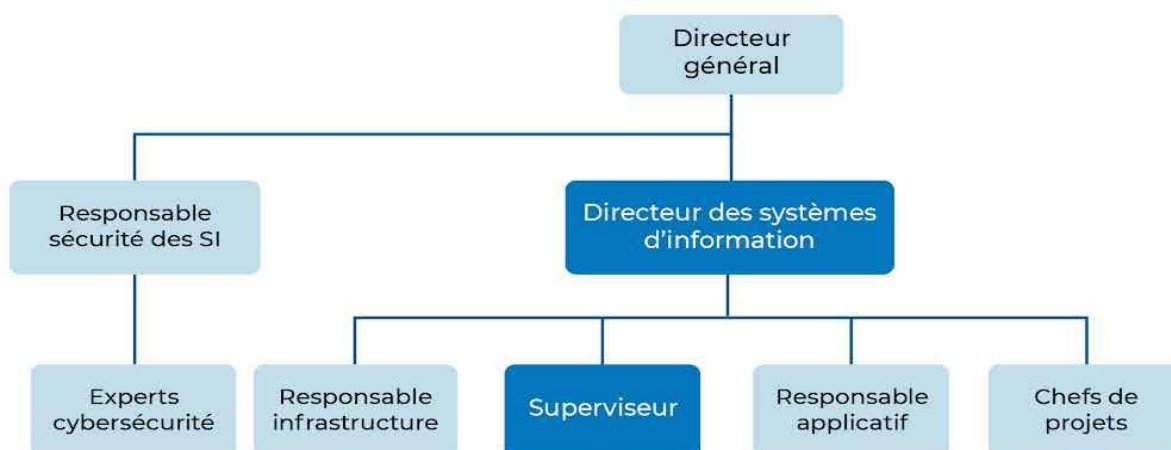


Figure 3 : Organigramme de l'entreprise

chap 3: Description du projet

3. 1) Objectif général

Le but fondamental de ce projet est de développer une plateforme web de recrutement qui simplifie le processus de recrutement pour les entreprises et qui offre aux candidats une expérience conviviale. Le projet se concentrera principalement sur la conception et la mise en œuvre du frontend et du backend de la plateforme, laissant de côté la composante liée à l'intelligence artificielle pour un projet ultérieur.

3. 2) Besoins des utilisateurs

Les recruteurs bénéficieront de la plateforme en simplifiant la gestion de leurs offres d'emploi grâce à une interface intuitive pour la création, la publication et la gestion des offres. De plus, la plateforme rationalisera la communication avec les candidats en offrant une visibilité sur le statut des candidatures, sans recourir à l'analyse des CV par l'IA.

Les candidats profiteront d'une expérience utilisateur conviviale pour postuler aux offres d'emploi. Ils auront la possibilité de télécharger leur CV, de se présenter via une vidéo, de répondre à des quiz pertinents, et de recevoir un accusé de réception pour chaque candidature, sans utiliser l'intelligence artificielle pour le classement.

3. 3) Services proposés

Les services proposés dans le cadre de ce projet seront centrés sur le développement du frontend et du backend de la plateforme. Ils incluront une interface utilisateur conviviale pour la création d'offres d'emploi, mais ne comporteront pas l'analyse des CV par l'IA. La gestion des candidatures sera également facilitée par une interface conviviale pour les candidats, avec la possibilité de partager les résultats de classement sans avoir recours à l'IA.

3. 4) Objectifs spécifiques

Les objectifs spécifiques de ce projet englobent plusieurs aspects essentiels. Tout d'abord, nous avons pour but de concevoir et mettre en œuvre de manière efficace le frontend et le backend de la plateforme de recrutement. Cela inclut la création d'une

interface utilisateur intuitive pour la gestion des offres d'emploi, la mise en place de bases de données robustes pour stocker les informations des candidats et des offres, ainsi que le développement de fonctionnalités avancées pour garantir la convivialité et la fluidité de la plateforme.

En parallèle, notre objectif est de simplifier de manière significative le processus de recrutement pour les recruteurs. Nous voulons leur offrir un outil qui leur permette de créer, de publier et de gérer leurs offres d'emploi de manière rapide et efficace, tout en améliorant la gestion des candidatures. Cela implique la création d'une interface administrative intuitive, des notifications en temps réel pour les nouvelles candidatures, et des outils de communication pour faciliter les échanges avec les candidats.

Pour les candidats, nous cherchons à offrir une expérience utilisateur exceptionnelle. Ils devront avoir la possibilité de postuler aux offres d'emploi de manière simple et rapide, avec la garantie de recevoir un accusé de réception pour chaque candidature. Nous mettrons en place des outils de téléchargement de CV, de présentation via des vidéos, et de réponse à des quiz pertinents pour chaque offre.

Un autre objectif essentiel est d'optimiser les ressources, en particulier le temps et l'effort investis dans le processus de recrutement. Nous visons à aider les recruteurs à se concentrer sur les candidats les plus pertinents en automatisant certaines étapes du tri et de la sélection. Cela permettra de réduire le gaspillage de ressources et d'accélérer la prise de décision en matière de recrutement.

Enfin, nous avons pour ambition d'accroître la visibilité des offres d'emploi en les publiant sur les réseaux sociaux et d'attirer un large vivier de candidats de qualité. Cela renforcera la notoriété de la plateforme et contribuera à son succès à long terme dans le domaine du recrutement en ligne.

3. 5) Moyens nécessaire a la réalisation du projet

3. 5. 1) Moyen humain

L'ensemble du projet sera pris en charge par un seul individu, qui sera responsable de la gestion globale du développement de la plateforme. Ce professionnel expérimenté dans le domaine du développement web sera chargé de superviser la conception du frontend et du backend, ainsi que de coordonner toutes les activités liées au projet.

3. 5. 2) Moyen logiciel

Un ensemble d'outils logiciels a été choisi pour soutenir le développement de la plateforme. Parmi ces outils figurent Visual Studio Code (VSCode), qui sera utilisé comme environnement de développement intégré (IDE), GitLab pour la gestion du code source et des versions, Visual Paradigm pour la modélisation et la conception et ClickUp pour la gestion de projet et la collaboration en équipe. Ces logiciels seront employés pour faciliter le développement, maintenir la qualité du code et de la conception, ainsi que pour permettre une gestion efficace du projet.

3. 6) Moyen matériel

En ce qui concerne les ressources matérielles, un ordinateur portable avec une connexion internet stable sera utilisé comme principal support pour le projet. Cette configuration permettra de mener à bien toutes les activités de développement, de gestion de projet et de communication en ligne de manière efficace et sans interruption.

PARTIE II. ANALYSE ET CONCEPTION

chap 4: Analyse Préalable

4. 1) Analyse de l'existant

4. 1. 1) Organisation actuelle

Les modes de traitement actuels incluent la manière dont les offres d'emploi sont créées, gérées et diffusées, ainsi que la façon dont les candidatures sont collectées, triées et évaluées. Il est essentiel de comprendre les étapes spécifiques du processus, les flux de travail existants et les interactions entre les différents acteurs.

Les applications logicielles actuelles jouent un rôle clé dans le processus de recrutement. Il est nécessaire de recenser ces applications, qu'il s'agisse de logiciels de gestion des candidatures, de systèmes de suivi des candidats (ATS), de plateforme de publication d'offres d'emploi ou d'autres outils pertinents. Cette liste d'applications actuelles permettra d'analyser leur efficacité et leur intégration dans le processus global.

Enfin, il est primordial d'identifier les personnels impliqués dans le processus de recrutement, en précisant leurs rôles et responsabilités. Cela comprend les recruteurs, les responsables des ressources humaines, les gestionnaires de recrutement, les candidats et toute autre partie prenante. Comprendre qui intervient à chaque étape du processus est essentiel pour concevoir une solution qui optimise les flux de travail et améliore l'expérience de tous les acteurs impliqués.

4. 2) Critique de l'existant

Il existe plusieurs lacunes et inefficacités dans le processus de recrutement actuel de l'entreprise. Voici quelques points de critique importants :

- **Complexité du processus** : Le processus de recrutement actuel est complexe, avec de nombreuses étapes manuelles, ce qui entraîne une perte de temps pour les recruteurs. La création, la publication et la gestion des offres d'emploi sont souvent laborieuses, nécessitant des tâches répétitives.
- **Manque de centralisation** : Les données liées aux candidatures et aux offres d'emploi ne sont pas centralisées. Cela rend difficile le suivi des candidats et des positions ouvertes, ce qui peut entraîner des erreurs et une perte de visibilité.
- **Communication inefficace** : La communication avec les candidats n'est pas optimale. Les délais de réponse sont parfois longs, ce qui peut frustrer les candidats et nuire à la réputation de l'entreprise en tant qu'employeur.

- **Tri des candidatures laborieux** : Le tri des candidatures est effectué manuellement, ce qui peut être fastidieux et subjectif. Cela peut également entraîner la sélection de candidats moins pertinents en raison d'une charge de travail importante.
- **Manque d'analyse des données** : L'absence d'outils d'analyse des données rend difficile l'évaluation de l'efficacité du processus de recrutement. L'entreprise n'a pas la capacité de mesurer les indicateurs clés de performance et d'identifier les domaines nécessitant des améliorations.
- **Expérience candidat insatisfaisante** : Les candidats ne bénéficient pas d'une expérience optimale. Les délais de réponse, l'absence d'accusé de réception et les processus de candidature complexes peuvent dissuader les talents potentiels de postuler.

En résumé, le processus de recrutement actuel présente des inefficacités majeures en termes de gestion des offres d'emploi, de suivi des candidatures, de communication avec les candidats et de collecte de données. Une analyse approfondie de ces lacunes est essentielle pour orienter la conception et le développement de la nouvelle plateforme de recrutement et pour améliorer l'efficacité globale du processus de recrutement de l'entreprise.

4. 3) Conception avant projet(solutions, méthodes et outils proposés)

Table 4 : Conception avant projet

Methodes proposées	Outils proposées	Langage de programmation et Frameworks
- MERISE - AGILE (Scrum)	- VSCode - Visual Paradigm - GitLab - ClickUp	- NextJs - NestJs - MongoDB

4. 3. 1) Solution proposées

Interface utilisateur intuitive : Une interface utilisateur conviviale sera développée pour permettre aux recruteurs de créer, gérer et publier facilement des offres d'emploi. Les candidats auront également une expérience fluide pour postuler aux offres.

Automatisation du tri des candidatures : Un système automatisé de tri des candidatures sera mis en place pour classer les candidats en fonction de leur pertinence pour le poste. Cela réduira la charge de travail manuelle et assurera un traitement plus efficace des candidatures.

Communication améliorée : Des fonctionnalités de communication en temps réel seront intégrées pour faciliter les échanges avec les candidats. De plus, un accusé de réception automatique sera généré pour chaque candidature, améliorant ainsi l'expérience candidat.

4. 3. 2) Méthodes proposées

a) Méthode de conception: MERISE

La méthode MERISE (Méthode d'Étude et de Réalisation Informatique pour les Systèmes d'Entreprise) sera utilisée comme cadre méthodologique pour concevoir et modéliser la nouvelle plateforme de recrutement. Elle fournira une structure rigoureuse pour la gestion des données et des processus.

i. Justification: Pourquoi MERISE?

La méthode MERISE est une méthode de conception et de développement d'un système d'information qui date de 1978-1979. Elle permet de représenter les interactions entre les différents composants d'un système d'information et d'en proposer une description formelle¹. La méthode MERISE préconise trois niveaux d'abstraction pour concevoir un système d'information¹:

Niveau conceptuel: Ce niveau décrit la statique et la dynamique du système d'information en se préoccupant uniquement de la gestion des données. Il répond à la question "QUOI" c'est-à-dire quoi faire avec quelle donnée.

Niveau organisationnel: Ce niveau décrit la nature des ressources utilisées pour supporter la description statique et dynamique du système d'information. Ces ressources peuvent être humaines, matérielles ou logicielles. Il répond aux questions "QUI", "OÙ" et "QUAND" et intègre les critères d'organisation du travail.

Niveau physique: Ce niveau concerne le choix des techniques du système d'information (données et traitements). Il s'agit de créer la base de données selon la syntaxe du SGBD (Système de Gestion de Base de Données).

La méthode MERISE comporte également trois modèles différents:

Modèle conceptuel de communication: Aussi appelé diagramme de flux, ce modèle permet de représenter les flux d'informations entre les acteurs internes ou externes.

Modèle conceptuel de données: Ce modèle formalise la structure et la signification des informations dans le domaine d'étude. Il est exprimé en entité-relation, où l'entité est un objet (par exemple, une chaise) et la relation est un verbe qui permet de lier deux ou plusieurs entités.

Modèle conceptuel de traitement: Ce modèle décrit le fonctionnement du système d'information d'une organisation au niveau conceptuel. Il consiste à recenser la liste de tous les acteurs intervenants dans le système d'information et à schématiser les flux d'information qu'ils échangent.

b) Méthode de développement: AGILE

Pour répondre aux besoins changeants de l'entreprise, la plateforme sera développée en utilisant la méthodologie Agile. Cette approche permet des itérations rapides et une flexibilité dans l'ajout de fonctionnalités.

i. Justification: Pourquoi utiliser la méthode agile?

La méthode Agile et Scrum sont utilisées pour les raisons suivantes :

- Les méthodes Agiles permettent de répondre aux besoins changeants des clients grâce à l'itération et la livraison continue. Les équipes recueillent des commentaires des clients à intervalles réguliers et adaptent le produit ou le service en fonction de ces commentaires.
- Les méthodes Agiles encouragent la communication et la collaboration entre les différentes parties prenantes. Cela permet de garantir que tout le monde est sur la même longueur d'onde et que les projets sont livrés à temps et dans les limites du budget.
- Les méthodes Agiles sont conçues pour augmenter la productivité et l'efficacité des équipes en encourageant la responsabilisation et la prise d'initiative. Cela permet aux équipes de se concentrer sur les tâches les plus importantes.

Scrum est une méthode Agile qui se concentre sur le développement itératif et incrémental de produits ou de services. Les équipes Scrum travaillent en sprints de deux semaines, au cours desquels elles livrent un ensemble de fonctionnalités ou de corrections de bogues.

Les avantages de Scrum incluent :

- La livraison de produits ou de services plus rapidement et plus fréquemment.
- Une meilleure communication et collaboration entre les équipes.
- Une plus grande responsabilisation et prise d'initiative des équipes.

- Une plus grande flexibilité et capacité d'adaptation aux changements.

Bref la méthode Agile et Scrum sont des approches de gestion de projet qui permettent de répondre aux besoins changeants des clients, d'améliorer la communication et la collaboration, et d'augmenter la productivité.

4. 3. 3) Outils proposées

- **Visual Studio Code (VSCode)** : VSCode est un environnement de développement léger et extensible, idéal pour le développement frontend et backend. Sa polyvalence en fait un choix judicieux pour le développement web.
- **GitLab** : GitLab est une plateforme de gestion du code source qui offre des fonctionnalités de collaboration, d'intégration continue et de déploiement continu, facilitant ainsi la gestion de tout le cycle de vie du code.
- **Visual Paradigm** : Visual Paradigm est un outil de modélisation UML robuste qui permet de représenter visuellement l'architecture et la conception de la plateforme, facilitant la compréhension de l'équipe de développement.
- **ClickUp** : ClickUp est un outil de gestion de projet collaboratif qui permet de suivre les tâches, les délais et la communication au sein de l'équipe de développement de manière transparente, contribuant ainsi à une gestion de projet efficace.

4. 3. 4) Langage de programmation et framework proposées

- **Next.Js**: Next.js est un framework frontend basé sur React.js qui offre de nombreux avantages pour le développement d'applications Web modernes. Voici trois raisons de choisir Next.js :
 - Pre-rendering : Next.js prend en charge le pré-rendu, ce qui signifie que le contenu de la page est généré sur le serveur avant d'être envoyé au navigateur. Cela permet d'améliorer les performances des applications Web, en particulier pour les pages statiques.
 - SEO : Next.js est optimisé pour le référencement naturel (SEO). Cela signifie que les pages Web générées par Next.js sont plus susceptibles d'être indexées par les moteurs de recherche.
 - Fonctionnalités avancées : Next.js offre une gamme de fonctionnalités avancées, notamment le chargement paresseux, les pages dynamiques et les API routes. Ces fonctionnalités peuvent être utilisées pour créer des applications Web plus puissantes et plus interactives.

- **NestJS:** NestJS est un framework Node.js open source qui permet de créer des applications côté serveur efficaces et évolutives. Il est construit sur TypeScript et utilise ExpressJS sous le capot. Donc c'est un framework puissant et polyvalent qui est un excellent choix pour les plateformes de recrutement. Il offre une combinaison d'efficacité, d'évolutivité et de flexibilité qui est essentielle pour ce type d'application.

- **MongoDB :**

MongoDB est un choix judicieux pour votre projet de plateforme de recrutement. Il offre plusieurs avantages spécifiques à ce contexte, notamment :

- Une modélisation flexible des données, qui permet de stocker des informations diverses de manière plus libre et évolutive.
- Une haute évolutivité, qui est essentielle pour gérer une grande quantité de données et répondre à la croissance future de la plateforme.
- De bonnes performances, qui garantissent une expérience utilisateur réactive.
- La prise en charge des données semi-structurées, qui est courante dans les données liées au recrutement.
- Une grande adoption, qui signifie qu'il existe une vaste communauté de développeurs, de ressources et de support.

chap 5: Analyse conceptuel

5. 1) Présentation de la méthode utilisé(Agile)

5. 2) Dictionnaire de données:

Un dictionnaire de données est un document qui décrit les données stockées dans une base de données. Il comprend des informations sur la signification, la source, le format, la taille et les relations entre les données.

Table 5 : Dictionnaire de données

NOM	SIGNIFICATION	TYPE	TAILLE
IdUser	Identifiant de l'utilisateur	AN	20
username	Nom de l'utilisateur	AN	255
mailUser	Mail de l'utilisateur	AN	255
role	Rôle de l'utilisateur	AN	50
mdp	Mot de passe de l'utilisateur	AN	255
Activee	Si le compte de l'utilisateur est activée	B	
contactUser	Contact de l'utilisateur	AN	255
IdFicImage	Identifiant du fichier pour le photos de profil	AN	20
IdOffre	Identifiant de l'offre	AN	20
IdFicDescription	Identifiant du fichier pour stocker le description	AN	20
datedebut	La date de début de candidature	D	
datefin	Date limite pour la candidature	D	
IdQuiz	Identifiant du quiz	AN	20
Question	Question a poser aux candidats	AN	255
IdCandidat	Identifiant d'un candidats	AN	20
numCIN	Numéro CIN des candidats	AN	100
mailCandidat	Adresse mail des candidats	AN	255

dateCandidature	Date d'envoi du candidature	D	
noteCV	Résultat de classification du CV	N	
noteVideo	Résultat de classification des réponses aux quiz	N	
moyenne	Moyenne des résultats de classification	N	
IdFicCV	Identifiant du fichier CV	AN	20
IdFicInfosCV	Identifiant du fichier contenant les informations extractés depuis le CV	AN	20
IdFicVideo	Identifiant du fichier Vidéo	AN	20
IdFic	Identifiant du fichier	AN	20
nomFic	Nom du fichier	AN	255
typeFic	Type du fichier(image,vidéo,...)	AN	50
categorie	Utilisation de la vidéo(photos de profil, CV,Description de poste)	AN	255

5. 3) Règle de gestions:

La création d'une règle de gestion est une étape indispensable pour définir les règles et les contraintes qui guideront le fonctionnement d'un système ou d'une organisation.

- RG1: Le recruteur peut créer plusieurs offres d'emplois et le publier sur les réseaux sociaux
- RG2: Un recruteur peut voir le résultat de classification en tout temps
- RG3: L'administrateur a accès a toutes les données et fonctionnalités de la plateforme tandis que le recruteur n'a accès qu'aux offres qu'il a crée
- RG4: L'administrateur doit d'abord activer le compte des recruteurs pour que ces derniers puissent publier des offres
- RG5: Les candidats peuvent postuler en allant vers le lien de candidature obtenu de la part des recruteurs via les réseaux sociaux
- RG6: Un candidat ne peut pas postuler qu'une fois sur un même poste, ceci est vérifiée en utilisant son numéro de CIN
- RG1: Pour postuler le candidat doit envoyer son CV et répondre a un QUIZ aléatoire par vidéo

5. 4) Représentation et spécification des besoins:

Dans le cadre du projet "Développement et implémentation d'une plateforme web de classification de CV basée sur l'analyse des critères du poste", la représentation et la spécification des besoins consistent à définir de manière détaillée les besoins fonctionnels et non fonctionnels du système. Cela inclut l'identification des besoins des recruteurs, des candidats, et du système lui-même.

5. 4. 1) Diagramme des cas d'utilisation :

Un diagramme des cas d'utilisation pourrait être élaboré pour représenter les interactions entre les acteurs du système. Les acteurs principaux seraient les recruteurs et les candidats. Les cas d'utilisation clés pourraient inclure "Créer une offre d'emploi", "Soumettre une candidature", "Classer les candidatures", "Publier une offre d'emploi sur les réseaux sociaux", etc. Ce diagramme permettrait de visualiser comment les acteurs interagissent avec le système pour atteindre leurs objectifs.

5. 4. 2) Priorisation des cas d'utilisation :

La priorisation des cas d'utilisation consisterait à déterminer quels cas d'utilisation sont les plus cruciaux pour le succès du projet. Dans ce contexte, des cas d'utilisation tels que "Créer une offre d'emploi" et "Soumettre une candidature" pourraient être considérés comme essentiels. D'autres cas d'utilisation, tels que "Publier une offre d'emploi sur les réseaux sociaux", pourraient être moins prioritaires, mais toujours importants.

5. 4. 3) Diagramme de séquence système pour chaque cas d'utilisation :

Pour chaque cas d'utilisation clé, un diagramme de séquence système pourrait être élaboré pour décrire comment le système interagit avec les acteurs. Par exemple, pour le cas d'utilisation "Classer les candidatures", le diagramme de séquence système montrerait comment le système traite les CV des candidats en utilisant l'IA, comment il génère des résultats, et comment il permet au recruteur de prendre des décisions.

5. 5) Spécification des besoins :

La spécification des besoins impliquerait la rédaction détaillée des exigences du système. Cela comprendrait la description des fonctionnalités, des contraintes techniques, des exigences de sécurité, de performance, etc. Par exemple, les exigences pour la capacité de stockage des CV, la sécurité des données personnelles des candidats, la vitesse de traitement de l'IA, etc.

5. 6) Modélisation du domaine:

La modélisation du domaine consisterait à créer une représentation abstraite des concepts et des entités liés au recrutement et à l'analyse des CV. Cela inclurait des éléments tels que les entités "Poste", "Candidat", "CV", "Résultats de classement", etc. La modélisation du domaine aiderait à clarifier la structure des données et des relations dans le contexte du projet.

L'application de ces étapes dans le projet contribuerait à une compréhension claire des besoins, à une communication efficace entre les parties prenantes et à une base solide pour le développement de la plateforme de recrutement.

chap 6: Conception détaillée

6. 1) Diagramme de flux de données (DFD):

Le diagramme de flux est un élément clé de la méthode MERISE. Il permet de représenter les flux d'informations entre l'organisation et les acteurs externes, en définissant le système et les éléments externes avec lesquels il échange des flux d'information.

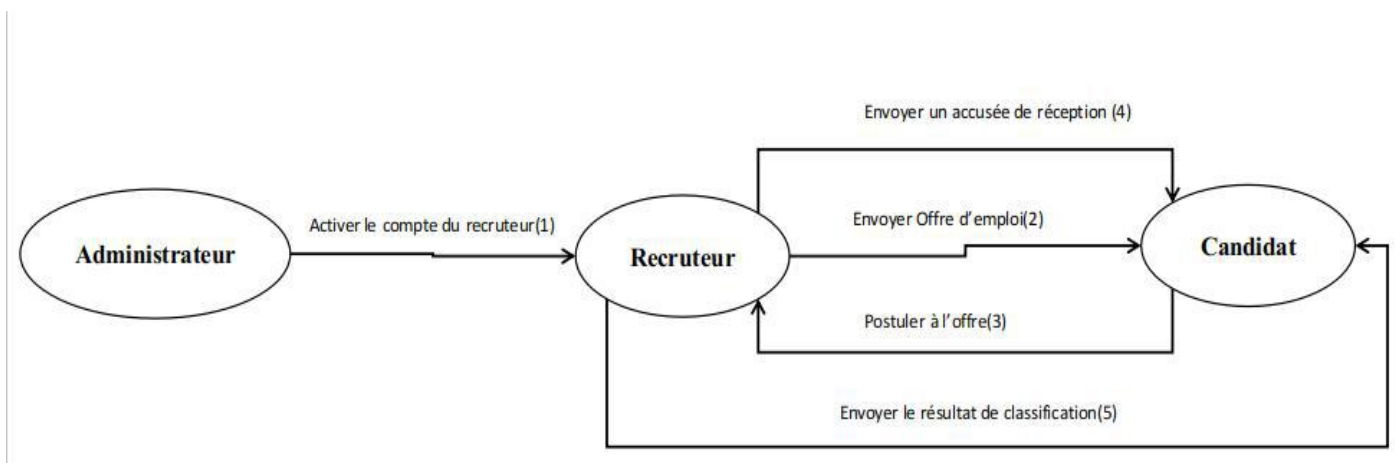


Figure 4 : Diagramme de flux de données

6. 2) Modèle conceptuel de données (MCD):

Le Modèle Conceptuel de Données (MCD) est un outil majeur dans la méthodologie MERISE. Il permet de représenter de manière structurée les données utilisées par le système d'information et leurs relations. Le MCD décrit la sémantique des données, c'est-à-dire le sens attaché à ces données et à leurs rapports. Il est utilisé pour décrire le système d'information à l'aide d'entités et de leurs relations. Le MCD est une représentation graphique de haut niveau qui permet de comprendre facilement comment les différents éléments sont liés entre eux.

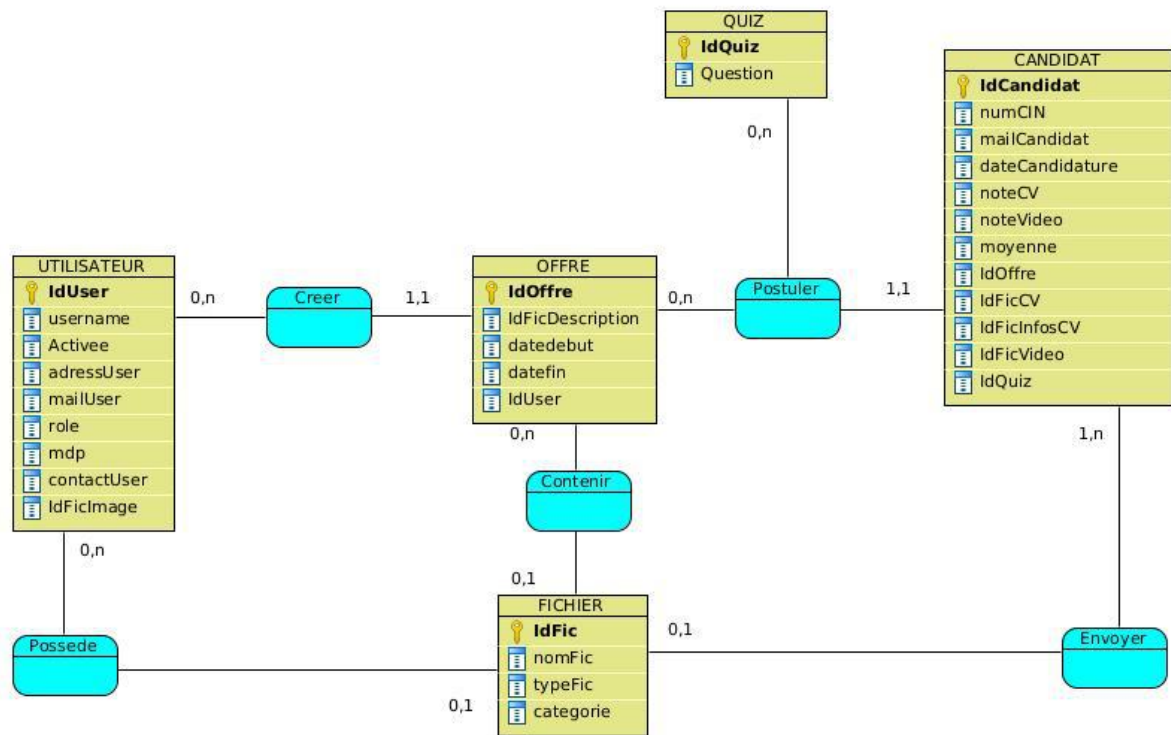


Figure 5 : Modele conceptuel de données

6. 3) Modèle conceptuel de traitement (MCT):

Le MCT est un élément clé de la méthode MERISE. Il représente l'activité d'un SI sans tenir compte de l'organisation ou des moyens d'exécution. Il définit simplement ce qui doit être fait, sans préciser les détails temporels, techniques ou géographiques.

Voir dans ANEXE C pour mieux comprendre l'importance des DFD, MCT et MCD dans le développement d'une plateforme web.

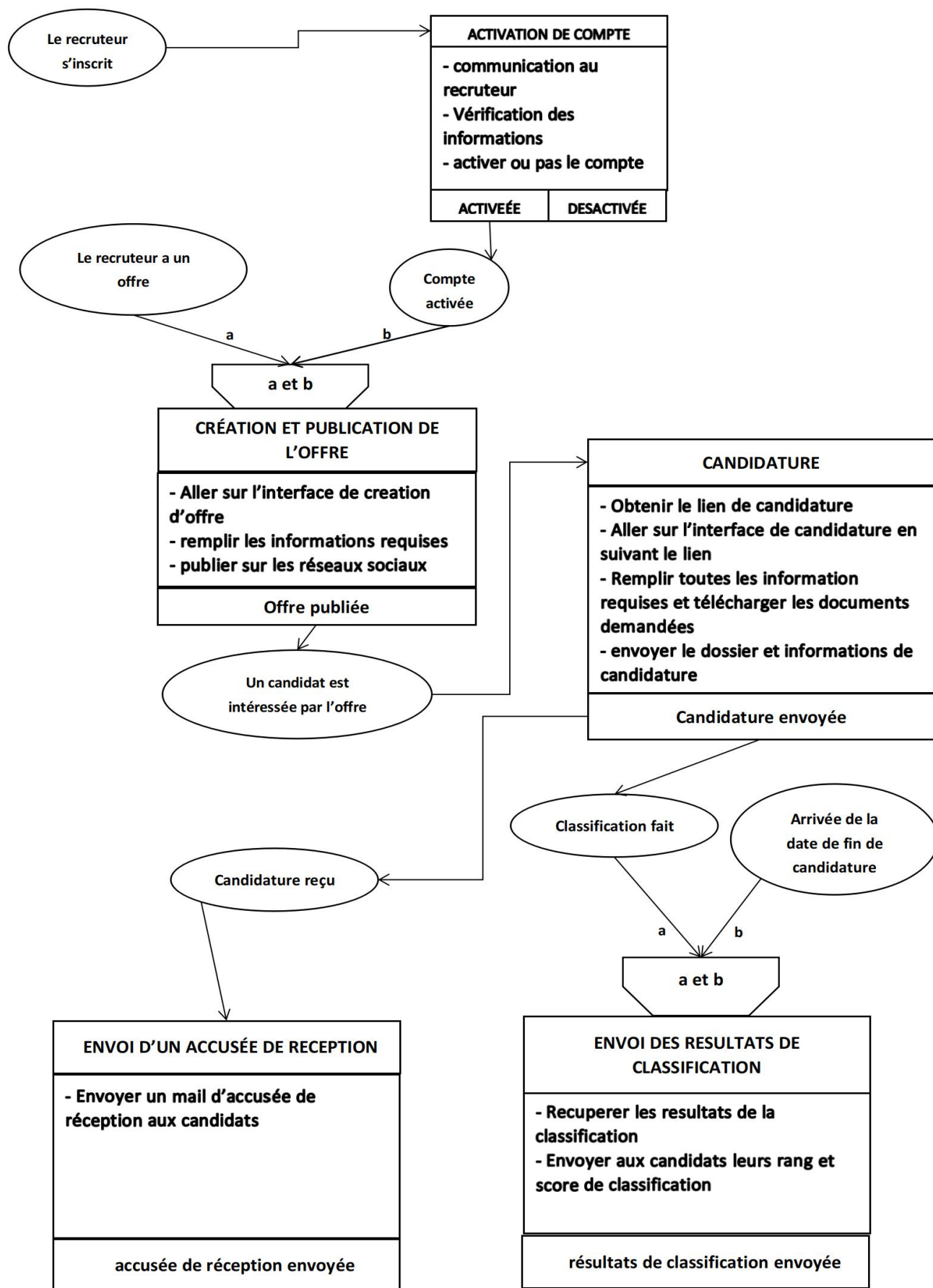


Figure 6 : Modele conceptuel de traitement

PARTIE III. REALISATION

La phase de réalisation de l'application dynamique, qui est l'objectif principal de cette partie, est divisée en deux chapitres. Le premier chapitre traite de la mise en place de l'environnement de développement, tandis que le second chapitre décrit le développement de l'application.

chap 7: Mise en place de l'environnement de développement

La réalisation du projet proprement dit aura besoin d'outils logiciels pour la conception ainsi que pour le développement.

7. 1) Installation et configuration des outils:

7. 1. 1) Création du dépôt gitlab:

Pour utiliser GitLab pour la gestion de versions du projet dès sa création, il faut suivre ces étapes :

1. se connecter à GitLab.
2. entrer dans le workspace de la société
3. Créez un nouveau subgroup "recrutement".
4. Créez trois nouveaux projets dans le subgroup "recrutement" :
 - frontend
 - fastapi
 - Nestjs
5. Créez un nouveau subgroup "backend" dans le subgroup "recrutement". Déplacez les projets "fastapi" et "nestjs" dans le subgroup.

7. 1. 2) Création des projet

a) Next.js:

- i. **Installer Next.js:** Pour installer Next.js, vous pouvez utiliser la commande suivante :

npm install next

- ii. **Créer un nouveau dossier pour votre projet:** Ouvrez un terminal et créez un nouveau dossier pour votre projet :

mkdir mon-projet

- iii. **Entrez dans le dossier de votre projet:** Accédez au dossier de votre projet en utilisant la commande suivante :

cd mon-projet

- iv. **Initialiser un nouveau projet Next.js:** Exécutez la commande suivante pour initialiser un nouveau projet Next.js :
`npx create-next-app@latest`

Cette commande créera un nouveau dossier pour votre projet Next.js et installera les dépendances nécessaires.

- v. **Démarrer le serveur de développement:** Pour démarrer le serveur de développement, utilisez la commande suivante :
`npm run dev`

- vi. **Ouvrez un navigateur web et rendez-vous à l'adresse suivante :**
`http://localhost:3000`

b) **Nestjs:**

- i. **Installer NestJS:** Pour installer NestJS, vous pouvez utiliser la commande suivante :**`npm install -g @nestjs/cli`**
- ii. **Cette commande installera la CLI NestJS sur votre système:** Créer un nouveau dossier pour votre projet. Ouvrez un terminal et créez un nouveau dossier pour votre projet :**`mkdir mon-projet`**
- iii. **Entrez dans le dossier de votre projet:** Accédez au dossier de votre projet en utilisant la commande suivante :**`cd mon-projet`**
- iv. **Initialiser un nouveau projet NestJS:** Exécutez la commande suivante pour initialiser un nouveau projet NestJS :
`nest new my-app`

Cette commande créera un nouveau dossier pour votre projet NestJS et installera les dépendances nécessaires.

- v. **Démarrer le serveur de développement:** Pour démarrer le serveur de développement, utilisez la commande suivante :
`nest start`

Cette commande lancera l'application NestJS sur le port **3000**.

7. 1. 3) **MongoDB:**

- i. Installer MongoDB: Pour installer MongoDB, vous pouvez utiliser la commande suivante :
`apt install mongodb-community`
- ii. Pour démarrer le serveur MongoDB, utilisez la commande suivante :
`systemctl start mongod`
- iii. Ouvrir le shell MongoDB: Pour ouvrir le shell MongoDB, utilisez la commande suivante :
`mongosh`
- iv. Créer une nouvelle base de données: Pour créer une nouvelle base de données, utilisez la commande suivante :
`use my_database`
- v. Fermer le shell MongoDB: Pour fermer le shell MongoDB, utilisez la commande suivante :
`exit`

7. 2) **Architecture de l'application:**

L'architecture de l'application est basée sur un modèle MVC (Model-View-Controller). Le modèle est responsable du traitement des données, la vue est responsable de l'affichage des données, et le contrôleur est responsable de la communication entre le modèle et la vue.

7. 2. 1) Model

Le modèle est responsable du stockage et du traitement des données. Il est divisé en trois sous-modèles :

- Le modèle offre d'emploi, qui stocke les informations sur les postes à pourvoir.
- Le modèle candidat, qui stocke les informations sur les candidats.
- Le modèle classement, qui stocke les résultats du classement des CV.

7. 2. 2) Vue

La vue est responsable de l'affichage des données. Elle est divisée en deux sous-vues :

- La vue recruteur, qui est accessible aux recruteurs.
- La vue candidat, qui est accessible aux candidats.

7. 2. 3) Contrôleur

Cette architecture est adaptée au projet de plateforme web de classification de CV. Elle permet de séparer les responsabilités entre le modèle, la vue et le contrôleur, et de faciliter le développement et la maintenance de l'application.

Recommandations

Pour améliorer l'architecture de l'application, il est possible de mettre en place les améliorations suivantes :

- Utiliser un système de cache pour améliorer les performances.
- Utiliser un système de sécurité pour protéger les données.
- Mettre en place un système de monitoring pour suivre l'état de l'application.
- Ces améliorations peuvent être réalisées au fur et à mesure du développement de l'application.

Architecture spécifique de l'application

L'architecture d'une application peut être adaptée en fonction des fonctionnalités spécifiques que l'on souhaite implémenter. Par exemple, pour traiter des CV en plusieurs langues, il est possible d'ajouter un modèle de traduction. Pour générer des rapports sur les résultats de classement, il est possible d'ajouter un modèle de reporting.

Il est important de bien réfléchir à l'architecture d'une application avant de commencer à la développer. Une architecture bien conçue permet de développer l'application plus rapidement et plus facilement, et de la maintenir plus efficacement.

chap 8: Développement de l'application

Avant d'entrer dans le vif du sujet, il faut savoir que le développement d'un site web peut être très long. Il est donc important de bien répartir le temps de développement en divisant l'application par fonctionnalités. Il faut ensuite exécuter d'abord les fonctionnalités les plus importantes.

Voici la liste des fonctionnalités de la plateforme, réparties par ordre de priorité :

- Envoi de la candidature:
 - ◆ Upload CV
 - ◆ Vidéo
 - ◆ Mail(pour envoyer l'accusé de réception)
- Extraction des données
- Traitement langage naturel
- Classification
 - A base du CV et l'offre
 - A base du vidéo
- Création de description de l'offre
 - Deux options
 - ◆ Saisir
 - Suggérer modèle existant
 - ◆ Upload de fichier contenant la description de l'offre
 - Il y a une date limite pour la candidature
 - Il y a une date de début pour l'envoi candidature
 - Créer des quiz a partir du description:
 - ◆ Stocké dans la base de données
 - ◆ Question générales
- Authentification
- Affichage des résultats

8. 1) Création de la base de donnes

La création de la base de donnes est assez simple avec mongosh,il suffit de suivre cet étape:

- ✓ Tout d'abord, vérifier que mongodb est installer
- ✓ Ouvrir le terminal (ctrl+T sur Linux)
- ✓ Lancer mongosh avec la commande: **mongosh**
- ✓ Créer la base de donner en tapant: **use recrutement**

Et voila la base de donnés est crée

8. 2) Faire un design Figma

La conception de maquettes dans des outils comme Figma est une étape importante lors du développement d'un site web pour plusieurs raisons :

- ✓ Visualisation et communication
- ✓ Réflexion préliminaire
- ✓ Collaboration
- ✓ Tests utilisateurs
- ✓ Conformité au design
- ✓ Gain de temps et d'argent
- ✓ Documentation du projet

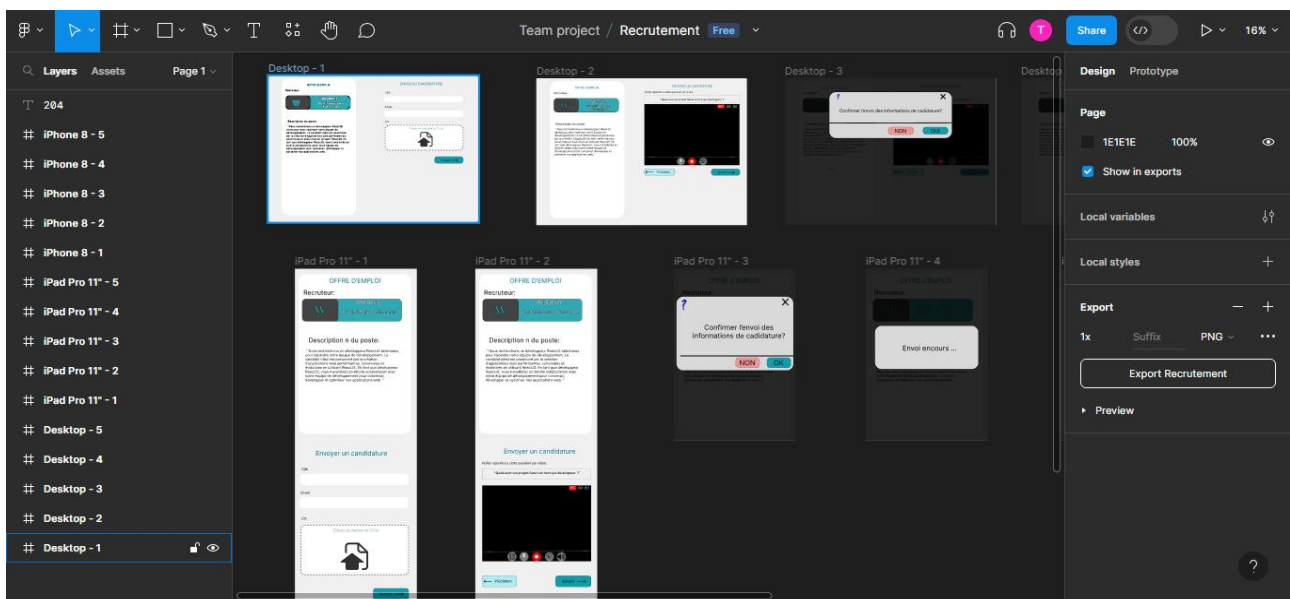


Figure 7 : Design Figma

8. 3) Codage de l'application

8. 3. 1) Coté Backend(NestJS)

NestJS est un framework Node.js qui permet de créer des applications serveur robustes et évolutives, avec les avantages suivants :

- ✓ **Une structure de projet organisée** : NestJS encourage une structure de projet organisée, avec une séparation claire entre les **modules**, les **contrôleurs** et les **services**. Cette organisation facilite la gestion de l'application à mesure qu'elle devient plus complexe.
- ✓ **Une architecture à microservices** : NestJS est basé sur les principes de l'architecture à microservices, ce qui permet de créer des applications plus évolutives et faciles à maintenir.
- ✓ **Une forte influence d' Angular** : NestJS est fortement influencé par le framework Angular, ce qui facilite la transition des développeurs Angular vers Node.js.
- ✓ **Des fonctionnalités avancées** : NestJS offre des fonctionnalités avancées pour gérer efficacement les requêtes HTTP, les bases de données et d'autres aspects courants du développement d'applications web.

a) Gérer la base de donnée

Pour manipuler une base de données dans NestJS, il faut généralement créer cinq fichiers distincts pour chaque entité, comme un candidat. Voici une brève description de chacun de ces fichiers :

- ✓ **candidat.controller.ts** : Ce fichier contient la logique de gestion des requêtes HTTP pour l'entité "candidat". Il définit les points de terminaison tels que la création, la récupération, la mise à jour et la suppression (CRUD) des candidat

```
7 @ApiTags('candidat')
8 @Controller('candidat')
9 export class CandidatController {
10     constructor(private candidatService : CandidatService){}
11
12     @Post()
13     @ApiOperation({ summary: 'Enregistrer un candidat' })
14     @ApiBody({type:createCandidatDTO})
15     @ApiResponse({ status: 200, description: 'candidature enregistré', type: [Candidat] })
16     saveCandidat(@Body() candidat : createCandidatDTO) {
17         return this.candidatService.saveCandidat(candidat);
18     }
19
20     @Post('verifierCandidat')//verifier si un candidat a deja postulé au poste
21     @ApiOperation({ summary: 'verifier si un candidat a deja postulé au poste' })
22     @ApiResponse({ status: 200, description: '" si la candidature est valide et l\'id du candidat s\'il n\'est pas valide', type: [Object] })
23     verifierCandidat(@Body() body){
24         return this.candidatService.verifierCandidat(body);
25     }
26
27
28     @Post('verifierIdCandidat')//verifier sil'id du candidat est valide
29     @ApiOperation({ summary: 'recuperer les information sur le candidat grace a son _id' })
30     @ApiResponse({ status: 200, description: '" si le candidat n\'existe pas et l\'id du candidat s\'il existe', type: [Object] })
31     verifierIdCandidat(@Body() body){
32         return this.candidatService.findById(body.id_candidat);
```

Figure 8 : candidat.controller.ts

- ✓ **candidat.service.ts** : Ce fichier est responsable de la logique métier associée à l'entité "candidat". Il gère les opérations de création, de récupération, de mise à jour et de suppression des candidats.

```
nestjs > src > candidat > TS candidat.service.ts > CandidatService
1  import { Injectable } from '@nestjs/common';
2  import { createCandidatDTO } from '../dto/candidat.dto';
3  import { InjectModel } from '@nestjs/mongoose';
4  import { Model } from 'mongoose';
5  import { Candidat } from '../schemas/candidat.schemas';
6  import { ObjectId } from 'mongoose';
7
8  @Injectable()
9  export class CandidatService {
10     constructor(@InjectModel(Candidat.name) private CandidatModel: Model<Candidat>){}
11
12     //enregistrer un candidat dans la base de données
13     saveCandidat(candidat: createCandidatDTO) : any {
14         const oneCandidat=new this.CandidatModel(candidat)
15         return oneCandidat.save()
16     }
17
18     //verifier s'il existe deja un utilisateur qui a le numeroCIN et l'idOffre,retourne true si oui et false sinon
19     verifierCandidat(infos:{numCIN:string,idOffre:string}) : any{
20         return this.CandidatModel.exists({numCIN:infos.numCIN,idOffre:infos.idOffre})
21     }
22
23     //trouver un candidat grace a son id
24     findById(id: string) : any {
25         if (id.length===24) return this.CandidatModel.findById(id).exec()
26         else return ''
27     }
28 }
```

Figure 9 : candidat.service.ts

- ✓ **candidat.module.ts** : Ce fichier encapsule le contrôleur, le service et d'autres composants liés à l'entité "candidat". Il est responsable de la configuration de l'injection de dépendances (DI) et de la déclaration des composants associés.

```
nestjs > src > candidat > TS candidat.module.ts > CandidatModule
1  import { Module } from '@nestjs/common';
2  import { CandidatController } from '../candidat.controller';
3  import { CandidatService } from '../candidat.service';
4  import { Candidat,CandidatSchemas } from '../schemas/candidat.schemas';
5  import { MongooseModule } from '@nestjs/mongoose';
6
7  @Module({
8     imports:[MongooseModule.forFeature([({name:Candidat.name,schema:CandidatSchemas}))],
9     controllers: [CandidatController],
10     providers: [CandidatService]
11  })
12  export class CandidatModule {}
13
```

Figure 10 :candidat.module.ts

- ✓ **candidat.dto.ts** : Ce fichier définit un modèle de données qui représente les données échangées entre le client et le serveur. Il permet de valider et de formater les données reçues dans les requêtes HTTP avant de les traiter.

```
nestjs > src > candidat > dto > TS candidat.dto.ts > ...
1  import {ApiProperty} from '@nestjs/swagger'
2
3  export class createCandidatDTO {
4      @ApiProperty()
5      numCIN: string;
6
7      @ApiProperty()
8      mailCandidat: string;
9
10     @ApiProperty()
11     dateCandidature?:Date;
12
13     @ApiProperty()
14     noteCV?:Number;
15
16     @ApiProperty()
17     noteVideo?:Number;
18
19     @ApiProperty()
20     moyenne?:Number;
21
22     @ApiProperty()
23     idOffre: string;
24
25     @ApiProperty()
26     idFicCV: string;
27
28     @ApiProperty()
29     idFicInfosCV?:string;
30
31     @ApiProperty()
32     idFicVideo?:string;
33
34     @ApiProperty()
35     IdQuiz?:string;
36
37     @ApiProperty()
38     CV?:any;
39 }
```

Figure 11 : candidat.dto.ts

- ✓ **candidat.schemas.ts (facultatif)** : Ce fichier définit le schéma de la base de données associé à l'entité "candidat" si vous utilisez une base de données NoSQL comme MongoDB. Il spécifie la structure des données stockées dans la base de données, y compris les champs, les types de données et les relations.

```
nestjs > src > candidat > schemas > TS candidat.schemas.ts > Candidat
1  import { Schema, Prop, SchemaFactory } from "@nestjs/mongoose";
2
3  @Schema()
4  export class Candidat {
5      @Prop()
6      numCIN: string;
7
8      @Prop()
9      mailCandidat: string;
10
11     @Prop()
12     dateCandidature?: Date;
13
14     @Prop()
15     noteCV?: Number;
16
17     @Prop()
18     noteVideo?: Number;
19
20     @Prop()
21     moyenne?: Number;
22     |
23     @Prop()
24     idOffre: string;
25
26     @Prop()
27     idFicCV: string;
28
29     @Prop()
30     idFicInfosCV?: string;
31
32     @Prop()
33     idFicVideo?: string;
34
35     @Prop()
36     IdQuiz?: string;
37 }
38
39 export const CandidatSchemas = SchemaFactory.createForClass(Candidat)
```

Figure 12 : candidat.schemas.ts

En résumé, ces cinq fichiers travaillent ensemble pour permettre la création, la manipulation et la gestion des données de l'entité "candidat" dans une application NestJS.

b) Gérer l' Upload de fichier

Pour gérer l'upload de fichiers en NestJS, il est possible d'utiliser le module intégré de Nest basé sur le middleware multer pour Express. Multer gère les données postées dans le format multipart/form-data, qui est principalement utilisé pour télécharger des fichiers via une requête HTTP POST. Ce module est entièrement configurable et son comportement peut être ajusté en fonction des exigences de l'application. Pour télécharger un seul fichier, il suffit d'attacher l'intercepteur `FileInterceptor()` à la fonction de gestionnaire de route et d'extraire le fichier de la demande à l'aide du décorateur `@UploadedFile()`. Pour valider les métadonnées des fichiers entrants, comme la taille ou le type MIME du fichier, il est possible de créer son propre tuyau et de le lier au paramètre annoté avec le décorateur `UploadedFile()`

```
14 @Post('uploadCV')
15 @ApiOperation({ summary: 'Upload de CV' })
16 @ApiResponse({ status: 200, description: 'CV téléchargé sur le serveur', type: [Object] })
17 @UseInterceptors(FileInterceptor('file', {
18   storage: diskStorage({
19     destination: './uploads/CV',
20     filename: (req, file, cb) => {
21       const randomName = Array(32).fill(null).map(() => Math.round(Math.random() * 16).toString(16)).join('');
22       return cb(null, `${randomName}${extname(file.originalname)}`);
23     }
24   })
25 }))
26 async uploadFile(@UploadedFile() file) {
27   console.log("saved file",file);
28   return file;
29 }
30
31 @Post('uploadVideo')
32 @ApiOperation({ summary: 'Upload de Video' })
33 @ApiResponse({ status: 200, description: 'Video téléchargé sur le serveur', type: [Object] })
34 @UseInterceptors(FileInterceptor('file', {
35   storage: diskStorage({
36     destination: './uploads/video',
37     filename: (req, file, cb) => {
38       const randomName = Array(32).fill(null).map(() => Math.round(Math.random() * 16).toString(16)).join('');
39       return cb(null, `${randomName}${extname(file.originalname)}`);
40     }
41   })
42 }))
43 async uploadVideo(@UploadedFile() file) {
44   return file
45 }
```

Figure 13 : Gérer l' upload de fichier

c) Gérer l'envoi de mail

D'abord, il faut importer MailerModule dans MailModule et configurer le transport du serveur de messagerie via SMTP. Fournir une adresse e-mail par défaut pour utiliser systématiquement le même courrier dans toute l'application. Il est toujours possible de remplacer la valeur par défaut si nécessaire. Dernière étape, il faut configurer le dossier des modèles et l'adaptateur dans ce cas HandlebarsAdapter.

```
9  @Module({
10    imports: [
11      MailerModule.forRootAsync({
12        imports: [ConfigModule], // import module if not enabled globally
13        useFactory: async (config: ConfigService) => {
14          return ({
15            transport: {
16              host: config.get('MAIL_HOST'),
17              secure: false,
18              auth: {
19                user: config.get('MAIL_USER'),
20                pass: config.get('MAIL_PASSWORD'),
21              },
22            },
23            defaults: {
24              from: `"No Reply" <${config.get('MAIL_FROM')}>`,
25            },
26            template: {
27              dir: join(__dirname, 'templates'),
28              adapter: new HandlebarsAdapter(),
29              options: {
30                strict: true,
31              },
32            },
33          })
34        },
35        inject: [ConfigService],
36      }),
37    ],
38    providers: [MailService],
39    exports: [MailService],
40    controllers: [MailController],
41  })
```

Figure 14 : mail.module.ts

Puis il faut créer le premier modèle d'e-mail confirmation.hbs dans le dossier src/mail/templates. Ajouter le modèle simple suivant pour une confirmation d'utilisateur.

```
nestjs > src > mail > templates > AccuseDeReception.hbs > ...
1 |<!DOCTYPE html>
2 |<html lang="en">
3 |<head>
4 |   <meta charset="UTF-8">
5 |   <meta name="viewport" content="width=device-width, initial-scale=1.0">
6 |   <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/css/bootstrap.min.css" rel="stylesheet" integrity="sha384-EVSTQN3/azprG1Anm3QD
7 |
8 |   <title>Document</title>
9 |</head>
10|<body>
11|  <h5 class="text-danger">Bonjour,</h5>
12|
13|  <div>Nous avons bien reçu votre candidature pour le poste de <h5>{{nomDuPoste}}</h5> et nous vous remercions de l'intérêt que vous portez à
14|
15|  <div>Nous avons bien pris en compte votre CV et nous allons l'examiner attentivement. Si votre profil correspond à nos attentes, nous vous c
16|
17|  <div>Nous vous remercions encore une fois pour votre candidature et nous vous souhaitons bonne chance dans vos recherches d'emploi.</div><br
18|
19|  <div>Cordialement,</div>
20|  <h5>{{nomEntrePrise}}</h5>
21|
22|  <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/js/bootstrap.bundle.min.js" integrity="sha384-MrcW6ZMFY1zcL88N1+NtUVF0sA7MsXs
23|</body>
24|</html>
```

Figure 15 : Modèle du mail à envoyer 'templates.hbs'

Voir dans ANEXE A pour en savoir un peu plus sur les fichier handelbars(*.hbs)

Par défaut, Nest ne distribue que les fichiers TypeScript compilés (.js et .d.ts) lors de l'étape de construction. Pour distribuer les fichiers .hbs, il faut ouvrir le fichier nest-cli.json et ajouter le répertoire de modèles à la propriété assets dans les compilerOptions globales.

```
{
  "collection": "@nestjs/schematics",
  "sourceRoot": "src",
  "compilerOptions": {
    "assets": ["mail/templates/**/*.hbs"], // 📁 or "**/*.hbs" all files
    "watchAssets": true // 📺 copy assets in watch mode
  }
}
```

Figure 16 : fichier de configuration de nestjs

En fin, il faut Intégrer MailerService au MailService et mettre en place la logique d'envoi de courrier ici.

```
nestjs > src > mail > TS mail.service.ts > MailService
1  import { MailerService } from '@nestjs-modules/mailer';
2  import { Injectable } from '@nestjs/common';
3  @Injectable()
4  export class MailService {
5      constructor(private mailerService: MailerService) {}
6
7      envoiAccuseDeReception(userEmail) {
8          return this.mailerService.sendMail({
9              to: userEmail,
10             from: 'webdevin 2023' + ' <devinrecruithub@gmail.com>',
11             subject: 'Accusé de réception - Candidature reçue',
12             template: './AccuseDeReception', // `.hbs` extension is appended automatically
13             context: {
14                 nomDuPoste: "developpeur web",
15                 nomEntrePrise: "webdevin 2023"
16             },
17         });
18     }
19 }
```

Figure 17 : mail.service.ts

d) Test unitaires:

Les tests unitaires, ou "unit tests" en anglais, sont une pratique courante dans le développement logiciel. Ils consistent à tester individuellement des composants ou des unités de code, tels que des fonctions, des méthodes , afin de s'assurer qu'ils fonctionnent correctement

```
54  //tester la recuperation des informations sur un candidat
55  it('should return one "candidat" ', async () => {
56      const id=""
57      const candidat = {
58          _id:new ObjectId(),
59          numCIN: '',
60          mailCandidat: '',
61          idOffre: '',
62          idFicCV: ''
63      };
64      const saveSpy = jest.spyOn(service, 'findById').mockReturnValue(candidat);
65
66      const result = await service.findById(id);
67      expect(saveSpy).toHaveBeenCalledWith(id);
68      expect(result).toEqual(candidat);
69  });
70
71  });
```

Figure 18 : un extrait de test unitaire

e) Test d'intégration:

Un test d'intégration est une phase de tests dans le développement informatique qui vérifie le bon fonctionnement d'une partie précise d'un logiciel ou d'une portion d'un programme (appelée « unité » ou « module »). Cette phase de test est précédée par les tests unitaires et généralement suivie par les tests de validation. Dans le test d'intégration, chacun des modules indépendants du logiciel est assemblé et testé dans l'ensemble. L'objectif de chaque phase de test est de détecter les erreurs qui n'ont pas pu être détectées lors de la précédente phase. Le test d'intégration permet également de vérifier l'aspect fonctionnel, les performances et la fiabilité du logiciel.

```
nestjs > src > candidat > ts candidat.controller.spec.ts > describe('CandidatController') callback
1  import { Test } from '@nestjs/testing';
2  import { CandidatController } from './candidat.controller';
3  import { CandidatService } from './candidat.service';
4  import { createCandidatDTO } from './dto/candidat.dto';
5  import { getModelToken } from '@nestjs/mongoose';
6  import { Model } from 'mongoose';
7  import { Candidat } from './schemas/candidat.schemas';
8
9  describe('CandidatController', () => {
10   let candidatController: CandidatController;
11   let candidatService: CandidatService;
12
13   beforeEach(async () => {
14     const moduleRef = await Test.createTestingModule({
15       controllers: [CandidatController],
16       providers: [
17         CandidatService,
18         {provide: getModelToken(Candidat.name), useValue: Model<Candidat>},
19       ],
20     }).compile();
21
22     candidatService = moduleRef.get<CandidatService>(CandidatService);
23     candidatController = moduleRef.get<CandidatController>(CandidatController);
24   });
25
26   describe('saveCandidat', () => {
27     it('should return a Candidat object', async () => {
28       const candidat: createCandidatDTO = {
29         numCIN: '',
30         mailCandidat: '',
31         idOffre: '',
32         idFicCV: ''
33       };
34       jest.spyOn(candidatService, 'saveCandidat').mockImplementation(() => Promise.resolve(candidat));
35       expect(await candidatController.saveCandidat(candidat)).toBe(candidat);
36     });
37   });
38 }
```

Figure 19 : extrait de test d'integration

f) Résultats des test unitaires et des test d'intégration:

Pour lancer les tests il faut lancer la commande '`npm run test`' dans la racine du projet.

```
tsilavo@tsilavo-dynabook-Satellite-B551-C:/media/tsilavo/Disque local/ENI/STAGE_L2/Application/nestjs$ npm run test
> backend@0.0.1 test
> jest

RUNS  src/candidat/candidat.controller.spec.ts
RUNS  src/mail/mail.service.spec.ts
RUNS  src/fichier/fichier.service.spec.ts

Test Suites: 0 of 4 total
Tests:       0 total
Snapshots:   0 total
Time:        11 s, estimated 36 s
```

Figure 20 : Lancement des test

Si tout va bien, le résultat devrait être comme le montre la 'figure 19'

```
tsilavo@tsilavo-dynabook-Satellite-B551-C:/media/tsilavo/Disque local/ENI/STAGE_L2/Application/nestjs$ npm run test
> backend@0.0.1 test
> jest

(node:30315) [DEP0040] DeprecationWarning: The 'punycode' module is deprecated. Please use a userland alternative instead.
(Use 'node --trace-deprecation ...' to show where the warning was created)
PASS  src/mail/mail.service.spec.ts (40.126 s)
(node:30317) [DEP0040] DeprecationWarning: The 'punycode' module is deprecated. Please use a userland alternative instead.
(Use 'node --trace-deprecation ...' to show where the warning was created)
PASS  src/fichier/fichier.service.spec.ts (46.675 s)
(node:30306) [DEP0040] DeprecationWarning: The 'punycode' module is deprecated. Please use a userland alternative instead.
(Use 'node --trace-deprecation ...' to show where the warning was created)
PASS  src/candidat/candidat.controller.spec.ts (46.945 s)
PASS  src/candidat/candidat.service.spec.ts (9.064 s)

Test Suites: 4 passed, 4 total
Tests:       13 passed, 13 total
Snapshots:   0 total
Time:        51.457 s
Ran all test suites.
```

Figure 21 : test réussi

g) Quand faut-il lancer les test?:

Les tests unitaires sont effectués pendant le processus de développement, tandis que les tests d'intégration sont effectués après la fin des tests unitaires . Les tests unitaires ont une portée limitée à un seul module et visent à vérifier le comportement du module en isolation, c'est-à-dire que son contrat d'interface est rempli. Les tests d'intégration ont une portée qui couvre plusieurs modules et visent à vérifier la composition de plusieurs modules en s'assurant que les interactions n'entraînent pas d'échec. Il est important de distinguer les différents niveaux de test pour avoir une vision claire de la couverture de code. Si l'indicateur de couverture de code ne fait pas de distinction, un bon score peut cacher une carence en test d'un niveau, voire des deux.

Aller dans ANEXE B pour savoir un peu plus sur la phase de test.

8. 3. 2) Côté Frontend(NextJs)

Next.js est un framework JavaScript qui permet de créer des applications web et des sites statiques. Il est basé sur React et offre une expérience de développement rapide et facile.

React est une bibliothèque JavaScript pour la création d'interfaces utilisateur. Next.js est un framework qui s'appuie sur React pour fournir des fonctionnalités supplémentaires telles que le routage, les appels d'API, l'authentification et plus encore.

Voici quelques différences clés entre React et Next.js :

- **Rendu côté client vs rendu côté serveur** : React utilise le rendu côté client, ce qui signifie que le code JavaScript s'exécute dans le navigateur de l'utilisateur pour générer l'interface utilisateur. Next.js utilise le rendu côté serveur, ce qui signifie que le code JavaScript s'exécute sur le serveur pour générer l'interface utilisateur avant de l'envoyer au navigateur de l'utilisateur.
- **Pré-rendu** : Next.js prend en charge le pré-rendu, ce qui signifie que les pages peuvent être générées à l'avance et stockées en cache pour une livraison plus rapide aux utilisateurs.
- **Fonctionnalités intégrées** : Next.js offre des fonctionnalités intégrées telles que le routage, les appels d'API, l'authentification et plus encore, tandis que React nécessite des bibliothèques tierces pour ces fonctionnalités.

En résumé, Next.js est un framework qui offre une expérience de développement plus complète et plus performante que React. Il est idéal pour les applications web et les sites statiques qui nécessitent des fonctionnalités avancées telles que le rendu côté serveur et le pré-rendu.

Voici quelques exemples d'utilisation de Next.js :

- **Sites web dynamiques** : Next.js est idéal pour la création de sites web dynamiques qui nécessitent un rendu côté serveur. Par exemple, un site web de commerce électronique peut utiliser Next.js pour générer des pages de produits à l'avance et les stocker en cache pour une livraison plus rapide aux utilisateurs.
- **Sites statiques** : Next.js peut également être utilisé pour créer des sites statiques. Par exemple, un site web de portfolio peut utiliser Next.js pour générer des pages statiques à partir de contenu statique.

a) Structure de l'application (route)

Next.js utilise un système de routage basé sur les dossiers. Chaque dossier représente un segment de route, qui est mappé à un segment d'URL. Pour créer une route imbriquée, vous pouvez imbriquer des dossiers les uns dans les autres. Un fichier spécial `page.js` est utilisé pour rendre les segments de route accessibles au public.

Le routeur de Next.js est flexible et puissant. Il prend en charge les fonctionnalités suivantes :

- **Routage dynamique** : les pages peuvent être générées à la volée en fonction des paramètres de l'URL.
- **Pré-rendu** : les pages peuvent être générées à l'avance et stockées en cache pour une livraison plus rapide aux utilisateurs.
- **Gestion des erreurs** : Next.js offre une gestion des erreurs intégrée pour les erreurs 404 et 500, ainsi que la possibilité de personnaliser les pages d'erreur.

Donc comme le montre ma figure 21, pour postuler on a l'adresse suivante:

http://localhost:3000/postuler/form/id_poste

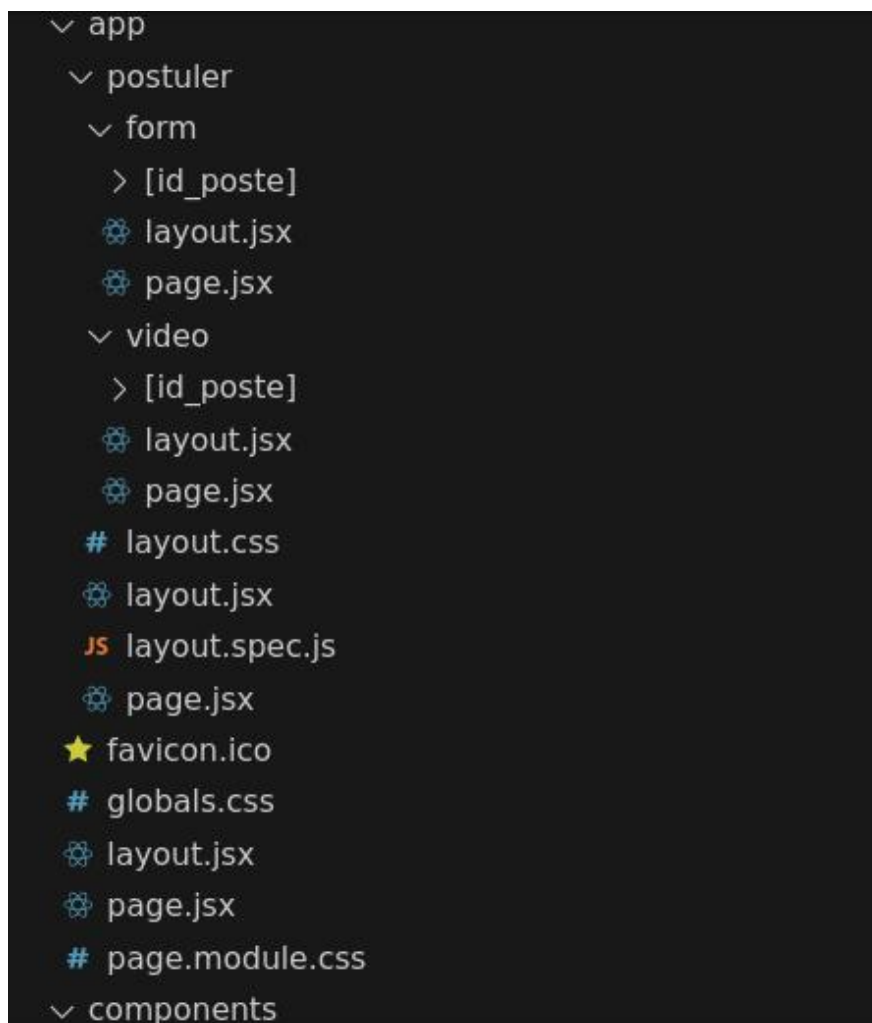


Figure 22 : gestion de route de l'application

i. Le fichier 'page.jsx'

Le fichier `page.js` est utilisé pour définir les pages de l'application. Chaque fichier `page.js` correspond à une route de l'application. Par exemple, si vous avez un fichier `page.js` dans le dossier `./app/about`, cela signifie que vous avez une route `/about` dans votre application. Le contenu de chaque fichier `page.js` est un composant React qui renvoie le contenu de la page.

```
app > postuler > form > [id_poste] > page.jsx > Postuler
1 import '@app/postuler/form/[id_poste]/page.css'
2 import Offre from '@components/jsx/Offre'
3 import Candidature from '@components/jsx/Candidature'
4
5 export default function Postuler({params}) {
6   return <div id="__container" className="container-fluid d-flex align-items-center justify-content-center flex-column flex-md-row p-0">
7     <div className="col-12 col-md-6 m-0 d-flex align-items-center justify-content-center __containre_child">~
9   </div>
10   <div className="col-12 col-md-6 m-0 d-flex align-items-center justify-content-center __containre_child">
11     <Candidature id_poste={params.id_poste}></Candidature>
12   </div>
13 </div>
14 }
```

Figure 23 : fichier `./app/postuler/form/[id_poste].page.jsx`

ii. le fichier 'layout.jsx'

Le fichier `layout.js` est utilisé pour définir la mise en page de l'application. Il enveloppe chaque page de l'application. Vous pouvez définir un fichier `layout.js` pour chaque page ou un fichier `layout.js` pour plusieurs pages. Le contenu de chaque fichier `layout.js` est également un composant React qui renvoie la mise en page de l'application. Les fichiers `layout.js` définissent la mise en page de l'application. Ils enveloppent les pages de l'application et peuvent être utilisés pour ajouter des éléments tels que des en-têtes, des pieds de page et des barres latérales. Vous pouvez utiliser des fichiers `layout.js` pour définir des mises en page partagées qui peuvent être utilisées par plusieurs pages dans l'application.

```
app > postuler > form > layout.jsx > FormLayout
1 export const metadata = {
2   title: 'Postuler a un poste | Formulaire',
3   description: 'Postuler a un poste(offre d\'emploi)',
4 }
5
6 export default function FormLayout({children}) {
7   return <> {children} </>
8 }
```

Figure 24 : le fichier `./app/postuler/form/layout.jsx`

Les composants renvoyés par les fichiers `page.js` et `layout.js` sont des composants React qui peuvent être personnalisés en fonction des besoins de l'application. Les composants renvoyés par les fichiers `page.js` sont des composants de page, tandis que les composants renvoyés par les fichiers `layout.js` sont des composants de mise en page.

b) Gestion des requêtes vers le backend (NestJS)

La façon la plus simple de faire les requêtes vers les api backend est de créer une fonction pour chaque requêtes a effectuer comme le montre la figure 24,

```
app > postuler > layout.jsx > ...
135 export function uploadCV(file) {
136   const formData = new FormData()
137   formData.append('file', file);
138
139   return axios.post('/fichier/uploadCV', formData, {
140     headers: {
141       'Content-Type': 'multipart/form-data'
142     }
143   });
144 }
145
146 export function uploadVideo(fic) {
147   return axios.post('/fichier/uploadVideo', fic, {
148     headers: {
149       'Content-Type': 'multipart/form-data'
150     }
151   });
152 }
153
154 export function saveFile(fichier){
155   return axios.post('/fichier/save',fichier)
156 }
157
158 export function saveCandidatReq(newCand) {
159   return axios.post('candidat',newCand)
160 }
161
162 export function verifierCandidat({numCIN,idOffre}) {
163   return axios.post('/candidat/verifierCandidat',{numCIN,idOffre})
164 }
165
166 export function sendAR(email) {
167   return axios.post('/mail/envoiAccuseDeReception',{
168     to:email
169   })
170 }
171
172 export function saveVideo(video){
173   return axios.post('/fichier/saveVideo',video)
174 }
```

Figure 25 : Déclaration des fonction pour faire les requêtes vers les api backend

Puis appeler les fonction en enchaînant les appel asynchrones et .then(), .catch() comment le montre la figure 25

```

const saveCandidat=() => {
  let idFichierCV=""

  const {numCIN,mailCandidat,idFicCV,CV}=candidat;

  const valid=((CV!=null) && (numCIN.trim()!="") && (mailCandidat.trim()!="") && (video!=null));

  console.log("confirmed:",confirm,"candidat:",candidat,"video:",video);
  if(valid && confirm){
    setShowLoadingModal(true)
    verifierCandidat(candidat).then(({data})=>{
      setEtape("Upload CV")
      if(data === ""){
        return uploadCV(CV)
      }else throw new Error("Le candidat avec ce numero CIN a déjà postuler a cet offre")
    }).then(({data})=>{//UPLOAD CV
      const CV={
        nomFic:data.filename,
        typeFic:data.mimetype,
        categorie:'CV'
      };
      return saveFile(CV)//ENREGISTRER LE FICHER DANS LA BASE DE DONNES
    }).then(({data})=>{
      idFichierCV=data._id
      setEtape("Upload Video")
      return uploadVideo(video)
    }).then(({data})=>{
      const video={
        nomFic:data.filename,
        typeFic:data.mimetype,
        categorie:'video'
      };
      return saveVideo(video)//ENREGISTRER LA VIDEO DANS LA BASE DE DONNES
    }).then(({data}) => {
      const idFichierVideo=data._id

      const tmp={
        numCIN : candidat.numCIN,
        mailCandidat : candidat.mailCandidat,
        idOffre : candidat.idOffre,
        idFicCV : idFichierCV,
        idFicVideo : idFichierVideo
      }

      console.log(tmp);
      setEtape("Enregistre la candidature")
      return saveCandidatReq(tmp)//ENREGISTRE LE CANDIDAT DANS LA BASE DE DONNES
    }).then(({data})=>{
      setEtape('Finalisation')
      return sendAR(data.mailCandidat)
    }).then(({data})=>{
      console.log(data);
      setEtape('')
      setShowLoadingModal(false)
      setShowSuccessModal(true)
    })
    .catch((err)=>{
      setShowLoadingModal(false)
      console.error("Error",err.toString());
      errorRef.current.innerText=err.toString();
      errorRef.current.classList.remove('d-none');
      setTimeout(() => {
        errorRef.current.innerText=""
        errorRef.current.classList.add('d-none');
      }, 4000);
    })
  }
}

```

Figure 26 : Exécution des requêtes vers la base de données

c) Gestion des requêtes vers le backend (NestJS)

Pour faire des tests unitaires dans une application Next.js, il est recommandé d'utiliser Jest, un framework de test JavaScript populaire. Jest est facile à configurer et à utiliser,

Exemple:

- La figure 26 montre la déclaration du composant Dropzone

```
export default function DropZone({file,setFile}) {
  const [showAlert,setShowAlert]=useState(false)

  const onDrop = useCallback(acceptedFiles => {
    const extension = acceptedFiles[0].name.split('.')[0].split('.').length-1;
    if (extension==='pdf')setFile(acceptedFiles[0])
    else{
      setShowAlert(true)
      setTimeout(() => {
        setShowAlert(false)
      }, 3000);
    }
  }, [])
  const {getRootProps, getInputProps, isDragActive} = useDropzone({onDrop})

  return <>
    {
      showAlert && (
        <div className="alert alert-danger position-absolute" role="alert">
          PDF seulement
        </div>
      )
    }
    <div data-testid="dropzone_container" className="d-flex flex-column align-items-center border-primary rounded __drop_zone" {...getRootProps()}>
      <input data-testid='_file_input' {...getInputProps()} />
      {
        isDragActive ?
        <p className='_dropzone_txt'>Deposer le CV ici ...</p> :
        <p className='_dropzone_txt'>Deposer le CV ici ou cliquer pour selectionner un fichier (.pdf seulement)</p>
      }
      <Image src="/file_input.png" width={50} height={50} alt=''></Image>
    </div>
    {file ? <FilePreview file={file}></FilePreview>:''}
  </>
}
```

Figure 27 : Déclaration du composant dropzone

- Pour tester que le composant Dropzone est bien rendu, il faut utiliser la fonction render() et l'objet screen depuis '@testing-library/react' comment le montre la figure 27

```
components > __test__ > js DropZone.spec.js > ...
1  import {render,screen} from '@testing-library/react'
2  import '@testing-library/jest-dom'
3  import DropZone from '../jsx/DropZone'
4
5  test('should render component "DropZone"', () => {
6    render(<DropZone/>)
7    expect(screen.getByTestId('_dropzone_container')).toBeInTheDocument();
8    expect(screen.getByTestId('_file_input')).toBeInTheDocument();
9  })
```

Figure 28 : test unitaire sur le composant dropzone

8. 4) Présentation de l'application :

Le processus de recrutement pour le projet de développement et de mise en œuvre d'une plateforme web de classification de CV basée sur l'analyse des critères du poste peut être décomposé en plusieurs étapes:

8. 4. 1) Création de l'offre d'emploi par le recruteur :

- Le recruteur se connecte à la plateforme web.
- Il crée un poste en spécifiant les détails de l'offre d'emploi, y compris le titre du poste, la date de début de la candidature et la date limite de candidature. La figure 28 montre l'interface de recrutement.

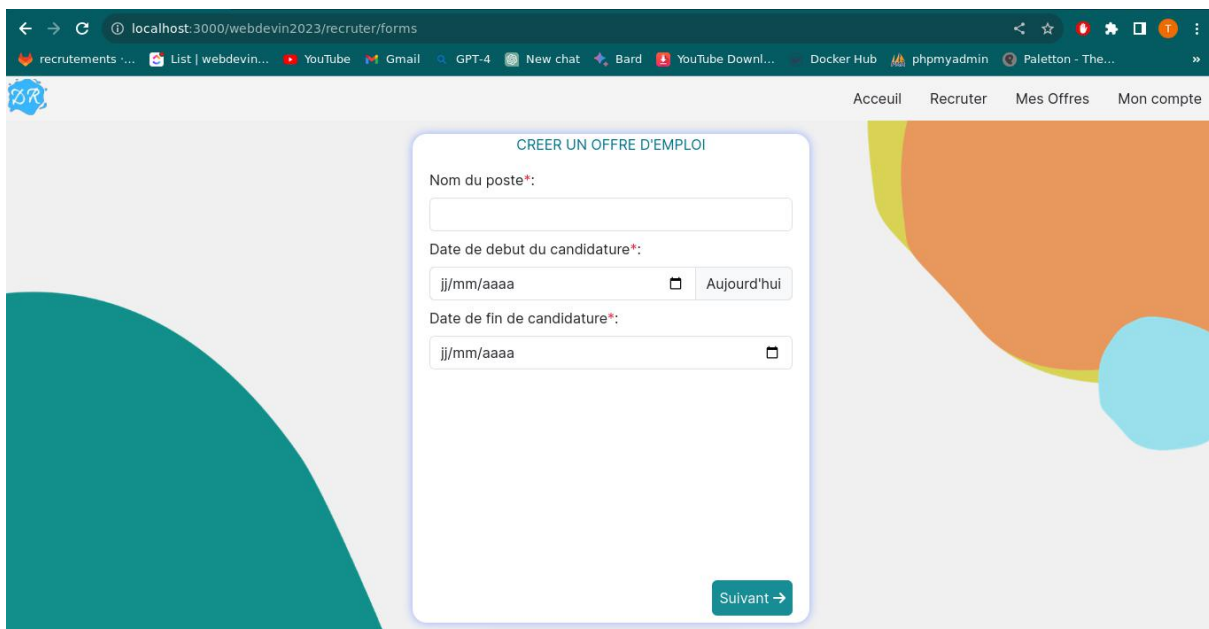


Figure 29 : formulaire de recrutement

- Puis le recruteur arrive sur l'interface pour décrire l'offre d'emploi plus en détail.

Qu'est ce qu'on met généralement dans la description de l'offre?

- ✓ Les responsabilités principales
- ✓ Les objectifs du poste
- ✓ Les qualifications requises
- ✓ Lieu de travail
- ✓ Mettez en évidence les avantages offerts aux employés

La figure 29 montre l'interface pour permettre au recruteur de remplir la description de l'offre.

Figure 30 : interface pour remplir la description de l'offre

- L'offre créée, le recruteur a la possibilité de publier l'offre avec le lien pour la candidature comme le montre la figure 30.

Figure 31 : Publier un offre d'emploi

Le recruteur a trois choix pour publier l'offre d'emploi:

- ✓ Soit en donnant directement le lien aux candidats
- ✓ Soit en le cliquant sur le bouton publier sur linkedin
- ✓ Soit sur facebook

8. 4. 2) Candidature des candidats :

Les candidats intéressés cliquent sur le lien de l'offre publiée par le recruteur pour postuler puis ils arrivent sur l'interface présentée par la figure 31.

The screenshot shows a web application interface for submitting a job application. On the left, a card displays the job title 'Développeur Nextjs/React Junior', the recruiter's name 'webdevin2023', and a description of the role. On the right, a form titled 'ENVOYER UNE CANDIDATURE' contains fields for 'CIN*', 'Email*', and 'CV*'. The CV field has a file upload icon and a warning message. A 'Suivant' button is at the bottom right.

Développeur Nextjs/React Junior

Recruteur:

webdevin2023
"Boite informatique a Fianarantsoa"

Description de l'offre:

L'entreprise **Webdevin** recherche un développeur web front-end talentueux et passionné pour rejoindre notre équipe de développement. En tant que développeur web front-end, vous serez responsable de la création et de la maintenance de l'interface utilisateur de nos applications web. Vous travaillerez en étroite collaboration avec nos concepteurs, nos développeurs back-end et

ENVOYER UNE CANDIDATURE

CIN*:

Email*:

CV*:

Deposer le CV ici ou cliquer pour selectionner un fichier (.pdf seulement)

vous ne pourrez plus postuler sur ce meme poste si vous continuez donc verifiez bien vos informations de candidature et CV avant de continuer.

☐ Confirmer les information de candidature et CV

Suivant →

Figure 32 : formulaire de candidature

Après que le candidat a rempli toutes les information il arrive sur l'interface pour enregistrer la vidéo comme le montre la figure 32:

The screenshot shows the same job card on the left. On the right, the interface has changed to a video recording screen titled 'ENVOYER UNE CANDIDATURE'. It includes a video player with a 'REC' indicator and a timer. Below the video are 'Précédent' and 'Envoyer' buttons.

Développeur Nextjs/React Junior

Recruteur:

webdevin2023
"Boite informatique a Fianarantsoa"

Description de l'offre:

L'entreprise **Webdevin** recherche un développeur web front-end talentueux et passionné pour rejoindre notre équipe de développement. En tant que développeur web front-end, vous serez responsable de la création et de la maintenance de l'interface utilisateur de nos applications web. Vous travaillerez en étroite collaboration avec nos concepteurs, nos développeurs back-end et

ENVOYER UNE CANDIDATURE

Veuillez repondre à cette question par video (MAX:5min):

... ?

REC 00:32

← Précédent

Envoyer →

Figure 33 : Interface pour enregistres la vidéo:

Et enfin le candidat n'a plus qu'à cliquer sur envoyer et confirmer la boîte de dialogue qui s'affiche et la candidature est fini, il n'a plus qu'à attendre que les recruteurs le contacte.

8. 4. 3) Accusée de réception par email :

Une fois les informations de candidature soumises, un accusé de réception est envoyé au candidat pour confirmer la réception de sa candidature.

La figure 33 montre un modèle d' email envoyée aux candidats



Figure 34 : Modèle d' email accusée de réception

Ce processus permet de gérer le recrutement de manière efficace en utilisant une plateforme web qui automatise une partie du processus de tri des candidats tout en offrant des fonctionnalités telles que la vidéo de présentation et les quiz pour évaluer les compétences des candidats.

CONCLUSION

L'École Nationale d'Informatique (ENI) est un établissement public professionnalisant placé sous tutelle de l'Université de Fianarantsoa. Actuellement, elle est habilitée à former des licenciés et des masters en Informatique. Ce stage s'est déroulé dans la société Webdevin où les recrutements et la classification des CV sont effectuées par le service RH(Relations Humaines).

L'objectif de ce projet est de créer et intégrer une plateforme web de recrutement pour automatiser les tâches pour les recrutements comme la classification des CV.

Pour réaliser ce projet, on a utilisé la méthode Merise, l'outil de conception Visual Paradigm, clickup et github comme outil de collaboration et gestion de projets, l'outil de design Figma, les framework NestJs pour le backend et NextJs pour le frontend, et mongodb comme SGBD.

L'application permet aux recruteurs de créer et publier une offre d'emploi sur les réseaux sociaux, et permette aux candidats de poster leurs candidatures grâce à un lien fourni par les recruteurs.

Ce logiciel est opérationnel et répond aux besoins du service de l'entreprise. Ce stage nous a permis de nous familiariser avec la vie de l'entreprise et d'approfondir nos connaissances pratiques en matière de développement des applications web.

BIBLIOGRAPHIE

[1] Gestion de projet agile , Véronique Messenger Rota , Préface de Jean Tabaka , e © Groupe Eyrolles, 2007, 2009, 2010, ISBN : 978-2-212-12750-8, 3 édition

[2] React.js , LE framework JavaScript de Facebook de Éric Sarrion, April 4, 2019

WEBOGRAPHIE

[1] <https://docs.nestjs.com/> , documentation de nestjs, consulté le 30 Août 2023

[2] <https://nextjs.org/docs> , documentation de nextjs, consulté le 30 Août 2023

[3] <https://www.mongodb.com/docs/> , documentation de mongodb, consulté le 31 Août 2023

[4] <https://tiptap.dev/introduction> , documentation de l'editeur de texte tiptat, consulté le 2 Novembre 2023

GLOSSAIRE

- **Backend** : La partie d'une application qui est invisible pour l'utilisateur et qui gère les fonctionnalités en arrière-plan. Le backend peut inclure des éléments tels que des serveurs, des bases de données, des API, etc.
- **Développement** : Le processus de création d'applications logicielles. Le développement peut inclure la conception, la programmation, les tests, le déploiement, etc.
- **Frontend** : La partie d'une application qui est visible et avec laquelle l'utilisateur interagit. Le frontend peut inclure des éléments tels que des pages web, des formulaires, des boutons, etc.
- **Intégration** : Le processus de combinaison de différents systèmes ou composants logiciels pour créer un système plus grand et plus complexe. L'intégration peut être utilisée pour connecter des applications, des bases de données, des services web, etc.
- **Plateforme** : Un environnement logiciel qui fournit des services et des outils pour développer et exécuter des applications. Les plateformes peuvent être utilisées pour créer des applications web, mobiles, de bureau, etc.
- **Requête** : Une demande envoyée à une base de données pour récupérer des informations. Les requêtes peuvent être utilisées pour récupérer des données, les modifier, les supprimer, etc.
- **Framework** : Un ensemble de bibliothèques, d'outils et de conventions de codage qui permettent aux développeurs de créer des applications plus rapidement et plus facilement. Les frameworks peuvent être utilisés pour développer des applications web, mobiles, de bureau, etc.

ANEXES

- **ANEXE A** : Les fichiers avec l'extension .hbs sont des fichiers de modèle JavaScript Handlebars.js. Ils contiennent des modèles écrits en code HTML et incorporant des expressions de guidon. Les fichiers avec l'extension .hbs peuvent être ouverts avec des logiciels tels que Visual Studio Code. Les fichiers .hbs sont couramment utilisés pour créer des modèles de pages Web dynamiques. Les modèles sont écrits en code HTML et contiennent des expressions de guidon qui sont ensuite interprétées par le moteur de modèle Handlebars.js pour générer du contenu dynamique . Les fichiers .hbs sont souvent utilisés dans les applications Web pour séparer la logique de présentation de la logique de l'application.
- **ANEXE B** : La phase de test est une étape cruciale dans le développement d'une plateforme web. Elle permet de vérifier que toutes les fonctionnalités de la plateforme fonctionnent correctement et qu'elle est prête à être déployée. Il existe différents types de tests d'application, tels que les tests de performance, les tests de fonctionnalité, les tests de charge, les tests d'interface utilisateur, etc. Les tests de performance évaluent la vitesse et la stabilité de la plateforme, tandis que les tests de fonctionnalité vérifient que toutes les fonctionnalités de la plateforme fonctionnent correctement. Les tests de charge évaluent la capacité de la plateforme à gérer un grand nombre d'utilisateurs simultanément, tandis que les tests d'interface utilisateur vérifient que l'interface utilisateur est conviviale et facile à utiliser . Il est important de tester la plateforme à chaque étape du développement pour s'assurer que tout fonctionne correctement. Les tests doivent être effectués à la fois manuellement et automatiquement pour garantir que toutes les fonctionnalités sont correctement testées
- **ANEXE C**: Les DFD, MCD et MCT sont des outils de modélisation des systèmes d'information qui permettent de représenter les flux de données, les données stockées et les processus exécutés dans un système. Ces outils sont complémentaires et permettent de comprendre de manière complète le fonctionnement d'un système, et il est important de bien comprendre le système a informatiser avant d' entrer dans la phase de développement de l'application.

TABLE DE MATIERE

CURICULUM VITAE.....	I
REMERCIEMENTS.....	II
LISTE DES FIGURES.....	III
LISTE DES TABLEAUX.....	IV
LISTE ABRÉVIATIONS.....	V
SOMMAIRE.....	VI
INTRODUCTION GENERALE.....	1
PARTIE I. PRESENTATION.....	2
chap 1: L'ENI:.....	2
1. 1) Information d'ordre général.....	2
1. 2) Missions et historique.....	2
1. 3) Organigramme institutionnel de l'ENI.....	4
1. 4) Domaine de spécialisation.....	6
1. 5) Architecture des formations pédagogiques.....	7
1. 6) Relations de l'ENI avec les entreprises et les organismes.....	10
1. 7) Partenariat au niveau international.....	12
1. 8) Débouchés professionnels avec des diplômés.....	14
1. 9) Ressources humaines.....	16
chap 2: Présentation de L' Entreprise.....	17
2. 1) Historique.....	17
2. 2) Notre Mission.....	17
2. 3) Organigramme.....	17
chap 3: Description du projet.....	18
3. 5) Moyens nécessaire a la réalisation du projet.....	19
3. 5. 1) Moyen humain.....	19
3. 5. 2) Moyen logiciel.....	20
3. 6) Moyen matériel.....	20
PARTIE II. ANALYSE ET CONCEPTION.....	21
chap 4: Analyse Préalable.....	21
chap 5: Analyse conceptuel.....	27
5. 1) Présentation de la méthode utilisé(Agile).....	27
chap 6: Conception détaillée.....	31
PARTIE III. REALISATION.....	34
chap 7: Mise en place de l'environnement de développement.....	34
7. 1) Installation et configuration des outils:.....	34
chap 8: Développement de l'application.....	38
8. 1) Création de la base de données.....	39
8. 2) Faire un design Figma.....	39
8. 3) Codage de l'application.....	40
8. 3. 1) Coté Backend(NestJS).....	40
a) Gérer la base de donnée.....	40
b) Gérer l' Upload de fichier.....	44
c) Gérer l' envoi de mail.....	45
d) Test unitaires:.....	47
e) Test d' intégration:.....	48
f) Résultats des test unitaires et des test d' intégration:.....	49
g) Quand faut-il lancer les test?:.....	49
8. 3. 2) Coté Frontend(NextJs).....	50
a) Structure de l'application (route).....	51
i. Le fichier 'page.jsx'.....	52
ii. le fichier 'layout.jsx'.....	52
b) Gestion des requêtes vers le backend (NestJS).....	53
c) Gestion des requêtes vers le backend (NestJS).....	55

8. 4) Présentation de l'application :	56
8. 4. 1) Création de l'offre d'emploi par le recruteur :	56
8. 4. 2) Candidature des candidats :	58
8. 4. 3) Accusée de réception par email :	59
CONCLUSION	60
BIBLIOGRAPHIE	VII
WEBOGRAPHIE	VIII
GLOSSAIRE	IX
ANEXES	X
TABLE DE MATIERE	XI
RÉSUMÉ	VII
ABSTRACT	VIII

RÉSUMÉ

Pour conclure ma deuxième année en Licence Professionnelle à l' École Nationale d'Informatique, j'ai effectué un stage de trois mois à la société Webdevin.

Durant ce stage, j'avais développée et implémente une plateforme web qui facilite les tâches pendant la phase de recrutement des nouveaux personnel en Next.js avec nestjs et mongodb pour stocker les données. Tout ceci portait sur la création d'une plateforme permettant non seulement de le terminer, mais aussi d'effectuer les tests indispensables et finir de rédiger les documents.

Cette application doit permettre a Webdevin de bien gérer et de bien les candidats pour leurs offres d'emplois.

Ce rapport explique les grands points de ce projet et des différentes impressions sur ce que le stage nous a apporté, sur ce que nous avons fait et appris.

ABSTRACT

To conclude my second year in the Professional License at the National School of Computer Science, I did a three-month internship at the Webdevin company.

tasks during the recruitment phase of new employees using Next.js with NestJS and MongoDB to store data. All of this was about creating a platform that not only allows to complete the recruitment process, but also to perform the necessary tests and finish drafting the documents.

This application should allow Webdevin to manage and well-qualify candidates for their job offers.

This report explains the main points of this project and the different impressions on what the internship brought us, what we did and learned.